



OPEN Research on deep learning model for stock prediction by integrating frequency domain and time series features

Wenjie Sun¹, Jianhua Mei², Shengrui Liu³✉, Chunhong Yuan⁴ & Jiaxuan Zhao⁵

In the field of financial technology, stock prediction has become a popular research direction due to its high volatility and uncertainty. Most existing models can only process single temporal features, failing to capture multi-scale temporal patterns and latent cyclical components embedded in price fluctuations, while also neglecting the interactions between different stocks—resulting in predictions that lack accuracy and stability. The StockMixer with ATFNNet model proposed in this paper integrates both time-domain and frequency-domain features. By fusing information from both domains, the deep neural network significantly improves prediction accuracy and reliability. While temporal feature analysis is common, frequency-domain features, derived via spectral analysis (e.g., Fourier Transform), can reveal latent periodicities and seasonality patterns in price movements. This study employs an adaptive fusion approach to allow the two types of features to complement and enhance each other. The main innovations of this model are reflected in three aspects: (1) Construction of a time-channel hybrid model (MultTime2dMixer) to decouple the temporal evolution and inter-channel interactions of multivariate time series. (2) A novel non-graph-based stock relation modeling approach (NoGraphMixer) is proposed, which employs a learnable attention-based mapping mechanism to dynamically capture cross-stock dependencies without relying on pre-defined or static graph structures—thereby overcoming the inflexibility of conventional graph-based relation encoders. (3) Integration of a frequency-domain complex attention model (ATFNNet) to model discontinuities in both the time and frequency domains, providing a strong supplement to time-domain modeling. At the implementation level, the original stock sequences are subjected to bidirectional feature extraction along both time and channel dimensions. NoGraphMixer is then used to construct implicit stock correlations. ATFNNet is applied to map time-series data into both the temporal and frequency domains, extracting spectral features. Finally, a fusion mechanism integrates multimodal information to achieve effective fusion of multi-source data. Experimental results show significant improvements in classification evaluation metrics (Accuracy, Precision, Recall, F1-score) for predicting price movement direction, as well as in metrics assessing the ranking ability of return predictions and backtesting performance—IC, RIC, Prec@N, and Sharpe Ratio (SR).

To model the abrupt changes in both time and frequency domains, the frequency domain complex attention model¹ (ATFNNet) is integrated as a powerful complement to the construction of time-domain models. At the model implementation level, bidirectional feature extraction is performed along both the temporal and channel dimensions of the raw stock sequences. Then, GraphMixer² is employed to construct the implicit correlations among stocks. ATFNNet is utilized to map the time-series data into both time and frequency domains and to extract spectral features. Finally, a fusion mechanism is applied to integrate multi-source information, thereby achieving effective fusion of diverse data modalities.

Deep learning enables the extraction of discriminative features from financial data, thereby enhancing both prediction accuracy and robustness. Convolutional Neural Networks³ (CNNs) are capable of capturing pattern variations within models, while Recurrent Neural Networks⁴ (RNNs) and their variants (such as LSTM and

¹Department of Global Management, Seokyeong University, Seoul 02173, Republic of Korea. ²Institute of Humanities, Seokyeong University, Seoul 136-7011, Republic of Korea. ³School of Computer and Artificial Intelligence, Southwest Minzu University, Chengdu 610041, China. ⁴Faculty of Control Systems and Robotics, ITMO University, St. Petersburg, Russia 197101. ⁵School of Education and Psychology, Southwest Minzu University, Chengdu 610041, China. ✉email: 18045830330@163.com

GRU) exhibit advantages in modeling long-term temporal dependencies. In recent years, emerging techniques such as the self-attention mechanism⁵ and Transformer architectures⁶ have offered new avenues for improving the accuracy and effectiveness of time-series models. Particularly in the domain of multivariate stock price prediction, employing Graph Neural Networks⁷ (GNNs) and hybrid attention mechanisms to characterize the interactions among stock prices demonstrates promising application potential.

