Adaptive Long-Short Pattern Transformer for Stock Investment Selection

Heyuan Wang\textsuperscript{1,3}, Tengjiao Wang\textsuperscript{1,3}, Shun Li\textsuperscript{2*}, Jiayi Zheng\textsuperscript{1,3}, Shijie Guan\textsuperscript{1,3} and Wei Chen\textsuperscript{1,3}

\textsuperscript{1}School of Computer Science, National Engineering Laboratory for Big Data Analysis and Applications, Peking University, China

\textsuperscript{2}University of International Relations

\textsuperscript{3}Institute of Computational Social Science, Peking University (Qingdao)

\{wangheyuan, tjwang, jiayizheng, guanshijie, pekingchenwei\}@pku.edu.cn, lishunmail@foxmail.com

Abstract

Stock investment selection is a hard issue in the Fintech field due to non-stationary dynamics and complex market interdependencies. Existing studies are mostly based on RNNs, which struggle to capture interactive information among fine granular volatility patterns. Besides, they either treat stocks as isolated, or presuppose a fixed graph structure heavily relying on prior domain knowledge. In this paper, we propose a novel Adaptive Long-Short Pattern Transformer (ALSP-TF) for stock ranking in terms of expected returns. Specifically, we overcome the limitations of canonical self-attention including context and position agnostic, with two additional capacities: (i) fine-grained pattern distiller to contextualize queries and keys based on localized feature scales, and (ii) time-adaptive modulator to let the dependency modeling among pattern pairs sensitive to different time intervals. Attention heads in stacked layers gradually harvest short- and long-term transition traits, spontaneously boosting the diversity of stock representations. Moreover, we devise a graph self-supervised regularization, which helps automatically assimilate the collective synergy of stocks and improve the generalization ability of overall model. Experiments on three exchange market datasets show ALSP-TF’s superiority over state-of-the-art stock forecast methods.

1 Introduction

As an essential ingredient of modern financial ecosystem, the forecast of stock market has aroused extensive research interest due to scientific and investment merits [Feng et al., 2019b]. Different from the general time series modeling, it is inherently difficult to assess stock’s evolving trend confronting with highly volatile and interrelated natures of the market. Traditional literatures leverage machine learning algorithms based on manual indicators [Nayak et al., 2015; Khaidem et al., 2016], where the hypothetical stochastic process may become stranded in catching non-stationary oscillations. In recent studies, deep neural networks have shown encouraging prospects in characterizing stock dynamics, especially the RNNs are the dominant choice. For instance, Zhang et al. [2017] used state frequency memory in the LSTM network to decompose stock transaction patterns. Sawhney et al. [2021] equipped LSTM with temporal attention to reward driving hidden signals. In view of the short of RNNs in capturing granular feature units, Wang et al. [2021] proposed to apply gated causal CNNs to convolve price subsequences. However, both of them are ineffective to learn the dependencies of long-range discontinuous temporal states.

As an alternative, the well-known Transformer [Vaswani et al., 2017] preliminarily designed in the NLP field has gained huge success in learning sequential data [Ding et al., 2020; Zhou et al., 2021]. The core of Transformer is multi-head self-attention, which explicitly performs information exchange between input tokens to fix the deficiencies of RNN and CNN structures. However, directly applying the canonical encoder to modeling stock movements is problematic in two non-neglectable aspects: (1) The global self-attention focuses on point-wise token similarities without contextual insights [Xu et al., 2020]. Since stock fluctuations are conditioned on composite signals over manifold time spans, the lack of pattern-wise interaction hinders the adequate discrimination of stock tendency and is susceptible to noise points. (2) The basic query-key matching paradigm is position agnostic. Though sinusoidal position embedding is inserted to the sequential input, it may not be optimal due to the inability to reveal precise distances [Wu et al., 2021]. As an empirical example, in Fig. 1, the subsequences \{A, B\} and \{C, D\} reflect stock wave patterns in the context of different long- and short-term spans; Intuitively, conducting multi-granular matching between them instead of simple dot-point projec-
tions is more conducive to characterizing the evolution status. Additionally, the patterns $C$ and $D$ are closer than $A$ and $B$, which means that their interaction is more responsible for observing high-frequency regularities. This inspires us to factor in the elapsed time between stock change patterns so as to modulate the self-attention on stock trading series.

