ECHO-GL: Earnings Calls-Driven Heterogeneous Graph Learning for Stock Movement Prediction

Mengpu Liu[1], **Mengying Zhu**[1*], **Xiuyuan Wang**[2], **Guofang Ma**[3], **Jianwei Yin**[1, 2], **Xiaolin Zheng**[2]

[1] School of Software Technology, Zhejiang University, China
[2] College of Computer Science, Zhejiang University, China
[3] School of Computer Science and Technology, Zhejiang Gongshang University, China
{mengpuliu, mengyingzhu, xiuyuanwang, xlzheng}@zju.edu.cn, maguofang@zjgsu.edu.cn, zjuyjw@cs.zju.edu.cn

Abstract

Stock movement prediction serves an important role in quantitative trading. Despite advances in existing models that enhance stock movement prediction by incorporating stock relations, these prediction models face two limitations, i.e., constructing either insufficient or static stock relations, which fail to effectively capture the complex dynamic stock relations because such complex dynamic stock relations are influenced by various factors in the ever-changing financial market. To tackle the above limitations, we propose a novel stock movement prediction model `ECHO-GL` based on stock relations derived from earnings calls. `ECHO-GL` not only constructs comprehensive stock relations by exploiting the rich semantic information in the earnings calls but also captures the movement signals between related stocks based on multimodal and heterogeneous graph learning. Moreover, `ECHO-GL` customizes learnable stock stochastic processes based on the post earnings announcement drift (PEAD) phenomenon to generate the temporal stock price trajectory, which can be easily plugged into any investment strategy with different time horizons to meet investment demands. Extensive experiments on two financial datasets demonstrate the effectiveness of `ECHO-GL` on stock price movement prediction tasks together with high prediction accuracy and trading profitability.

Introduction

Stock movement prediction serves an important role in quantitative trading, which aims to predict the future trends of a stock in order to assist investors in making good investment decisions. Traditional solutions for stock movement prediction are based on deep time-series models, which treat stock movements as independent of each other, ignoring the valuable rich signals between related stocks' movements.

Recently, a burgeoning research trend has emerged, focusing on enhancing stock movement prediction by modeling stock relations, which can be divided into two categories, i.e., price-based methods (Li et al. 2021; Zhu et al. 2022) and side information-based methods (Feng et al. 2018; Sawhney et al. 2021). However, both categories of methods have their limitations as follows. On the one hand, price-based methods predict stock movement based on the estimated empirical stock correlation matrix (Li et al. 2021) or modeling predicted stock correlation (Zhu et al. 2022). However, relying solely on stock prices is sub-optimal for the real-world financial market because dynamic stock relations are influenced by multiple factors, e.g., macroeconomic, industry relations, company management, and investor perception. On the other hand, there are side information-based methods incorporating wiki or industry relations (Feng et al. 2018; Sawhney et al. 2021) to capture stock relations. However, these methods represent stock relations as static graphs, thus struggling to adapt to ever-changing markets. Based on the above analysis, effective modeling of stock relations necessitates being driven by dynamic information that encompasses multiple influencing factors. Fortunately, taking inspiration from previous studies (Qin and Yang 2019; Medya et al. 2022) that leverage earnings calls' rich semantic information to enhance financial forecasting, we explore the potential of utilizing earnings calls to model stock relations.

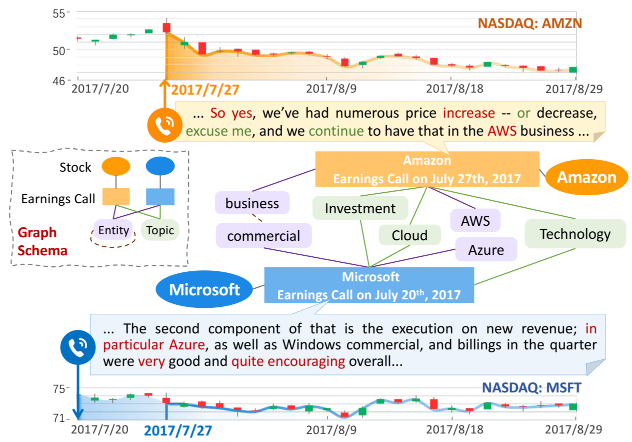

Figure 1: An example of earnings calls-driven stock relations' effect on stock movements.

Motivation Example: Taking two earnings calls in July 2017 from Amazon and Microsoft as an example, we first plot the stock price movements of Amazon and Microsoft after Amazon's earnings call on 27th July, along with the two earnings calls' main content in Figure 1. Then we depict a graph between the two stocks derived from their earnings calls. From Figure 1, we draw three observations below.

*Corresponding Author

First, subsequent to the announcement of Amazon's earnings call, the effect of this earnings call on Amazon's stock trend lasted for several weeks, resulting in a fluctuating stock decline. This observation is consistent with the widely documented Post Earnings Announcement Drift (PEAD) phenomenon (Qin and Yang 2019) in the literature. Second, the two earnings calls shared abundant overlapping topics and entities that indicate stock features, e.g., industry and business, demonstrating that rich semantics within earnings calls have the potential to reflect intricate stock relations. Third, stock prices of Amazon and Microsoft experienced similar downward trends, underscoring the capacity of stock relations to enhance prediction models by capturing valuable rich movement signals between related stocks.

In this paper, inspired by the aforementioned phenomenon on stock relations derived from earnings calls, we propose an **E**arnings **C**alls-driven **H**eter**O**geneous **G**raph **L**earning model `ECHO-GL` to model complex stock relations in an earnings calls-driven heterogeneous dynamic graph (termed as E-Graph) to enhance stock movement prediction. Our model processes follow three steps: (1) construct an E-Graph based on earnings calls to uncover the underlying stock relations; (2) leverage a heterogeneous graph learning module to learn the stock representations that aggregate useful movement signals on E-Graph; (3) build a time-varying stochastic process to model earnings call's PEAD effect on multiple time horizons for stock movement prediction.

Nevertheless, three challenges corresponding to the three steps above make it non-trivial to implement `ECHO-GL`.

Challenge I: How to construct E-Graph? Due to the non-stationary nature of markets, outdated data fails to capture the most current market dynamics. To tackle *Challenge I*, we design two novel mechanisms: the time assignment mechanism and the sliding window mechanism. Specifically, we apply the time assignment mechanism to assign time attributes to all nodes and edges, intending to preserve the dynamic nature of the heterogeneous graph. We design a sliding window mechanism to filter the most recent relevant information in the E-Graph to participate in stock relation modeling.

Challenge II: How to capture the stocks' spatial dependencies in E-Graph? The E-Graph encompasses diverse heterogeneous information, i.e., earnings calls, topics, and entities, which form complex relations and collectively reflect the spatial dependencies among stocks. To tackle *Challenge II*, we propose a novel stock spatial relational module in `ECHO-GL`, which preserves distinct feature spaces for each node type and edge type in the E-Graph and conducts cross-type features aggregation based on the attention mechanism. Moreover, considering multimodal information in earnings calls, during the aggregation, we introduce audio features to adjust the influence of each text sentence on the stock.