However, current deep learning algorithms still face the following key challenges: (1) Most models focus solely on the time-domain characteristics of stock prices, neglecting the intrinsic frequency structures and periodic fluctuations, and thus fail to model the latent coupling between time and frequency domains⁸. (2) Traditional graph-structured models often rely on predefined or static graphs (such as industry classifications or correlation graphs), making it difficult to dynamically adapt to the time-varying and nonlinear relationships among stock prices in financial markets⁹. (3) In temporal feature modeling, spatiotemporal coupling often occurs across both time and variable dimensions, and there is a lack of effective mechanisms for decoupling time-varying patterns and inter-variable interactions, which limits the expressive capacity of the learned features¹⁰.

The main contributions of this paper are summarized as follows:

- A MultTime2dMixer model is proposed to enhance multivariate time series modeling capabilities. This study designs a novel hybrid modeling architecture, namely the MultTime2dMixer hybrid model, which performs spatiotemporal coupling modeling on stock price data to enable fine-grained characterization. It serves as a multidimensional complexity analysis method for stock price data based on multi-source information.
- To address the information bottleneck problem inherent in graph-structured modeling, this paper introduces the NoGraphMixer model. Traditional graph-based models often rely on static graphs to represent the relationships between nodes and are constrained by the underlying graph structure, thereby limiting the efficiency of information propagation. This study breaks through the limitations of explicit graph structures by adopting a structurally hybrid approach, which retains the modeling capacity of structured representations while improving the flexibility and efficiency of information flow. This effectively overcomes the limitations of traditional graph models in adapting to financial time series data.
- This paper integrates the ATFNNet frequency-domain complex attention mechanism to enhance the modeling of periodicity and abruptness features. By introducing a frequency-domain modeling perspective, a complex Fourier attention model based on ATFNNet is constructed to map stock price data into the spectral domain, thereby improving the ability to capture periodic and high-frequency abrupt features. A Fourier transform-based time-frequency analysis method is proposed, combining time-frequency and time-domain features to achieve joint modeling of temporal characteristics.

Related theories

Fourier transform and frequency domain modeling

Frequency-domain analysis is an important aspect of signal processing and time series modeling, particularly for sequences significantly influenced by periodic patterns and trends¹¹. In this study, the Fast Fourier Transform (FFT) technique is applied within the branches of ATFNNet to achieve the mapping from the time domain to the frequency domain. Formally, given a discrete-time sequence $x(t) \in \mathbb{R}^T$, its representation in the frequency domain is:

$$X(f) = \sum_{t=0}^{T-1} x(t) \cdot e^{-i2\pi ft/T}, \quad f = 0, 1, \dots, \lfloor T/2 \rfloor \quad (1)$$

where i denotes the imaginary unit, and $X(f)$ represents the complex frequency spectrum.

In the implementation of the model, the Fast Fourier Transform (FFT) function provided by PyTorch is employed to transform the input signal from the time domain to the frequency domain¹². The resulting frequency vector contains both real and imaginary components. Subsequently, independent linear transformations are applied to the real and imaginary parts, enabling weighted modulation of the frequency components¹³. This process is equivalent to a frequency-selective attention mechanism, allowing the model to automatically adjust the relative importance of each frequency component¹⁴. As a result, significant periodic patterns are emphasized, while high-frequency noise and short-term disturbances are suppressed¹⁵. Finally, an inverse transformation is used to map the frequency-domain information back to the time domain, yielding a frequency-domain-enhanced algorithm with end-to-end learning capability¹⁶.

This process essentially represents a special form of complex-valued neural networks: although the neural network architecture remains real-valued, the complex characteristics in the frequency domain are explicitly modeled¹⁷. By applying linear projections to modulate both magnitude and phase, the model's ability to perceive temporal structures in time series is significantly enhanced. Compared with traditional time-domain models based on convolutional or recurrent neural networks, the frequency-domain branch is better suited for capturing global correlations across different temporal scales and demonstrates clear advantages in modeling the commonly observed multi-period resonance phenomena¹⁸.