The synergy effect is another prominent trait of stock market, i.e. related stocks are apt to exhibit synchronous changes, offering a desiderative pointcut for trend predictions. Nevertheless, it is non-trivial to anchor full-scale stock interdependencies given that relationship sources may originate from various aspects (industry rotation, common shareholder, supply chain, etc). Previous works mostly treat individual stocks as isolated [Wang et al., 2020; Ding et al., 2020]. Some studies presuppose a graph structure by resorting to limited domain knowledge [Chen et al., 2018; Feng et al., 2019b; Sawhney et al., 2021], which may lead to information bias in revealing intricate market factors. More particularly, task-specific graph building requires massive expertise, which prevents the model from being applied to extended scenarios.

Along these lines, in this paper, we present Adaptive Long-Short Pattern Transformer (ALSP-TF) for stock investment selection. The model is structurally innovated for hierarchical representation and interaction of stock price series at different context scales. In addition, with the help of a learnable sigmoid function, we make the self-attention aware of the weighted time intervals between patterns, in order to adaptively adjust their dependencies beyond similarity matching. Further, to get rid of reliance on any prior knowledge, we devise a graph self-supervised regularization which automatically learns stock topology through dynamic path alignment, and thereby boosting the generalization capacity of the overall model. Our major contributions are as follows:

- We delve into the issues of basic Transformer in modeling stock price series, and present a reformed self-attention encoder to exploit adaptive pattern-wise interactions supported by temporal representations at different grain levels.
- We construct a data-driven adjacency graph to uncover the implicit similarities in volatility across different stocks. It helps reduce the stochasticity of stock input sequences and serves as a self-supervision signal to guide the model’s representation learning.
- All the components are seamlessly integrated and jointly trained to predict stock expected returns. Experiments on datasets from NYSE, NASDAQ and TSE markets show the effectiveness of proposed model for investment selection.

2 Related Work

Technical Analysis. TA is at the heart of stock trend prediction, which is developed on top of price-volume indicators from historical quote data. Traditional mathematical algorithms such as HMM, SVM [Kavitha et al., 2013; Nayak et al., 2015; Khaidem et al., 2016] leverage manual feature engineering and hypothetical stochastic processes. Later studies move to exploiting deep neural networks to model the hidden dependency of stock dynamics, where RNNs are commonly utilized [Qin et al., 2017; Zhang et al., 2017; Wang et al., 2020]. To enhance the capacity of handling fine-granular transition signals, some efforts have explored other architectures such as hybrid SAE-LSTM units [Bao et al., 2017], adversarial training [Feng et al., 2019a], and gated causal convolutions [Wang et al., 2021]. These approaches have made progress in several stock prediction tasks, whereas they are commonly framed for the regression of prices or classification of bucketing stock movements. The latest work [Sawhney et al., 2021] claimed that the absence of optimization toward expected returns will harm practical investment choices. They reformulated stock forecasts as a learning to rank task and realized state-of-the-art profitability.

Market Relation Modeling. A new line of studies revolve around employing the collective synergy among stocks based on their metadata relevance. For instance, Lai et al. [2017] acquired stock relatedness by querying company collaboration and competition from the search engine, then made inferences based on unary and binary potentials in Markov random fields. Chen et al. [2018] built a graph of corporations based on their shareholding properties, and transferred stock prediction to node classification using graph convolutional network (GCN) [Kipf and Welling, 2017]. Feng et al. [2019b] clustered stocks from the same industries and supply chains to make the temporal price encoder aware of inter-stock relations. Sawhney et al. [2021] augmented the corporate relevance based on Wikidata and used hypergraph convolution to propagate higher-order neighbor’s information. Despite advances in graph-based stock forecasts, the preset knowledge-based stationary graph structure requires extensive domain expertise and may strain model’s extensibility.