Challenge III: How to model the stocks' temporal dependencies? According to the PEAD phenomenon, the stock movement is not entirely stochastic in the near future after an earnings call announcement. To address *Challenge III*, we devise a learnable stochastic process in the post earnings call stock dynamics module for stock representations learned from the E-Graph. The post earnings call stock dynamics module can capture stock price changes in the effect of earnings calls over any near-future horizon, facilitating stock movement prediction on multiple time horizons.

Our main contributions are as follows: (1) this is the first work in the literature to model stock relations derived from earnings calls, which deeply captures dynamic stock relations based on earnings calls' rich semantic information; (2) we propose an `ECHO-GL` model with multimodal heterogeneous graph (E-Graph) construction and two specific modules, i.e., stock spatial relational module and post earnings call stock dynamics module, to capture the stocks' spatial-temporal relations influenced by earnings calls and model the latent PEAD effect on stock movements as a stochastic process, facilitating stock movement prediction in multiple time horizons; (3) we conduct extensive experiments on two real-world datasets, which verify the effectiveness of `ECHO-GL` on stock price movement prediction tasks together with high accuracy and trading profitability.

Related Work

Graph Learning in Stock Prediction

Graph learning has emerged as a crucial area in financial prediction tasks, which involves incorporating stock relations into the learning procedure to achieve improved performance. Stock relation graphs have previously been created based on historical prices from a risk perspective (Fan, Han, and Liu 2014; Ke, Lian, and Zhang 2020). Recently, an increasing number of studies have focused on graph learning based on integrating rich additional information, such as industry relations (Feng et al. 2018; Sawhney et al. 2021)and media reviews (Wang et al. 2022). These studies aim to learn more comprehensive representations to enhance stock prediction. There is a study (Medya et al. 2022) closely related to our approach that constructs graphs based on earnings calls. However, it only builds graphs for individual earnings call transcripts and does not consider the interconnections between different earnings calls and stock relations.

Earnings Calls in Stock Prediction

In light of the financial environment's rapid evolution, market-assisted information, such as news, analyst reports, social media, and financial conference calls, offers pervasive and fast-evolving unstructured financial data. Recently, a more promising approach has shown better results by exploiting semantic information (Keith and Stent 2019) from earnings calls during financial decision-making. Following this, a substantial group of methods focuses on fusing multimodal text-audio information in earnings calls for financial multitask prediction (Qin and Yang 2019; Yang et al. 2020, 2022; Sawhney et al. 2020b). Moreover, Medya et al. (Medya et al. 2022) proposed the STOCKGNN method based on Gated GNN (Li et al. 2015) to capture semantic features from earnings calls. However, the relations between stocks based on earnings calls have not been studied yet.

Problem Setting

We consider earnings call-based financial forecasting, which refers to predicting stock movements over multiple time horizons after the earnings call announcement. We consider a market with N stocks. Let $p_{s,t}$ denote a closing price of stock s on day t, and $\mathbf{p}_{s,\Delta t}$ denote the price series of stock s during Δt period, where Δt is from day $t - \tau + 1$ to day t, and τ is the window size. For each stock s, there are K earnings calls that have been announced. Let $\mathcal{C}_s = \{c_s^1, ..., c_s^K\}$ denote the set of all earnings calls corresponding to stock s, and $\mathcal{C} = \bigcup_{s=1}^{N} \mathcal{C}_s$ denotes earnings calls of all stocks. Following (Qin and Yang 2019; Li et al. 2020a), each $c \in \mathcal{C}$ has been segmented into a set of audio clips $\mathcal{A}_c = \{a_c^1, ..., a_c^M\}$ for $i \in [1, M]$, and M is the sentence number. The corresponding document-level transcript of earning call c is denoted as $\mathcal{D}_c = \{d_c^1, ..., d_c^M\}$ and d_c^i is the i-th text sentence.

The input and output for the prediction task are defined as follows:

Input: Window size τ, earnings calls in the Δt historical period $(\mathcal{C}, \mathcal{A}_{c\in\mathcal{C}}, \mathcal{D}_{c\in\mathcal{C}})$, all stocks' historical closing prices $\mathbf{p}_{\Delta t}$, multiple time horizons $\mathcal{W}$ for stock movement prediction tasks, e.g., $\mathcal{W} = \{1, 3, 7, 15, 30\}$, and target stock s.

Output: Stock movement prediction results $\{y^1, ..., y^{|\mathcal{W}|}\}$ under all prediction time horizons $\mathcal{W}$. Note that the ground truth $y_{s,t+w}$ is calculated as follows:

$$y_{s,t+w} = \begin{cases} 1, & p_{s,t+w} > p_{s,t} \\ 0, & p_{s,t+w} \le p_{s,t} \end{cases}, \tag{1}$$

where w is the prediction time horizon, $p_{s,t}$ is the closing price of target stock s on day t, and $p_{s,t+w}$ is the closing price of target stock s on day $t + w$.

Method

The architecture of `ECHO-GL` is presented in Figure 2. In the following subsections, we first elaborate on how to construct a heterogeneous graph derived from earnings calls (E-Graph), which fully exploits semantic relations between stocks in earnings calls (for addressing *Challenge I*). We then present two key modules of `ECHO-GL`, i.e., the stock spatial relational module and the post earnings call stock dynamics module, to capture the spatial and temporal dependencies in E-Graph (for addressing *Challenges II* and *III*). Finally, we elaborate on how to predict stock movement and train `ECHO-GL`.

Heterogeneous Stock Graph Structure Construction

Firstly, we denote the E-Graph with a four-tuple $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R}_\mathcal{V}, \mathcal{R}_\mathcal{E})$, where $\mathcal{V}$ is a node set, $\mathcal{E}$ is an edge set, $\mathcal{R}_\mathcal{V}$ is the set of node types, and $\mathcal{R}_\mathcal{E}$ is the set of edge types.

Then, we elaborate on the construction of each element.

Node set $\mathcal{V}$: The node set $\mathcal{V}$ contains four types of nodes marked with time attribute: stock price node (P) n^P, earnings call text sentence node (S) n^S, topic node (O) n^O, and entity node (E) n^E. We adopt LSTM (Sak, Senior, and Beaufays 2014) to encode stocks' historical price sequence to obtain stock price nodes. Following Hu et al. (Hu et al. 2019), we utilize LDA (Blei, Ng, and Jordan 2003) and TAGME[1] to extract topics and entities from each text sentence. Then, we generate initial representations of S, O, and E nodes by FinBERT (Araci 2019), which incorporates financial specialized linguistic features into earnings call modeling.

[1]https://sobigdata.d4science.org/group/tagme/

Algorithm 1: E-Graph construction algorithm

Input: Existing E-Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R}_\mathcal{V}, \mathcal{R}_\mathcal{E})$, incoming earnings call $(c_s, \mathcal{A}_{c_s}, \mathcal{D}_{c_s})$ on day t, stock s, graph window size τ.