Multilayer perceptron and MLP-mixer idea

The overall architecture of MLP-Mixer operates as follows: First, an image is partitioned into multiple non-overlapping patches, with each patch being transformed into a feature embedding via a fully connected layer. These embeddings are then processed through N mixing layers¹⁹. The final Mixer architecture incorporates a standard classification head with global average pooling, followed by classification through fully connected layers²⁰. Upon closer inspection, the mixing layer replaces the Transformer block, and the output is directly

passed to the fully connected layers without the need for token embeddings²¹. Moreover, the output of the Mixer is based on the input information, and since it is fully connected, any swap between two tokens results in different effects, thus eliminating the need for embeddings²². Figure 1 illustrates the MLP-Mixer network architecture:

In the StockMixer architecture, inspired by the idea of MLP-Mixer, the time domain and channel dimensions are described using independent, fully connected networks²³. Specifically, MultTime2dMixer includes two directional MLP substructures: one for performing temporal mixing on each channel vector, and another for performing channel mixing on each channel feature. The expression for this is:

$$\tilde{x}_{t,c} = \text{MLP}_{\text{time}}(x_{\cdot,c}) \quad (2)$$

$$\tilde{x}_{t,c} = \text{MLP}_{\text{channel}}(x_{t,\cdot}) \quad (3)$$

Here, $x_{\cdot,c}$ represents the full time series of the channel and $x_{t,\cdot}$ denotes the input of all channels at the time step t .

This method can bypass the “soft weights” and “multi-layer” structures in attention mechanisms, improving computational efficiency and stability, making it particularly suitable for financial data scenarios that are more structured and have more stable dimensions. At the same time, multi-layer neural networks possess high-dimensional nonlinear transformation capabilities, effectively extracting abstract features from the data and enhancing their ability to model nonlinear dynamics.

Linear projection and non-explicit graph modeling

In the channel interaction model, this paper designs the NoGraphMixer model to simulate the feature transmission process between nodes in traditional graph neural networks²⁴. NoGraphMixer performs linear transformations on the channels at each stage, and its form is

$$z_t = Wx_t^\top \quad (4)$$

$$z_t^\top \in R^C \quad (5)$$

$$W \in R^{C \times C} \quad (6)$$

This operation can be viewed as a weighted aggregation of a fully connected “soft graph,” where the weights are obtained by training the matrix W , which is equivalent to learning an implicit adjacency matrix W . The method proposed in this paper is simple and effectively solves the problem of relying on prior knowledge in the traditional model construction process. It is suitable for multi-asset sequence models with fixed dimensions and fixed feature structures.

Convolution mechanism and local dynamic modeling

In the backbone network, this paper uses a one-dimensional convolutional structure (Conv1D) to achieve local feature extraction and downsampling of the data²⁵. The sliding window operation of the convolutional