Transformer. A powerful attention neural model that is preliminarily designed for machine translation [Vaswani et al., 2017] and now has attained huge success in various fields such as computer vision, multimodal reasoning and video classification [Gao et al., 2019; Dosovitskiy et al., 2021]. There are several recent studies in applying Transformer to time series modeling. Zhou et al. [2021] devoted to optimizing the efficiency of time complexity and memory usage of Transformer on extremely long time series. Ding et al. [2020] first tried to exploit Transformer on trading sequences to classify stock price movements. Differently, we delve and revamp the deficiencies of basic Transformer in grasping important context and temporal information of stock volatility patterns.

3 Methodology

3.1 Problem Formulation

To avoid the gap between stock movement prediction and investment profit, we follow the setup of [Sawhney et al., 2021] and adopt a learning to rank formulation for stock selection. Given the candidate set $S = \{s_1, \ldots, s_N\}$ of $N$ stocks, on any trading day $t$, each stock $s_i$ entails a feature sequence $X_i = [x_{t-\Delta T}, \ldots, x_{t-1}] \in \mathbb{R}^{\Delta T \times F}$ of historical $\Delta T$ time steps (where $F$ is the feature dimension), an associated closing price $p_i$ and a 1-day return ratio $r_i^t = \frac{p_i^t - p_i^{t-1}}{p_i^{t-1}}$. The model $F_\theta(X_{1:N})$ aims to output an ordering of all stocks $\mathcal{Y}_t = \{y_1^t > y_2^t \ldots > y_N^t\}$, where the top-ranked ones are expected to gain more investment revenues on day $t$. 

3971
3.2 Framework Overview

Fig. 2 illustrates the overview of ALSP-TF. It consists of three major parts: (i) **Graph-based Initialization** to shift the dynamic path alignment of stock pairs into a topology structure, and apply a gated graph smoothing operation to enrich the initial embeddings of stock time series; (ii) **Reformed Transformer Encoder** with hierarchical blocks of multi-head self-attention to exploit interdependencies between different stock volatility patterns. Each block jointly leverages fine-grained pattern distiller and time-adaptive modulator to calculate the attention scores on the basis of contextualized feature units; (iii) At last, regularized graph self-supervision signals are added to tune the entire stock ranking framework.

3.3 Time Series Embedding Initialization

**Stock Graph Construction**

The future movement of a stock is conditioned on its historical dynamic patterns as well as its neighbors’ synchronous information. To incorporate inter-stock dependencies, recent research relies to a large extent on prior domain knowledge, which can be labor-intensive, difficult to adequately reveal intricate market factors, and may narrow the applicability to new prediction tasks. In this regard, we propose to learn the implicit collective synergy among all candidate stocks in a data-driven manner. Specifically, we utilize a proximity function by applying multi-dimensional Dynamic Time Warping (DTW) [Jeong et al., 2011] on the input signals $X_{[1:N]}$. It finds the minimum alignment cost of each pair of stock sequences through dynamic programming on a path matrix $D$:

$$cost(s_i, s_j) \leftarrow DTW(D_{pq})_{\Delta T \times \Delta T}.$$  

Here we set each pixel $D_{pq} = \sum_{f=1}^{F}(X_{ij} - X_{ij})^2$. Then, we attach an edge between the stock pairs whose cost value is less than a limit threshold (by sorting alignment costs and controlling the graph sparsity $\rho$) to build the stock graph $G$.