Add stock price node of s: $\mathcal{V} = \mathcal{V} \cup n^P$
n^P.time $= t$
for *text sentence* $d_c^i \in \mathcal{D}_c$ **do**
  Add node: $\mathcal{V} = \mathcal{V} \cup n^{S_i}$
  Add edge: $\mathcal{E} = \mathcal{E} \cup (n^{S_i}, n^P, 0)$
  n^{S_i}.time $= t$
  for n^O / n^E *related to* n^{S_i} **do**
    if n^O / n^E *not in* $\mathcal{V}$ **then**
      Add node: $\mathcal{V} = \mathcal{V} \cup n^O$ / n^E
      if n^O / n^E *is entity node* **then**
        for *possible entity node pairs* $\{n^E, n^{E'}\}$ **do**
          if $\mathrm{COS}(n^E, n^{E'}) > \delta_c$ **then**
            Add edge: $\mathcal{E} = \mathcal{E} \cup (n^E, n^{E'}, 0)$
    Add edge: $\mathcal{E} = \mathcal{E} \cup (n^{S_i}, n^O/n^E, 0)$
    n^O/n^E.time $= n^{S_i}$.time
for $n \in \mathcal{V}$ *and* n*.time* $\le t - \tau$ **do**
  Remove all edges related to node n. Remove node n.
Update all edges' time $\Delta T(n_i, v_i) = |n_i.\text{time} - v_i.\text{time}|$.

Edge set $\mathcal{E}$: Edge set in E-Graph can be represented as a sequence of relations that come in over time, i.e., $\mathcal{E} = \{(n_1, v_1, \Delta T(n_1, v_1)), (n_2, v_2, \Delta T(n_2, v_2)), ...\}$, where $(n_i, v_i, \Delta T(n_i, v_i))$ corresponds to a relation between the node pair (n_i, v_i), and $\Delta T(n_i, v_i)$ is the time attribute of the edge. There are four types of edges in $\mathcal{E}$, i.e., P-S, S-O, S-E, and E-E, where (1) P-S refers to an edge between a stock price node and a text sentence node from the corresponding stock's earnings calls; (2) S-O refers to an edge between a text sentence node and a topic node with the relevance probability above a threshold δ_r; (3) S-E refers to an edge between a text sentence node and an entity node which is formed when the text sentence includes the respective entity; (4) E-E refers to an edge between two entities if their cosine similarity is above a threshold δ_c.

Node Type $\mathcal{R}_\mathcal{V}$: Each node $v \in \mathcal{V}$ is associated with the node type mapping function $\pi(v) : \mathcal{V} \to \mathcal{R}_\mathcal{V}$.

Edge Type $\mathcal{R}_\mathcal{E}$: Each edge $e \in \mathcal{E}$ is associated with the relation type mapping function $\phi(e) : \mathcal{E} \to \mathcal{R}_\mathcal{E}$.

Given the set of earnings calls $\mathcal{C}$ from N stocks in the Δt period, we aim to construct an E-Graph $\mathcal{G}$ that encompasses all the earnings calls' text sentences that occur within this period. In order to model the dynamic nature of the heteroge-

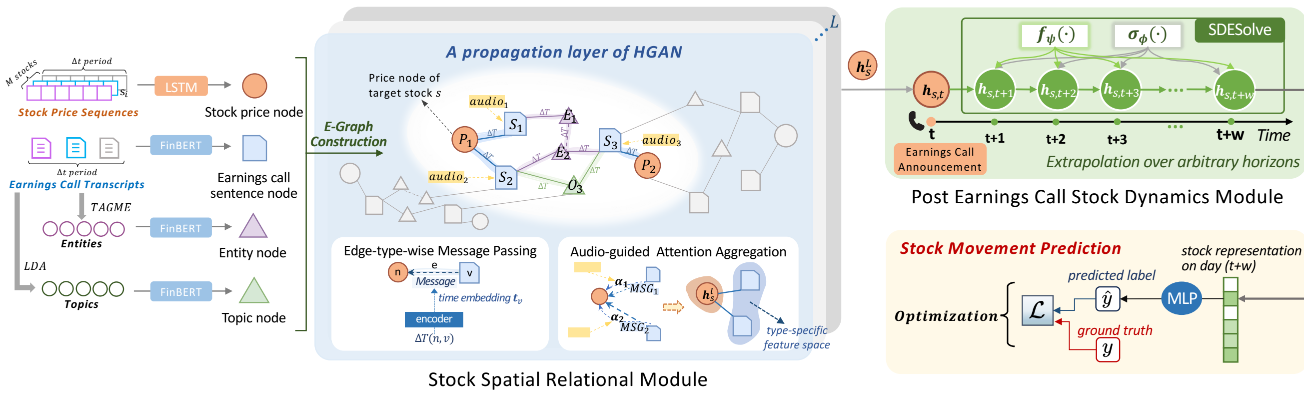

Figure 2: Network architecture of our model ECHO-GL.

neous graph, we design two mechanisms that assign a time attribute to all nodes and edges and maintain all edges and nodes within a specific window as a whole. The procedure is presented in Algorithm 1. Next, we present the details of the two crucial mechanisms.

Time assignment mechanism: Time plays an important role in the dynamic E-Graph; however, there are some nodes and edges that do not have time attributes. Therefore, we design a time assignment mechanism that can assign the announcement time of the most recent earnings call to all its associated heterogeneous nodes and edges. As shown in Lines 2, 6, 15, and 18 of Algorithm 1, when a new earnings call c_s of a stock s is announced, we add its corresponding stock price node and all its text sentence nodes into the E-Graph and assign the announcement time of c_s to these nodes as their time attributes. For the topics and entities associated with c_s' text nodes, we add new O and E nodes that are not present in the current E-Graph. Following that, we update the time attributes of all the nodes typed O and E related to c_s' text sentence nodes to match the announcement time of earnings call c_s. Next, we add the edges between these P, S, O, and E nodes to E-Graph and update the time attributes of all edges.

Sliding window mechanism: In order to accommodate the ever-changing market, we design a sliding window mechanism (refer to Lines 16-17 in Algorithm 1) to filter the most recent stock relations. Note that we set the window size as the announcement time gap of two consecutive earnings calls for the target stock. Whenever an outdated earnings call slides outside the window, all its P and S nodes with their connected edges are eliminated from the E-Graph.

Stock Spatial Relational Module

Within the E-Graph, various types of nodes with heterogeneous information coexist. Considering the heterogeneity in the graph and preserving specific domain knowledge of each node type, we design a heterogeneous graph attention-based neural network (HGAN) for learning the enhanced stock representations that incorporate various types of relational information from the E-Graph. The stock spatial relational module is formed by stacking L layers of HGAN, with our target node type being the stock price (P). HGAN follows a message-passing architecture, which includes two components, i.e., message passing and aggregation.