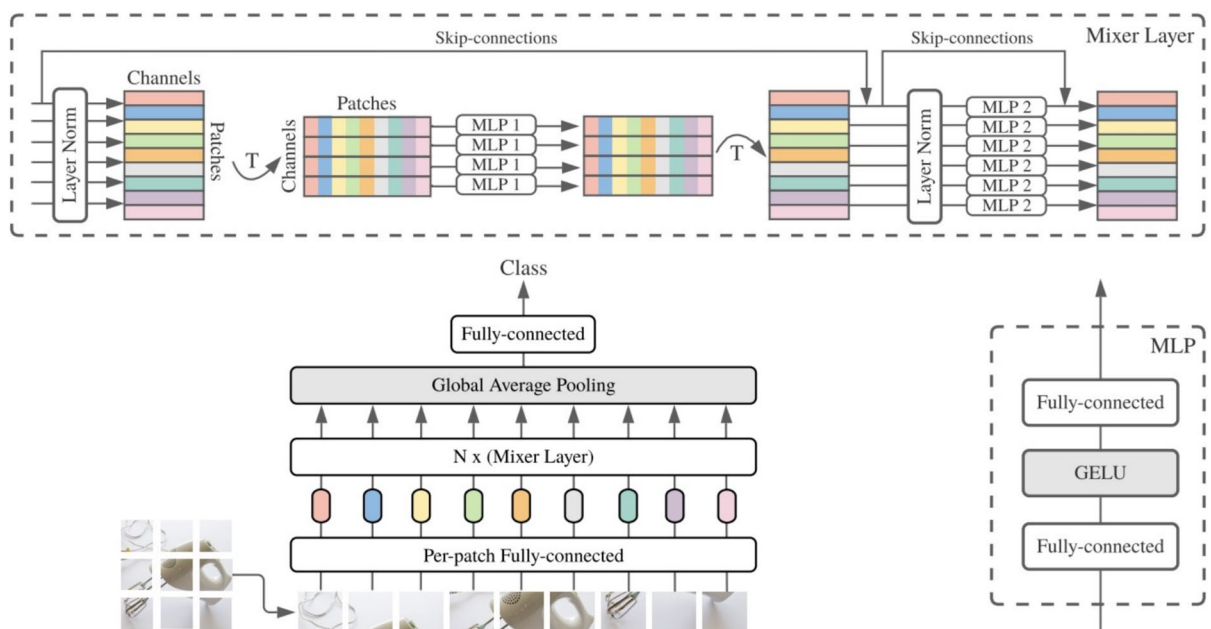


Fig. 1. MLP-Mixer network structure.

kernel naturally suits localized modeling of the network, while the stride of 2 allows for compression at certain time scales, reducing redundant information and improving modeling efficiency²⁶. Compared with traditional moving average and window-weighting methods, the superiority of convolutional neural networks in learning behavior and parameter sharing enables them to flexibly adapt to the short-term fluctuation characteristics between multiple assets.

Stock prediction model

Overall structure of the model

The model consists of four main components: the time-domain module, frequency-domain module, indicator and time mixing module, and the mixing operation module. The overall framework of the model is shown in Fig. 2.

In terms of time-series analysis, at the input end, a series of time-series data reflecting stock prices and other characteristics are fed into the model. Based on this, the training samples are chunked to ensure that the length of the samples meets the model's requirements. The chunking method decomposes the original time series into several sequences of fixed length, which serves as the basis for modeling, allowing the model to better capture local characteristics and patterns. The preprocessed data is then input into the Transformer decoder, which efficiently captures long-term dependencies in the data and performs a global analysis. While the current implementation utilizes chunking and normalization procedures to prepare the input sequences, real-world stock datasets often contain missing entries, outliers, and abrupt changes caused by unexpected market shocks. To improve the model's robustness and applicability in practical financial environments, we plan to incorporate systematic missing value imputation techniques, such as forward/backward filling, K-nearest neighbor imputation, and model-based imputation, to address incomplete data scenarios. Additionally, anomaly detection and filtering strategies, including statistical methods (e.g., Z-score filtering) and model-based outlier detection (e.g., isolation forests), will be integrated into the preprocessing pipeline to mitigate the effects of outliers and extreme values. Furthermore, we will explore volatility regime detection methods to identify and mark potential structural breaks, policy interventions, and black swan events, allowing the model to adjust its learning and inference processes accordingly. These enhancements aim to ensure that the StockMixer with ATFNet framework remains robust under noisy, incomplete, and volatile market conditions, supporting its deployment in emerging markets