**Graph-based Sequence Smoothing**

To further reduce the stochasticity of stock sequences, we develop a graph-based smoothing operation which substantiates stock proximity correlations using neighboring features. Let $\hat{A} \in \mathbb{R}^{N \times N}$ denote $G$’s normalized adjacency matrix with self-loops and $e^0_{i;h} \in \mathbb{R}^{(N \times \Delta T)^F}$ the input signals of $X_{[1:N]}$, we utilize the aggregation layer of graph attention (GAT) network [Velickovic et al., 2017] to convolve the $k$-order neighbor messages of stock $s_i$ into embedding of $a^k_i$.

Then we engage a GRU-like gate manner to rectify the fusing proportion in node updating process ($e^k_i = Gate(e^{k-1}_i, a^k_i)$):

$$\begin{align*}
    z^k_i &= \sigma(W_z a^k_i + U_z e^{k-1}_i + b_z), \\
    r^k_i &= \sigma(W_r a^k_i + U_r e^{k-1}_i + b_r), \\
    \tilde{e}^k_i &= \tanh(W_h a^k_i + U_h (r^k_i \cdot e^{k-1}_i) + b_h), \\
    e^k_i &= z^k_i \cdot \tilde{e}^k_i + e^{k-1}_i \cdot (1 - z^k_i),
\end{align*}$$

where $\odot$ is Hadamard product, $W_*, U_*, b_*$ are trainable weights and biases. $r^k_i$ and $z^k_i$ are reset and update gates, which are responsible for catching irrelevant information to forget and the part of past state to move forward, respectively.

3.4 Transformer Encoder

Next, we describe our reformed Transformer encoder to adaptively capture the interactive information between short- and long-term volatility patterns with different time intervals. The representation hierarchy consists of $L$ blocks of multi-head self-attention and feed-forward layers. Taken initialized stock embedding sequences $E \in \mathbb{R}^{N \times \Delta T \times d}$ as input, the canonical self-attention in [Vaswani et al., 2017] performs information exchange between every pair of time points for each stock $s_i$. Specifically, it transforms $E_i \in \mathbb{R}^{\Delta T \times d}$ (the series of $s_i$) into query $Q_{i,h} = E_i W_Q^h$, key $K_{i,h} = E_i W_K^h$ and value matrices $V_{i,h} = E_i W_v^h$ with distinct linear projection parameters, where $h = 1, \ldots, H$ is the head index, and
\( W^Q, W^K, W^V \in \mathbb{R}^{d \times d_f} \). Then scaled dot-product attention scores are computed to acquire a weighted sum of the sequential values. Afterwards, the final layer output is represented by the concatenation of all attention heads:

\[
F_i = \text{Multihead}(Q_{i,h}, K_{i,h}, V_{i,h}) = \|_{h=1}^{H} \text{Softmax}(Q_{i,h}K_{i,h}^T/\sqrt{d_f})V_{i,h},
\]

where \( h \) is the concatenator operator and \( d_f \) is the dimension of projected feature space. Thereby all stocks' representations are formed as \( F = [F_1; F_2; \ldots; F_N] \in \mathbb{R}^{N \times \Delta T \times H d_f} \) followed by feed-forward layers.

**Enhancing Locality with Fine-grained Pattern Distiller**

The dot-product attention calculated on top of point-wise tokens exhibits a powerful ability in extracting global dependencies for words in NLP and regions in CV. Whereas, the compound patterns implied in different short- and long-term local time spans are both important to shed light on the intricate stock market dynamics, while cannot be well exploited in such scheme. To solve it, in each block, we inject a local interaction (Lit) layer before global matching which that composite signals are shifted into new contextualized query-key tuples. To save computational consumptions, we borrow the concept of dilated causal convolutions in WaveNet [Oord et al., 2016], and skip interval “holes” on cascading Lit layers instead of making projection over contiguous subsequences in \( E_t \) to obtain wider receptive fields hierarchically.