Heterogeneous Edge-type-wise Message Passing. Given a target node n and all its neighbor nodes $v \in \mathcal{N}(n)$ within the E-Graph, to ensure effective message propagation to n across different neighbor types without being restricted by their feature distribution gaps, we devise an edge-type-wise message passing strategy. Specifically, for an edge $e = (n, v)$ on the l-th layer, we calculate the multi-head message from node v to node n as follows:

$$\begin{aligned}\textbf{Message}(n,e,v) &= \mathop{\Big\|}_{i\in[1,h]} \text{MSG-head}^{\text{i}}(n,e,v)\\ \text{MSG-head}^{\text{i}}(n,e,v) &= \text{M-Linear}^{i}_{\pi(v)}\left(\mathbf{h}_v^{(l-1)}+\mathbf{t}_v\right)W^{\text{MSG}}_{\phi(e)},\end{aligned} \tag{2}$$

where $\mathbf{h}_v^{(l-1)} \in \mathbb{R}^d$ is the representation of node v on the $(l-1)$-th layer, $\mathbf{t}_v \in \mathbb{R}^d$ is the time embedding of v, $\text{M-Linear}^{i}_{\pi(v)}$ represents the linear projection specific to node type $\pi(v)$, and $W^{\text{MSG}}_{\phi(e)} \in \mathbb{R}^{\frac{d}{h}\times\frac{d}{h}}$ is a learnable edge type-specific matrix to capture the edge dependency.

Since two earnings calls' announcements have a time gap, we embed the time embedding $\mathbf{t}_v$ in Eq.(2) based on a relative time gap between the node pair (n, v), that is, the edge e's time attribute assigned by the time assignment mechanism. We encode the time embedding $\mathbf{t}_v$ based on random Fourier features (Wang et al. 2020) as follows:

$$\begin{aligned}\mathbf{t}_v = [&\sin(\omega_1\Delta T(n,v)), \cos(\omega_1\Delta T(n,v)),\\ &\cdots, \sin(\omega_{d/2}\Delta T(n,v)), \cos(\omega_{d/2}\Delta T(n,v))],\end{aligned} \tag{3}$$

where $\Delta T(n, v)$ is the time attribute of edge $e = (n, v)$, ω_i are learnable parameters, and d is the dimension of $\mathbf{t}_v$.

Audio-guided Attention Aggregation. We devise a heterogeneous node-level attention mechanism for aggregating messages on the E-Graph.

For node n, we map it into a Query vector and its neighbor node v into a Key vector and propagate h-head cross-attention for each relation $e = (n, v)$. To address the heterogeneity of nodes, for the i-th attention head, each node type has a unique linear projection:

$$K_v^i = \text{K-Linear}_{\pi(v)}^i \left(\mathbf{h}_v^{(l-1)} + \mathbf{t}_v\right)$$
$$Q_n^i = \text{Q-Linear}_{\pi(n)}^i \left(\mathbf{h}_n^{(l-1)}\right), \tag{4}$$

where $\mathbf{t}_v$ is the time embedding of v on edge e, $\text{K-Linear}_{\pi(v)}^i$ and $\text{Q-Linear}_{\pi(n)}^i$ are node-type-wise linear projections to generate the Key vector K_v^i and Query vector Q_n^i, respectively. By such type-wise projection operations, the node-level attention enables to handle of multiple types of nodes. Then the i-th attention head for $e = (n, v)$ is obtained:

$$\text{ATT-head}^i(n, e, v) = \left(K_v^i W_{\phi(e)}^{\text{ATT}} Q_n^{i\top}\right) \cdot \frac{\mu_{(\pi(n),\phi(e),\pi(v))}}{\sqrt{d}}. \tag{5}$$

Here, $W_{\phi(e)}^{\text{ATT}} \in \mathbb{R}^{\frac{d}{h} \times \frac{d}{h}}$ is a learnable edge-based matrix for each edge type $\phi(e)$, and $\mu \in \mathbb{R}^{|\mathcal{R}_\mathcal{V}| \times |\mathcal{R}_\mathcal{E}| \times |\mathcal{R}_\mathcal{V}|}$ represents a prior tensor signifying the general importance of individual relation $e = (n, v)$.

To incorporate the multimodal information of earnings calls, we further devise an audio-guided attention mechanism specific to aggregate the messages on P-S type edges to the target type node, i.e., the stock price node. According to E-Graph construction, the P-S type edges of a stock price node n are between n and all text sentence nodes v from one corresponding earnings call. In order to capture the audio clues of each earnings call, we leverage the audio feature of each text sentence to re-weight each text sentence node's attention. The i-th attention head of $e^{P-S}(n^P, v^S)$ is derived as follows:

$$K_{v^S}^i = \text{K-Linear}_{\pi(v^S)}^i \left(W_S^{\text{ATT}} \mathbf{h}_{v^S}^{(l-1)} + W_A^{\text{ATT}} \mathbf{a}_{v^S}\right),$$
$$Q_{v^S}^i = \text{Q-Linear}_{\pi(v^S)}^i \left(W_S^{\text{ATT}} \mathbf{h}_{v^S}^{(l-1)} + W_A^{\text{ATT}} \mathbf{a}_{v^S}\right), \tag{6}$$
$$\text{ATT-head}^i\left(n^P, e^{P-S}, v^S\right) = \left(K_{v^S}^i W_{\phi(e^{P-S})}^{\text{ATT}} Q_{v^S}^{i\top}\right) \frac{\mu_{(\pi(n^P),\phi(e^{P-S}),\pi(v^S))}}{\sqrt{d}},$$

where $\mathbf{a}_{v^S}$ is the audio feature of node v^S, W_S^{ATT} and W_A^{ATT} are two learnable matrices.

After the corresponding node type i-th attention head calculation, we concatenate h attention heads together to get the final attention vector for each node pair $e = (n, v)$:

$$\textbf{Attention}(n, e, v) = \underset{\forall v \in \mathcal{N}(n)}{\text{Softmax}} \left(\underset{i \in [1,h]}{\Vert} \text{ATT-head}^{\text{i}}(n, e, v) \right). \tag{7}$$

Then we aggregate information to target node n from all its neighbors:

$$\mathbf{z}_n^l = \oplus_{\forall v \in \mathcal{N}(n)} \left(\textbf{Attention}(n, e, v) \cdot \textbf{Message}(n, e, v)\right). \tag{8}$$

Finally, we map target node n's aggregation representation back to its type-specific distribution through a linear projection layer, followed by applying a residual connection (He et al. 2016):

$$\mathbf{h}_n^l = \text{A-Linear}_{\pi(n)} \left(\sigma\left(\mathbf{z}_n^l\right)\right) + \mathbf{h}_n^{(l-1)}. \tag{9}$$

Post Earnings Call Stock Dynamics Module

In the post earnings call stock dynamics module, `ECHO-GL` propagates the dynamics from the initial stock representation $\mathbf{h}_{s,t}$ on the earnings call's announcement time t to extrapolate the entire stock representation trajectory $\mathbf{h}_{s,t:t+w}$ within any length of prediction time horizon w.