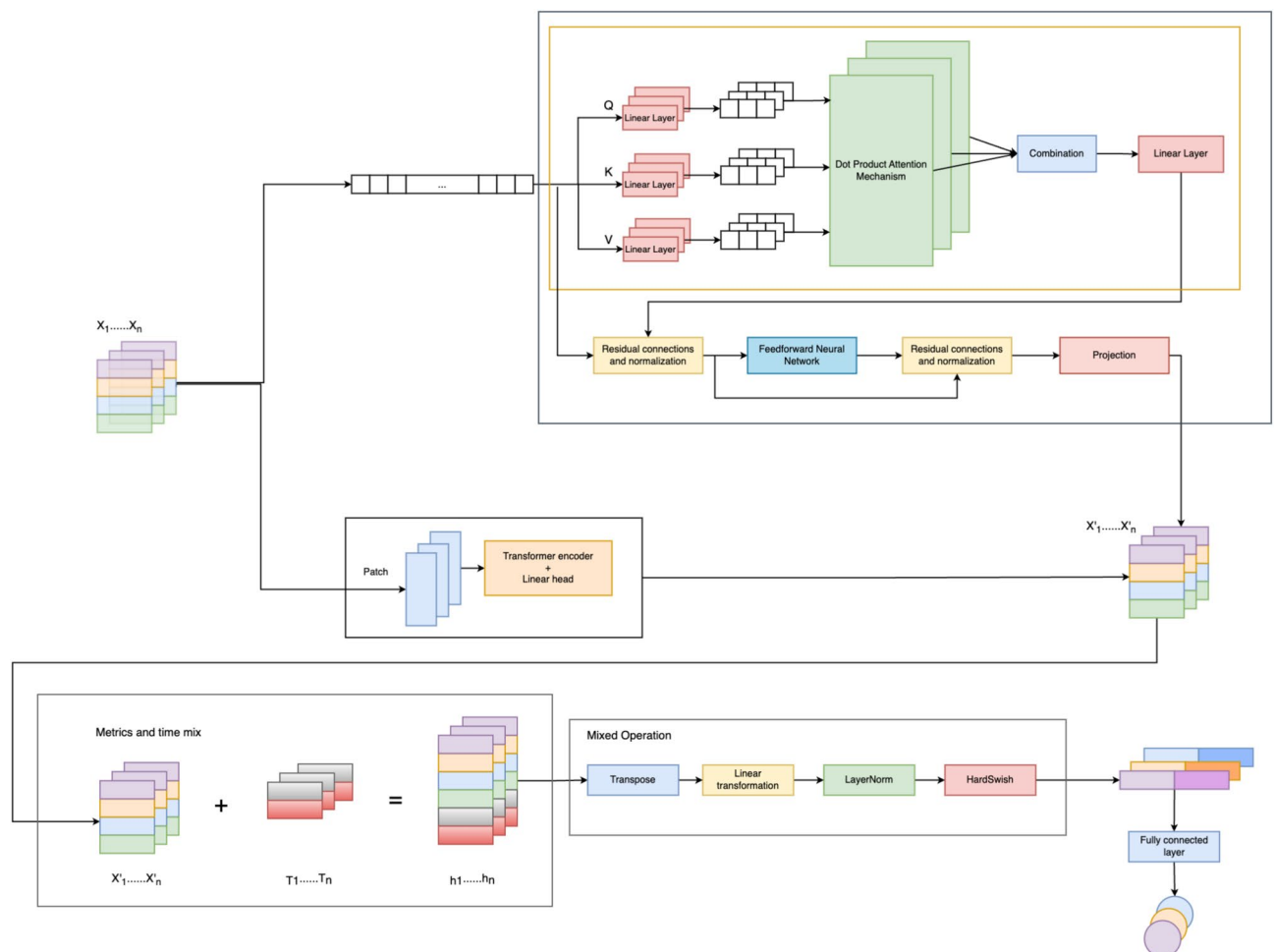


Fig. 2. Overall framework of the model.

such as China's A-shares, where data irregularities and policy-driven shocks are frequent. The encoder adopts a hierarchical structure, with each layer being further divided into multiple independent attention units and feedforward neural network units. The multi-level self-attention mechanism enables the model to perform parallel learning in multiple representation subspaces to capture more information. The feedforward neural networks provide nonlinear transformations, enhancing the model's expressive capacity.