Regarding Lit at the \( l.th \) block, at any time-step \( \tau \) we apply self-attention to process its surrounding \( \tau \)-length context, i.e., \( E^\tau_{i,h} = [E_{i-\tau}^\tau, \ldots, E_{i}^\tau] \in \mathbb{R}^{\tau \times d} \) (we use padding when \( \tau \leq w \)). Note that the context is a predecessor time period from \( \tau \), which is like a position mask to filter out temporal attention to future information. Therefrom we represent transformed key-value tuples as \( K_{i,h}^\tau = V_{i,h}^\tau = E^\tau_{i,h}W_Q^\tau \in \mathbb{R}^{\tau \times d_f} \), and form the state of current time-step as a query matrix \( Q_{i,h}^\tau = E^\tau_{i,h}W_Q^\tau \in \mathbb{R}^{1 \times d_f} \). After that, we exploit the context dependency and attentively sum up localized elements as a specific compound pattern for the moment \( \tau \):

\[
P_i^{1\rightarrow \tau} = \text{Multihead}(Q_{i,h}^\tau, K_{i,h}^\tau, V_{i,h}^\tau).
\]

By concatenating all time-steps, the embedding output of this layer is denoted as \( P_i^{1} \in \mathbb{R}^{\Delta T \times H d_f} \). Further, in the higher \((l+1)_th\) block, we pass \( P_i^{1} \) as input to Lit and \( \delta_{l+1} \) a wider skipping distance to handle longer-term local contexts \( E_{i+1}^{\tau_{l+1} \rightarrow \tau} = \|_{\mu=0}^{\mu=\tau_{l+1}} P_i^{1 \rightarrow (\tau-\mu)_{l+1}} \in \mathbb{R}^{\tau \times H d_f} \) along time-steps. In this way, the local receptive fields in the stacking Lits are exponentially expanded, supported by linearly growing parameters (e.g., we can obtain hierarchical patterns over 3→7→15 days by setting the draw-out size as \( w = 3 \) and skipping distances as 1→2→4). The \((l+1)\)-level compound patterns \( P_i^{l+1} \) are derived in the manner of Eq. 4 with different parameters \( W_{Q^{l}}^\tau \). Starting with \( \delta_1 = 1 \) to ensure no loss of coverage on the sequence, we devise \( \delta_{2:l} \) according to the performance in validation to gradually distill \( L \) different granularities of transition patterns for all stock sequences.

**Pattern Interaction with Time-adaptive Modulator**

Building on the fine-grained patterns distilled from Lit at each block, we turn to capturing global intra dependencies across the entire sequence. To this end, the position information of time series plays a critical role in measuring the variant of elapsed time between patterns. The canonical Transformer adds sinusoidal relative position embeddings to the input, which is proved weak due to loss of precise distance information. Let \( \mathbf{Z}_{\tau, \mu} = [\tau - \mu] \) denote the distance (i.e, temporal intervals) between \( \tau_{l+1} \) and \( \mu_{l+1} \) moments, we construct a matrix \( \mathbf{Z} \in \mathbb{R}^{\Delta T \times \Delta T} \) to indicate temporal distance signals between every pair of patterns in the embedding sequence \( P_i^{l} (i \in \{1, \ldots, N\}, \ell \in \{1, \ldots, L\}) \). Motivated by [Wu et al., 2021], we use a learnable sigmoid function \( f(\cdot) \) to rescale raw temporal values into an appropriate range, which is bounded, tunable and monotone to guarantee the model training:

\[
\mathbf{Z}_h = f(\mathbf{Z} ; a_h, v_h) = \frac{1 + \exp(v_h)}{1 + \exp(a_h - \mathbf{Z})},
\]

where \( a_h \) is a weight parameter that determines whether preferring to capture long-distance interactions (with a positive value) or focusing on short-distance dependencies (with a negative value). \( v_h \) controls the curve’s upperbound and ascending steepness, whose larger value means intenser effect of the time distance between patterns. Both \( a_h \) and \( v_h \) are tailored for the \( h_{th} \) global attention head in a learnable way.