For the target stock s, we take its stock price node representation $\mathbf{h}_s^L$ obtained from the stock spatial relational module as the initial stock representation $\mathbf{h}_{s,t} = \mathbf{h}_s^L$ on the earnings call announcement time t.

To model the latent PEAD effect on movements, we devise a stochastic process, which extends the Neural ODEs to incorporate intrinsic stochasticity in the stock dynamics. We specify the transformation of stock representations at time τ by a stochastic differential equation (SDE) (Li et al. 2020b):

$$d\mathbf{h}_{s,\tau} = \underbrace{f_\psi(\tau, \mathbf{h}_{s,\tau}) \cdot d\tau}_{\text{drift}} + \underbrace{\sigma_\phi(\tau, \mathbf{h}_{s,\tau}) \cdot dW_\tau}_{\text{diffusion}}, \tag{10}$$

where W_t is the Wiener process, and the two learnable functions f_ψ and σ_ϕ are dynamics drift and diffusion functions, respectively. We parameterize dynamics drift and diffusion functions by two multi-layer perceptrons (MLP).

Starting $\mathbf{h}_{s,t}$, the extrapolated stock representation at any specific time $t + w$ after the earnings call announcement is given by integrating an SDE forward by time, as follows:

$$\mathbf{h}_{s,t+w} = \mathbf{h}_{s,t} + \int_{\tau=t}^{t+w} \frac{d\mathbf{h}_{s,\tau}}{d\tau} d\tau. \tag{11}$$

Prediction and Optimization

For stock movement prediction over multiple time horizons, the stock representation $\mathbf{h}_{t+w}$ on each time horizon w is fed into a linear layer to generate the predicted movement label.

We optimize `ECHO-GL` using a Cross-Entropy loss as:

$$\mathcal{L} = \sum_{s \in \mathcal{Y}_L} \sum_{w \in \mathcal{W}} -[(1 - y_{s,t+w}) \log(1 - \hat{y}_{s,t+w}) + y_{s,t+w} \log(\hat{y}_{s,t+w})], \tag{12}$$

where $\mathcal{Y}_L$ is the set of stock price node indices that have labels, $y_{s,t+w}$ is the movement ground-truth default label of target stock s on the prediction day, $\hat{y}_{s,t+w}$ is the predicted movement label.

Experiments

In this section, we present extensive experiments to answer the following questions: **Q1:** How does `ECHO-GL` perform on predicting stock movements? **Q2:** How do the key components contribute to the performance of `ECHO-GL`? **Q3:** How about the real-world trading profitability of `ECHO-GL`?

Experimental Settings

Dataset Descriptions. We conduct extensive experiments on two real-world datasets, i.e., Qin's (Qin and Yang 2019) and MAEC (Li et al. 2020a) datasets, which contain both the text transcripts and audio records of earnings calls from S&P 500 and S&P 1500 companies in U.S. stock exchanges, respectively. We collect dividend-adjusted closing prices from Yahoo Finance[2]. Following previous studies (Qin and Yang 2019; Yang et al. 2020), we split the datasets into mutually exclusive training/validation/testing sets in the ratio of 7:1:2 in chronological order. The statistics of datasets are presented in Table 1. We provide the original dataset with constructed E-Graphs, code of `ECHO-GL`, and implementation details in our GitHub repository[3].

[2]https://finance.yahoo.com

[3]https://github.com/pupu0302/ECHOGL

Dataset	Qin's	MAEC
Earnings call features		
Start Date	2017-01-17	2015-02-25
End Date	2017-12-20	2018-06-21
Number of companies	253	963
Number of earnings calls	400	2,725
Average sentences per earnings call	157	115
Total number of tokens	1,593,360	7,601,758
Average tokens per sentence	25	24
Audio features dimensions	29	29
Graph features		
Number of entity nodes	1615	4504
Number of topic nodes	100	100
Number of sentence nodes	62,894	314,660
Number of edges	174, 095	1,412,356

Table 1: Dataset descriptions.

Comparison Methods. We compare ECHO-GL with sixteen baselines in three categories as follows:

(1) *Price-based methods:* **ARIMA** (Hamilton 1994) is a traditional statistics-oriented time series modeling approach. **LSTM** and **BiLSTM** (Yang and Wang 2022) are RNN-based methods for price series modeling. **Transformer**, **Informer**, **Autoformer**, and **Convtrans** are four transformer-based time series models for stock movement prediction.

(2) *Graph-based methods:* **RSR** (Feng et al. 2018) is a graph learning method based on stock industry relations. **VolTAGE** (Sawhney et al. 2020a) pass and aggregate earnings call information based on wiki-industry relations for stock prediction. **STHAN-SR** (Sawhney et al. 2021) is a state-of-the-art method learning on wiki-industry hypergraph for stock movement prediction. **HISN** (Wang et al. 2022) is a state-of-the-art method that builds stock-media interactive heterogeneous snapshots for stock prediction.

(3) *Earnings calls-based methods:* The following baselines leverage earnings calls for prediction. **MDRM** (Qin and Yang 2019) is the first work to study the impact of CEOs' vocal clues on stock predictions. **HTML** (Yang et al. 2020) and **NumHTML** (Yang et al. 2022) are two state-of-the-art transformer-based methods for financial multitask predictions. **Ensemble** (Sawhney et al. 2020b) is a state-of-the-art method that leverages text-audio attentive alignment for stock prediction. **STOCKGNN** (Medya et al. 2022) is a state-of-the-art GNN-based method exploiting semantic features of earnings calls for stock movement prediction.

Evaluation Metrics. Following previous studies (Yang et al. 2022; Sawhney et al. 2020b), we use the F1 score and Mathew's Correlation Coefficient (MCC) for stock movement prediction. Note that all metrics are measured for each prediction time horizon $w \in \{1, 3, 7, 15, 30\}$.

Comparison with Baselines (for Q1)

To answer **Q1**, we compare the performance of ECHO-GL and all baselines on stock movement prediction tasks for

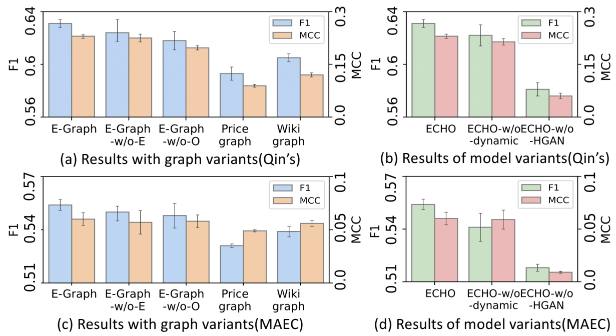

Figure 3: Ablation results for graph and model variants.