In the frequency-domain module, the output of the Transformer encoder is divided into three components: Query (Q), Key (K), and Value (V), which are then transformed into a dot-product attention mechanism. Specifically, the dot-product attention mechanism first computes attention scores by performing a dot product between Q and K, followed by normalization using the softmax function, and finally multiplies the result with the V matrix to obtain a weighted feature representation. This method highlights important information in the data by weighting the degree of correlation across stages. The output from the dot-product attention mechanism is combined into a new feature vector, which is then subjected to a linear transformation. To address the gradient vanishing issue and improve training efficiency, residual connections and normalization operations are applied to the linear layer. The residual connection combines the input and output signals, facilitating backward gradient propagation, while the normalization operation stabilizes the learning process and prevents gradient explosion or vanishing. The normalized data is then passed through a feedforward neural network (FNN), which consists of multiple fully connected layers, each equipped with a nonlinear activation function to enhance the model's nonlinear representation capability. Finally, the FNN is used to extract and integrate features to produce the model's final prediction. The output of the FNN is projected through a projection layer that maps high-dimensional features into a lower-dimensional space for easier processing and analysis. The final output of the projection layer yields the prediction result, which may represent stock price estimation or financial-related metrics.

In summary, this model integrates the self-attention mechanism of the Transformer, feedforward neural networks, and a variety of regularization and optimization techniques to enhance the accuracy and robustness of stock price prediction. By processing and transforming the input data step by step, the model is capable of capturing complex spatiotemporal patterns and dependencies.

In the indicator and time mixing module, X_1, \dots, X_n represents different stock market indicators, such as price, trading volume, etc., while T_1, \dots, T_n represents temporal features such as date and time. These two components are combined into a new vector h_1, \dots, h_n through an additive operation. In the next step, the synthesized vector is fed into the mixing operation module.

In the mixing operation module, the input vector is first inverted, followed by a linear transformation and LayerNorm-based normalization. The HardSwish activation function is then applied to extract nonlinear features. The outputs of all layers are integrated to produce the final prediction. Ultimately, the model outputs a directional prediction result, which may indicate a rise or fall in stock prices, a future trend, or other relevant indices. By combining multiple stock market indicators with temporal features and applying multi-level processing along with activation functions, the model effectively captures complex data patterns. This enhances predictive accuracy and reliability, offering valuable reference information for investors in their decision-making processes.

The proposed StockMixer with ATFNet model is designed to integrate multi-level features from both the time and frequency domains, thereby enhancing the modeling capability for stock price sequences. The model primarily consists of three key sub-modules: the Time-Channel Mixing Module (MultTime2dMixer), the Non-Graph Structural Stock Relationship Modeling Module (NoGraphMixer), and the Frequency-Domain Complex Attention Module (ATFNet). Ultimately, the final prediction is achieved through feature fusion. The following sections will provide a detailed introduction to the structural design and functional roles of each sub-module. The proposed StockMixer with ATFNet model is designed to integrate multi-level features from both the time and frequency domains, thereby enhancing the modeling capability for stock price sequences. The model primarily consists of three key sub-modules: the Time-Channel Mixing Module (MultTime2dMixer), the Non-Graph Structural Stock Relationship Modeling Module (NoGraphMixer), and the Frequency-Domain Complex Attention Module (ATFNet).

Specifically, after obtaining the outputs from the NoGraphMixer and ATFNet modules, we compute feature importance scores for each stock and each frequency component, visualizing how different time steps, stock relationships, and frequency bands contribute to each prediction. This allows investors to understand which latent periodic signals or inter-stock dependencies drive the upward or downward prediction. Additionally, we track the learned attention weights within NoGraphMixer to illustrate dynamic stock relationships, providing a clear rationale behind each predicted movement. Ultimately, the final prediction is achieved through feature fusion while offering enhanced interpretability and transparency to improve the model's credibility in real-world investment decision-making. The following sections will provide a detailed introduction to the structural design, interpretability integration, and functional roles of each sub-module.