Afterwards, we use the time-adaptive coefficients to adjust the attention operation inside Eq. 3. At each level of granularity \( l \), we linearly transform the local pattern sequence \( P_i^{l} \) and compute Multihead\((Q_{l,i,h}^{cross}, K_{l,i,h}^{cross}, V_{l,i,h}^{cross})\) as follows:

\[
F_{l,i} = \|_{h=1}^{H} \text{Softmax}(\text{ReLU}(Q_{l,i,h}^{cross}K_{l,i,h}^{cross}V_{l,i,h}^{cross})^{\mathbf{Z}_h})V_{l,i,h}^{cross},
\]

where \( Q_{l,i,h}^{cross} = P_i^{l}W_Q^Q, K_{l,i,h}^{cross} = P_i^{l}W_K^Q, V_{l,i,h}^{cross} = P_i^{l}W_V^Q \). * means element-wise product, ReLU activation is applied to keep non-negativity and sharpen the original attention weights. Hence, the feature similarity and time distance are jointly measured to make the volatility patterns in the \( l_{th} \) block crossly attend to each other. We keep one feed-forward network and residual connection of canonical Transformer to further process \( F_{l,i} \). The transformation matrices are shared for all stocks. Finally, by averaging all steps of each layer and then concatenating multi-granular embeddings, we represent the stock set \( \mathcal{S} \) as a compact tensor \( \mathcal{O} \in \mathbb{R}^{N \times (L \times H d_f)} \).

### 3.5 Prediction and Network Optimization

**Rank Loss.** For ranking optimization, we acquire stock predicted return ratios on day \( t \) by feeding stock representations \( \mathcal{O} \) to a dense layer with the activation of Leaky-ReLU. Then we jointly compute the point-wise regression and pairwise ranking loss with a weighting coefficient \( \alpha \), to minimize the discrepancy between predicted \( r_i^t \) and ground-truth \( r_i^t \) meanwhile maintaining the relative order of stocks:

\[
\mathcal{L}_R = \sum_{i=1}^{N} \|r_i^t - r_i^t\| + \alpha \sum_{i=1}^{N} \max \{0, -(r_i^t - r_j^t)(r_i^t - r_j^t)\}.
\]
Graph Proximity Loss. To ensure that the learned stock embeddings can effectively capture the correlation information stored in the alignment graph structure $\mathcal{G}$, we further introduce a graph reconstruction strategy, which regulates stock representations by explicitly drawing closer the node neighbors (we denote $N_i$ as the 1-hop neighbors of $s_i$ including itself) and pushing farther the negative ones in feature space:

$$
\mathcal{L}_{GP}(i) = - \sum_{j \in N(i)} \log(\sigma(o_i o_j^T)) - \sum_{p \in S - N_i} \log(\sigma(-o_i o_p^T)),
$$

where $o_i$, $o_j$, $o_p$ denote the embeddings of stock node $i$ and its neighbor $j$ in a pair, as well as one sampled negative node.

Combining the supervisory ranking signals and the self-supervised graph proximity loss, we reach the complete end-to-end loss function with a weighting coefficient $\eta$:

$$
\mathcal{L} = \mathcal{L}_R + \eta \mathcal{L}_{GP}.
$$

### 4 Experiments

#### 4.1 Experimental Setup

**Datasets.** We examine ALSP-TF on three real-world datasets from US and Japanese Exchange markets. Table 2 shows the detailed statistics. The first dataset [Feng et al., 2019b] comprises 1,026 shares from fairly volatile US S&P 500 and NASDAQ Composite Indexes; The second dataset [Feng et al., 2019b] targets at 1,737 stocks from NYSE, which is by far the world’s largest stock exchange w.r.t. market capitalization of listed companies and is relatively stable compared to NASDAQ; The third dataset [Li et al., 2021] corresponds to the popular TOPIX-100 Index, which includes 95 stocks with the largest market capitalization in Tokyo stock exchange.