1, 3, 7, 15, and 30-day time horizons (see Table 2). Overall, ECHO-GL consistently achieves the best performance across diverse time horizon prediction tasks on the two datasets, with average improvements of 2.297% on F1 score and 15.629% on MCC over the best-performing baselines. Compared with baselines in three categories respectively, we draw the following three conclusions. Firstly, ECHO-GL outperforms price-based methods relying solely on historical prices, indicating that side information provides rich movement signals to enhance stock predictions. Secondly, compared with graph-based baselines, by exploiting rich information in earnings calls, ECHO-GL is capable of capturing more comprehensive spatial-temporal stock relations, resulting in better performance. Thirdly, ECHO-GL outperforms all earnings call-based baselines, which demonstrates its capability of not only modeling rich semantics within earnings calls but also effectively capturing meaningful movement signals between related stocks based on stock relations derived from earnings calls. Moreover, due to the existence of the PEAD phenomenon, traditional earnings call approaches suffer from an inability to effectively depict the time-evolving stock features affected by earnings calls, resulting in their poor performances.

Ablation Studies (for Q2)

To answer **Q2**, we conduct the following ablation experiments to analyze the contributions of earnings call-based heterogeneous graph and two key components of ECHO-GL.

Effect of Different Stock Relation Graphs. We evaluate the effectiveness of different graph construction approaches in modeling stock relations. As shown in Figure 3 (a) and (c), we consider ECHO-GL with E-Graph and four graph variants, i.e., the price graph, wiki graph, E-Graph-w/o-E, and E-Graph-w/o-O. Specifically, the price graph is constructed based on the stock price covariance matrix. The wiki graph is constructed by incorporating both sector-industry and wiki relations. The E-Graph-w/o-E and E-Graph-w/o-O are variants of E-Graph without entity and topic nodes, respectively.

From Figure 3 (a) and (c), we can draw the following two conclusions. On the one hand, the model employing earnings call-based heterogeneous graphs achieves significant improvements over price and wiki graphs, which indicates the superiority of dynamic stock relations derived from earn-

Stock Movement Prediction (***Qin's***)										
(%)	$F1_1$	$F1_3$	$F1_7$	$F1_{15}$	$F1_{30}$	MCC_1	MCC_3	MCC_7	MCC_{15}	MCC_{30}
Price-based Methods										
ARIMA	60.2 ± 0.0	55.7 ± 0.6	49.9 ± 1.0	49.6 ± 3.0	59.5 ± 7.9	0.0 ± 0.0	5.1 ± 7.0	0.2 ± 1.1	-10.7 ± 9.2	-5.5 ± 6.4
LSTM	60.2 ± 0.0	55.3 ± 0.0	49.3 ± 0.6	55.3 ± 0.0	64.9 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
BiLSTM	60.2 ± 0.0	55.4 ± 0.3	49.8 ± 0.9	55.5 ± 0.8	64.9 ± 0.0	0.0 ± 0.0	1.7 ± 5.2	0.0 ± 0.0	1.3 ± 4.1	0.0 ± 0.0
Transformer	60.4 ± 0.6	58.2 ± 1.7	54.2 ± 1.8	57.0 ± 1.6	65.3 ± 0.5	1.3 ± 4.1	14.1 ± 5.3	10.2 ± 3.9	9.8 ± 5.4	5.1 ± 6.6
Informer	60.7 ± 1.1	56.2 ± 4.8	54.2 ± 2.6	56.5 ± 3.4	65.0 ± 0.3	4.0 ± 6.9	9.0 ± 8.2	8.2 ± 6.2	8.5 ± 8.4	1.3 ± 4.0
Autoformer	61.1 ± 0.9	56.3 ± 1.6	55.8 ± 3.0	56.3 ± 1.8	65.0 ± 0.3	7.5 ± 6.7	8.5 ± 6.9	12.2 ± 5.6	10.3 ± 5.2	1.1 ± 3.4
Convtrans	60.2 ± 0.0	55.3 ± 0.0	52.0 ± 1.7	56.3 ± 1.4	64.9 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	3.1 ± 4.3	5.0 ± 6.4	0.0 ± 0.0
Graph-based Methods										
RSR	59.3 ± 2.2	58.3 ± 2.9	56.4 ± 2.5	55.6 ± 4.7	54.6 ± 2.4	12.5 ± 4.6	15.1 ± 5.9	13.8 ± 5.5	15.6 ± 9.3	13.0 ± 5.6
VolTAGE	57.6 ± 2.9	57.8 ± 3.1	56.9 ± 1.6	57.0 ± 1.7	61.7 ± 6.0	10.1 ± 8.4	14.2 ± 7.5	14.6 ± 3.7	11.2 ± 1.5	8.2 ± 12.7
STHAN-SR	62.8 ± 0.2	57.3 ± 1.5	52.2 ± 1.7	51.2 ± 2.9	44.9 ± 7.6	15.6 ± 9.0	11.6 ± 6.5	13.7 ± 7.8	17.7 ± 9.2	2.5 ± 8.2
HISN	62.0 ± 2.4	56.5 ± 1.2	58.8 ± 1.4	60.0 ± 3.1	61.7 ± 1.8	15.1 ± 5.5	8.1 ± 7.9	13.8 ± 3.5	12.5 ± 2.0	3.9 ± 0.8
Earnings Calls-based Methods										
MDRM	60.4 ± 0.6	56.9 ± 1.7	57.4 ± 4.3	60.4 ± 3.4	66.2 ± 1.1	2.7 ± 5.1	8.9 ± 6.2	15.3 ± 8.9	19.2 ± 8.4	12.5 ± 9.2
Ensemble	62.7 ± 1.7	59.6 ± 2.0	56.7 ± 0.6	61.8 ± 2.0	65.4 ± 1.1	16.0 ± 2.4	17.2 ± 5.1	15.1 ± 0.5	21.7 ± 1.3	3.7 ± 8.1
HTML	60.6 ± 1.0	58.5 ± 1.4	59.5 ± 1.5	60.2 ± 1.4	66.2 ± 2.5	13.9 ± 3.0	15.8 ± 3.0	19.9 ± 3.8	18.7 ± 3.2	19.2 ± 6.4
NumHTML	61.0 ± 0.9	58.7 ± 1.4	59.8 ± 1.5	60.5 ± 1.5	66.6 ± 2.5	13.9 ± 3.1	15.8 ± 3.0	20.0 ± 1.2	18.8 ± 3.2	19.3 ± 0.7
StockGNN	62.6 ± 2.1	58.7 ± 1.7	55.4 ± 2.1	60.0 ± 2.9	66.1 ± 1.0	14.9 ± 7.6	16.3 ± 4.2	12.2 ± 4.1	19.0 ± 5.9	13.5 ± 7.0
Ours	**63.5** ± 0.3	**63.5** ± 2.2	**61.6** ± 1.8	**63.3** ± 0.6	**67.1** ± 1.9	**20.2** ± 4.4	**30.7** ± 3.2	**21.3** ± 0.9	**22.9** ± 0.9	**19.8** ± 1.1
Improvement	+1.1	+6.5	+3.0	+2.4	+0.7	+26.3	+78.5	+6.5	+5.5	+2.6
(%) p-value	0.0	0.0	2.3	3.6	0.0	1.7	0.0	2.2	2.3	2.6