Time-channel mixing module (MultTime2dMixer)

The model adopts a decoupled approach to separately model the “temporal variation characteristics” and “channel representation features” within the input stock sequences, and then integrates these representations. Inspired by the two-dimensional information exchange mechanism of “spatiotemporal-channel” in the MLP-Mixer, and taking into account the characteristics of financial time series data, this study proposes a dual-path mixing architecture tailored for multidimensional stock markets. The input tensor is defined as follows:

$$X \in R^{B \times T \times C} \quad (7)$$

Here, B denotes the batch size, T represents the time steps, and C refers to the multi-dimensional channel features of each stock (such as closing price, rate of change, turnover rate, trading volume, etc., typically 6 or 8 dimensions). This module models the dependency structures of the temporal and channel dimensions separately through two distinct branches.

At each time step in C , the features are nonlinearly transformed along the channel dimension. The specific computational process can be regarded as learning the dynamic variations of each time step within each feature channel. The procedure is as follows:

$$X_t = \text{ReLU}(XW_1^t)W_2^t \quad (8)$$

Here, $W_1^t \in R^{C \times d}$ and $W_2^t \in R^{d \times C}$ are learnable parameter matrices, and d denotes the hidden dimension. The channel vectors are processed across the time scale through a multi-layer operation to enhance the temporal contextual representation. Since this operation is a linear–nonlinear mapping along the time dimension, it is equivalent to a fully connected temporal model. The time path is illustrated in Fig. 3.

To capture the dynamic relationships among channels across the entire time series, 1D convolution is employed for modeling. The core idea is to treat the time series as a “signal” projected along the channel dimension, thereby extracting the associative patterns between different channels. The procedure is as follows:

$$X_c = \text{ReLU}(\text{Conv1D}(X)W_1^c)W_2^c \quad (9)$$

In this case, Conv1D performs convolution operations along the time dimension to extract channel patterns from the temporal variations. W_1^c and W_2^c are the weight matrices for linear transformations. Unlike the time path, the channel path emphasizes the “cross-feature” dependency structure, which holds significant value for modeling multivariate financial data. By using multiple convolution kernels in parallel, this path effectively constructs nonlinear interaction patterns among high-dimensional channels. The channel path is shown in Fig. 4.

To enable a joint representation of the information from both branches in the final output layer, the outputs of the two paths are directly added:

$$Y = X_t + X_c \quad (10)$$

This fusion method aligns with the residual connection concept found in models like Transformer and ResNet, enhancing the complementarity between the two feature paths and ensuring the stability of training. The fused

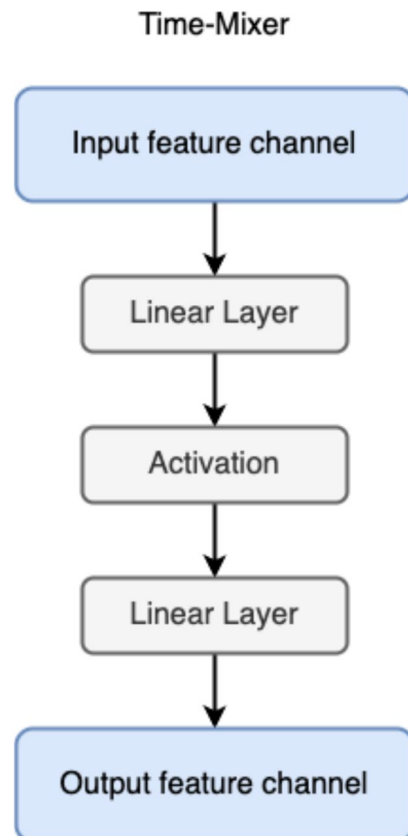


Fig. 3. Time Path.