**Implementation Details.** Our model is implemented with PyTorch. We collect daily quote data of all stocks including normalized opening-high-low-closing prices (OHLC) and trading volumes from professional Wind-Financial Terminal. For fair comparison, we follow [Sawhney et al., 2021] and generate samples by moving a 16-day lookback window along trading days. We keep $p = 0.85$, $0.85$, $0.90$ for NASDAQ, NYSE and TSE respectively, and set the hop of graph convolutional operation to 2. For temporal modeling, we test stacking 1-5 Lit layers with varied skipping rates. The reported results utilize a 3-layers’ hierarchy assigning $\delta_{[1:3]}$ to $1 \rightarrow 2 \rightarrow 3$ and the number of attention heads $H$ to 6 according to scores on validation set. The dimension of hidden feature space $d_f$ is 16. The loss factors are set to $\alpha = 4$ and $\eta = 0.5$. We tune the model and ablation variants on a GeForce RTX 3090 GPU by Adam optimizer [Kingma and Ba, 2015] for 100 epochs, the learning rate is 1e-3 and batch size is 16.

**Metrics.** Following previous studies [Feng et al., 2019b; Sawhney et al., 2021], we adopt a daily buy-hold-sell trading strategy to assess the profitability of ALSP-TF in terms of Sharpe ratio (SR) and cumulative investment return ratio (IRR). That is, the trader buys $\kappa$ stocks with the highest expected revenues once the market is closed on day $t$, then sells off these shares on next day’s close market. Formally,

$$
\text{IRR}^t = \sum_{s \in S^t} \frac{p_{t+1} - p_t}{p_t},
$$

where $S^t$ denotes the stocks in portfolio on day $t$. SR denotes the stocks in portfolio on day $t$. SR is a measure of risk-adjusted return describing the additional earnings an investor receives for per unit of increase in risk. We also compare the model’s ranking ability adopting the widely used metric nDGC@5. We report the mean results of five individual runs for $\kappa = 5$.

Figure 3: Ablation study over different components (Graph smoothing operation, Local interaction (Lit) & Time-adaptive modulator in Transformer, Self-supervised graph proximity loss) on NASDAQ.

Figure 4: Influence of hyper-parameters $\rho$, $\Delta T$, and $\kappa$. (a) Graph structure (b) Lookback window (c) Trade stocks

4.2 Overall Performance
We consider four categories of baselines for comparison. The results are shown in Table 1, from which we have several observations: 1) In general, RL and ranking approaches (e.g., IRDPG, RSR) perform better in investment returns than conventional price classification and regression methods (e.g., HATR, SFM), which justifies the effectiveness of learning to rank optimization toward stock selection. 2) Transformer-based encoders (HMG-TF, RAT) appreciably have better capability of modeling stock temporal dependencies, while it is hard for RNN-based models (e.g., SFM, Data-RNN, SAE-LSTM) to harvest fine-granular interactive information among discontinuous time steps. 3) Exploiting inter-stock relationships is demonstrated to be conducive to investment forecasting (e.g., RSR, STHAN-SR). This accentuates the collective synergistic effect of stock dynamics. Whereas, the requirement to predefine graph topologies based on domain knowledge raises the difficulty of generalizing these methods in new scenarios. 4) By revamping Transformer to hierarchically perform pattern-wise interactions being aware of time intervals and fusing self-supervision signals of stock proximity, our proposed ALSP-TF obtains the best results across all datasets. Specifically, it fetches an average relative performance gain of 8.57% and 18.54% in regard of risk adjusted returns and cumulative profits (t-test $p < 0.01$) over the best baselines. In addition, greater degrees of improvements are observed on NASDAQ and NYSE datasets than TSE, which may indicate the advantage of ALSP-TF in dealing with large candidate pools for stock portfolio selection.

4.3 In-depth Analysis
Next we conduct further analysis to learn the influence of various components and key hyperparameters in ALSP-TF.