Stock Movement Prediction (***MAEC***)										
(%)	$F1_1$	$F1_3$	$F1_7$	$F1_{15}$	$F1_{30}$	MCC_1	MCC_3	MCC_7	MCC_{15}	MCC_{30}
Price-based Methods										
ARIMA	53.2 ± 0.7	52.2 ± 0.8	49.4 ± 0.7	48.7 ± 1.1	53.3 ± 1.0	4.6 ± 3.1	4.3 ± 1.7	-0.5 ± 1.7	-4.8 ± 2.4	-2.4 ± 2.1
LSTM	47.6 ± 0.0	51.4 ± 0.0	50.7 ± 0.0	53.0 ± 0.0	55.7 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
BiLSTM	47.6 ± 0.0	51.4 ± 0.0	50.7 ± 0.0	53.0 ± 0.0	55.7 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Transformer	49.8 ± 1.9	52.1 ± 1.1	50.6 ± 0.3	53.0 ± 0.0	55.5 ± 0.0	1.3 ± 3.5	2.9 ± 4.0	0.3 ± 0.9	0.7 ± 0.7	0.0 ± 0.0
Informer	51.9 ± 1.9	51.3 ± 1.1	50.5 ± 0.3	51.2 ± 0.0	48.9 ± 0.0	0.0 ± 3.5	0.7 ± 4.0	0.0 ± 0.9	0.0 ± 0.7	0.0 ± 0.0
Autoformer	50.0 ± 2.5	52.2 ± 1.1	51.7 ± 1.3	52.7 ± 2.1	54.4 ± 3.5	2.4 ± 2.8	3.8 ± 2.3	2.7 ± 3.2	1.7 ± 3.1	0.7 ± 3.2
Convtrans	53.1 ± 1.2	50.6 ± 1.8	50.5 ± 1.0	53.0 ± 0.0	54.4 ± 3.5	2.8 ± 4.6	1.7 ± 2.2	1.1 ± 2.3	0.0 ± 0.0	0.0 ± 0.0
Graph-based Methods										
RSR	53.6 ± 0.9	53.0 ± 1.1	52.7 ± 1.2	53.8 ± 0.9	55.8 ± 0.7	6.5 ± 0.5	6.1 ± 3.1	5.6 ± 0.4	5.5 ± 3.7	2.8 ± 4.2
VolTAGE	53.2 ± 1.3	50.3 ± 0.8	49.6 ± 0.8	50.1 ± 0.4	56.2 ± 0.3	5.1 ± 2.3	1.4 ± 1.6	0.9 ± 1.6	1.6 ± 1.1	5.9 ± 2.1
STHAN-SR	53.5 ± 1.3	49.8 ± 1.2	50.6 ± 0.8	52.3 ± 1.9	55.5 ± 0.1	4.4 ± 4.3	2.9 ± 4.0	2.9 ± 2.1	0.3 ± 2.7	3.1 ± 1.4
HISN	52.0 ± 2.0	51.5 ± 1.2	50.8 ± 1.4	53.0 ± 1.1	56.7 ± 0.8	2.3 ± 0.5	0.1 ± 0.4	0.0 ± 0.5	0.7 ± 0.6	4.2 ± 0.8
Earnings Calls-based Methods										
MDRM	48.7 ± 1.6	52.0 ± 0.5	51.0 ± 0.5	53.7 ± 1.1	55.9 ± 0.2	1.9 ± 2.2	2.8 ± 1.8	2.7 ± 2.4	2.8 ± 4.5	6.3 ± 0.5
Ensemble	52.1 ± 2.0	52.2 ± 0.9	51.1 ± 0.7	53.8 ± 0.4	55.9 ± 0.6	2.3 ± 3.3	2.8 ± 2.9	2.4 ± 3.1	4.9 ± 1.8	6.0 ± 7.8
HTML	53.1 ± 1.5	52.2 ± 0.9	52.7 ± 2.2	54.1 ± 1.0	56.0 ± 0.5	6.4 ± 2.9	4.0 ± 1.8	5.6 ± 4.5	5.8 ± 2.6	5.5 ± 2.0
NumHTML	53.4 ± 1.5	52.4 ± 1.0	53.0 ± 1.6	54.0 ± 1.2	56.4 ± 0.6	6.4 ± 2.9	4.0 ± 1.8	5.6 ± 4.5	5.8 ± 2.6	5.6 ± 2.0
StockGNN	53.0 ± 0.3	51.1 ± 1.7	52.0 ± 1.0	52.6 ± 1.9	55.2 ± 1.2	4.0 ± 1.6	4.3 ± 2.9	5.0 ± 2.3	6.4 ± 0.3	6.0 ± 2.4
Ours	**54.4** ± 0.7	**54.3** ± 1.4	**54.4** ± 0.7	**54.9** ± 0.4	**57.3** ± 0.4	**6.7** ± 0.2	**6.3** ± 0.3	**6.0** ± 0.3	**6.7** ± 0.3	**6.9** ± 0.7
Improvement	+1.5	+2.5	+2.6	+1.5	+1.1	+3.1	+12.5	+7.1	+4.7	+9.5
(%) p-value	3.5	3.2	1.2	3.5	0.0	3.2	0.2	3.7	2.1	3.7

Table 2: Results of all methods on two metrics (mean ± std, computed across 10 runs). Improvements are calculated as the performance differentials between our model `ECHO-GL` over the best baselines (underscored for emphasis). The improvement is significant based on paired t-test at the significance level of 5% (p-value with paired t-test). Note that all F1, MCC, improvement, and p-value scores in the table are percentage values (e.g., 60.2 ± 0.0 means 60.2% ± 0.0%, and 1.1 means 1.1%).

ings calls in enhancing stock movement prediction. On the other hand, the absence of either topic or entity nodes weakens the model's performance, which demonstrates that the E-Graph encompasses multiple heterogeneous information, comprehensively reflecting dynamic stock relations.

Effect of Model Variants. We conduct ablation analysis on two model variants, i.e., `ECHO`-w/o-HGAN and `ECHO`-w/o-dynamic, which corresponds to `ECHO-GL` without the stock spatial relation module and post earnings call stock dynamics module, respectively. As shown in Figure 3 (b) and (d), `ECHO-GL` significantly outperforms the two model variants, which demonstrates that aggregating rich relational

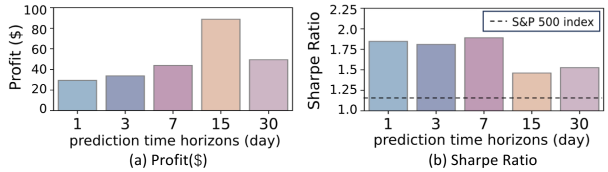

Figure 4: Trading simulation on various time horizons.

information from the E-Graph is crucial to our model, and post earnings call stock dynamics module effectively models stock price trajectories, enhancing the model's predictive ability.

Trading Simulation

To answer **Q3**, we conducted real-world trading simulations from 24th Oct 2017 to 2nd Feb 2018, employing a w-day strategy based on the stock movement prediction results of ECHO-GL. The w-day strategy decides to buy the stock on day t if it predicts a rise from day t to $t+w$ and then sells it on day $t+w$; otherwise, short sell the stock and hold the short position until day $t+w$. We take the S&P 500 index as a benchmark and conduct five w-day strategies with w in $[1, 3, 7, 15, 30]$, respectively. The simulated trading profits ($) and annual Sharpe Ratio (SR) (Sharpe 1998) are shown in Figure 4. Results illustrate that trading strategies based on ECHO-GL prediction results can effectively help investors obtain positive profits on all time horizons. Moreover, the annual SR achieved by ECHO-GL significantly exceeds that of the S&P 500 index, demonstrating ECHO-GL's stable and reliable profitability in the ever-changing stock market.

Discussion and Conclusion

Ethical Considerations. Previous studies (Sawhney et al. 2020b; Li et al. 2020a) have acknowledged that biases exist in earnings calls, e.g., gender and demographic biases. Such modeling biases can result in real-world harm if deployed without care. In order for AI systems to be useful to society as a whole, they must perform equally well for all populations regardless of perceived skin tone or gender. In future work, we plan to extend our study to address these biases.

Conclusion. In this paper, we propose a novel ECHO-GL model, which not only captures complex movement signals between related stocks on the earnings calls-driven dynamic heterogeneous graph, but also builds learnable stochastic processes tailored to generate stock representation trajectories over various prediction horizons. Extensive experiments demonstrate ECHO-GL's superiority in stock movement prediction and trading profitability over multiple time horizons. Considering the effectiveness of E-Graph, we intend to further exploit more potential of E-Graph in practical applications for different investment demands in future work.

Acknowledgments

This work was supported in part by the National Key R&D Program of China (No. 2022YFF0902704), the Natural Science Foundation of Zhejiang Province (LQ23F020007), Zhejiang Gongshang University "Digital+" Disciplinary Construction Management Project (SZJ2022B001), and Lingjun Investment.

References

Araci, D. 2019. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. *arXiv preprint arXiv:1908.10063*, 1–11.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan): 993–1022.

Fan, J.; Han, F.; and Liu, H. 2014. Challenges of big data analysis. *National science review*, 1(2): 293–314.

Feng, F.; He, X.; Wang, X.; Luo, C.; Liu, Y.; and Chua, T.-S. 2018. Temporal Relational Ranking for Stock Prediction. *ACM Transactions on Information Systems (TOIS)*, 37: 1 – 30.

Hamilton, J. D. 1994. Time Series Analysis. *Princeton University Press*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proc. of the IEEE*, 770–778.

Hu, L.; Yang, T.; Shi, C.; Ji, H.; and Li, X. 2019. Heterogeneous Graph Attention Networks for Semi-supervised Short Text Classification. *ACM Transactions on Information Systems (TOIS)*, 39: 1 – 29.

Ke, Y.; Lian, H.; and Zhang, W. 2020. High-Dimensional Dynamic Covariance Matrices With Homogeneous Structure. *Journal of Business & Economic Statistics*, 40: 96 – 110.

Keith, K. A.; and Stent, A. 2019. Modeling financial analysts' decision making via the pragmatics and semantics of earnings calls. *arXiv preprint arXiv:1906.02868*, 1–12.

Li, J.; Yang, L.; Smyth, B.; and Dong, R. 2020a. Maec: A Multimodal Aligned Earnings Conference Call Dataset for Financial Risk Prediction. In *Proc. of CIKM*, 3063–3070.

Li, W.; Bao, R.; Harimoto, K.; Chen, D.; Xu, J.; and Su, Q. 2021. Modeling the Stock Relation with Graph Network for Overnight Stock Movement Prediction. In *Proc. of IJCAI*, 4541–4547.

Li, X.; Wong, T.-K. L.; Chen, R. T.; and Duvenaud, D. 2020b. Scalable Gradients for Stochastic Differential Equations. In *Proc. of AISTATE*, 3870–3882. PMLR.

Li, Y.; Tarlow, D.; Brockschmidt, M.; and Zemel, R. S. 2015. Gated Graph Sequence Neural Networks. *CoRR*, abs/1511.05493.

Medya, S.; Rasoolinejad, M.; Yang, Y.; and Uzzi, B. 2022. An exploratory study of stock price movements from earnings calls. In *Proc. of WWW*, 20–31.

Qin, Y.; and Yang, Y. 2019. What You Say and How You Say It Matters: Predicting Stock Volatility Using Verbal and Vocal Cues. In *Proc. of ACL*, 390–401.

Sak, H.; Senior, A. W.; and Beaufays, F. 2014. Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition. *arXiv preprint arXiv:1402.1128*, abs/1402.1128: 1–5.

Sawhney, R.; Agarwal, S.; Wadhwa, A.; Derr, T.; and Shah, R. R. 2021. Stock Selection via Spatiotemporal Hypergraph Attention Network: A Learning to Rank Approach. In *Proc. of AAAI*, 497–504.

Sawhney, R.; Khanna, P.; Aggarwal, A.; Jain, T.; Mathur, P.; and Shah, R. R. 2020a. VolTAGE: Volatility Forecasting via Text-Audio Fusion with Graph Convolution Networks for Earnings Calls. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, 8001–8013.

Sawhney, R.; Mathur, P.; Mangal, A.; Khanna, P.; Shah, R. R.; and Zimmermann, R. 2020b. Multimodal multi-task financial risk forecasting. In *Proc. of ACM Multimedia*, 456–465.

Sharpe, W. F. 1998. The Sharpe Ratio. *Streetwise–the Best of the Journal of Portfolio Management*, 3: 169–185.

Wang, H.; Wang, T.; Li, S.; Guan, S.; Zheng, J.; and Chen, W. 2022. Heterogeneous Interactive Snapshot Network for Review-Enhanced Stock Profiling and Recommendation. In *Proc. of IJCAI*, 3962–3969.

Wang, Y.; Chang, Y.-Y.; Liu, Y.; Leskovec, J.; and Li, P. 2020. Inductive Representation Learning in Temporal Networks via Causal Anonymous Walks. In *ICLR*, 1–13.

Yang, L.; Li, J.; Dong, R.; Zhang, Y.; and Smyth, B. 2022. NumHTML: Numeric-Oriented Hierarchical Transformer Model for Multi-task Financial Forecasting. In *Proc. of AAAI*, 11604–11612.

Yang, L.; Ng, T. L. J.; Smyth, B.; and Dong, R. 2020. Html: Hierarchical transformer-based multi-task learning for volatility prediction. In *Proc. of WWW*, 441–451.

Yang, M.; and Wang, J. 2022. Adaptability of financial time series prediction based on BiLSTM. *Procedia Computer Science*, 199: 18–25.

Zhu, M.; Wang, Y.; Wu, F.; Yang, M.; Chen, C.; Liang, Q.; and Zheng, X. 2022. WISE: Wavelet based Interpretable Stock Embedding for Risk-Averse Portfolio Management. In *Proc. of WWW*, 1–11.