Ablation Study. We investigate the effect of ablated variants from perspectives of both temporal and relational embedding. Due to space limitation, we depict the results on NASDAQ in Fig. 3, similar regularities can be observed on other datasets. As shown, different components jointly contribute to the performance. The main benefits stem from the locality and time-sensitive peculiarities inside Transformer blocks, which serve for extracting multi-grained dynamic patterns and endow different attention heads with adaptive ability to time intervals. The variant only retaining the temporal module can beat other Transformer baselines (HMG-TF, RAT) as well as state-of-the-art STHAN-SR that integrates LSTM and hypergraph convolution of pre-fixed corporate relations. Besides, introducing the graph-based smoothing and regularization further helps generate more stable and profitable stock selections. This inspires us that the data-driven relation embedding channel can furnish useful hidden dependency information when prior domain knowledge is unavailable.

Parameter Sensitivity. We next look closer to building stock graphs with different sparsities. It can be seen from Fig. 4a that either free of the proximity guidance (i.e., sparsity = 1.0) or pushing too many edges will cause degradation of performance. This is intuitive because among the vast number of inter-stock connections, only a few are meaningful enough to significantly influence the dynamic of related stocks. Fig. 4b elaborates the impact of varied lookback lengths. We find that the advantage of ALSP-TF compared to STHAN-SR is greater in the case of longer input, revealing the merit of mining elaborate interactions among sequential patterns. We also explore the change of profitability regarding the number of selected top-rank stocks. Fig. 4c demonstrates ALSP-TF’s suitability to trading strategies accompanied by different risk appetites.

Interpretation of Time-adaptive Self-attention. We further interpret what is of importance the time-adaptive modulator in self-attention learns. Taking NASDAQ as example, the upper part of Fig. 5 shows the proportion of positive/negative $a_h$ and $v_h$ parameters tuned for the learnable rescale function $f(\cdot)$ over all attention heads. It is intriguing that the number of positive $a_h$ gradually increases along with the stacking hierarchy. It means that lower-layer attention heads prefer to capture interactions around local contexts, while the heads at higher layers are responsible for modeling both short- and long-term dependencies. In addition, most values of $v_h$ are positive, which may indicate that time information
has relatively strong impact on weighing the pattern interactions. Moreover, the bottom half of Fig. 5 visualizes the attention heatmaps of a stock sequence produced by different heads. We find the attention scores are structurally sharper supported by the time-adaptive modulator. Specifically, the first heatmap concerns more on long interval dependencies, the matching between time-steps $5_{1h}, 13_{1h}$ and global contexts is prominently highlighted. In contrast, the latter two heatmaps put more emphasis on the information exchange inside local contexts. The interactive areas upon $L_2$ layer’s embedding are broader, probably because self-attention acts on patterns of a higher-level granularity. These results show that our model can flexibly catch dynamic time factors and adjust attention on multi-grained sequential signals.

5 Conclusion
In this paper, we present ALSP-TF, a new temporal-relational embedding framework for stock selection. The temporal module performs hierarchical representation and interaction of stock dynamic patterns based on a modified Transformer encoder. Different from vanilla self-attention that is context and position agnostic, our model can adaptively capture short- and long-term pattern matching signals taking advantage of locality and time-aware peculiarities. For relational view, we propose a graph self-supervised regularization, which integrates collective synergies of stocks while relieving the dependence on prior domain knowledge. Through quantitative and qualitative analyses on three global market datasets, we probe the effectiveness of ALSP-TF and set forth its applicability in investment forecast and recommendation.

Acknowledgements
This research is supported by the National Key Research and Development Program of China (No. 2021ZD0111202), the National Natural Science Foundation of China (No.62176005, 91846303), PKU-Haier “Smart Home” Qingdao Joint Innovation Lab, Project 2020BD002 supported by PKU-Baidu Fund, and Research Funds for NSD Construction, University of International Relations.

References