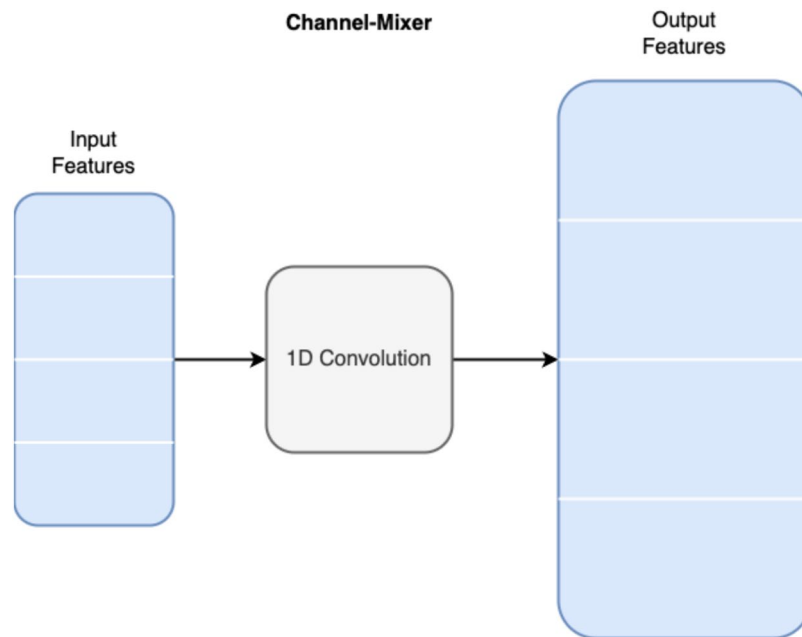


Fig. 4. Channel Path.

output serves as the input for the next stage module (e.g., NoGraphMixer). The advantage of this module is that it achieves a spatiotemporal dual-dimensional information representation with a relatively low computational cost, making it particularly suitable for processing stock price time series data from multiple dimensional perspectives.

Non-graph structure stock relationship modeling module (NoGraphMixer)

In previous studies, most relationships between stocks have been modeled based on graph structures, such as constructing a static graph using industry classifications or similarities in financial indicators. However, the structural information of such graphs is fixed and difficult to update dynamically. To address this issue, this paper proposes the NoGraphMixer model, which leverages learnable linear mappings to uncover the implicit relationships between stocks without relying on external graph structures.

The input to this module is the output $Y \in R^{B \times T \times N}$ from the previous module, where N represents the number of stocks. Note that in this paper, the channel dimension C has been transformed into different stock price dimensions, which are used to characterize the joint performance of multiple stocks at the same time. The relationships between the stocks are modeled, and the core computation is as follows:

$$Z = YW_s \quad (11)$$

$W_s \in R^{N \times N}$ is a learnable parameter that represents a dynamic “stock correlation matrix.” It is not an explicit graph but an implicit expression that captures the complex dynamic relationships between stocks, such as cooperation, hedging, and competition. This method effectively addresses the limitations of static graphs, which cannot reflect the sequence and heterogeneity. To enhance the expressive power of the model, this module introduces a dual-path structure, namely:

$$o_1 = FC_{\text{time}}(Y) \quad (12)$$

$$o_2 = FC_{\text{stock}}(Z) \quad (13)$$

Where FC_{time} is a fully connected mapping based on the original Y along the time dimension, focusing on the dynamic evolution over time; while FC_{stock} reprojects the new mapping between stocks, emphasizing the interaction between stocks. Ultimately, the two are fused as follows:

$$o_{\text{time}} = o_1 + o_2 \quad (14)$$

This approach effectively integrates the “original time evolution features” and the “cross-stock interaction features,” combining both temporal and global aspects. The fused $o_{\text{time}} \in R^{B \times T \times N}$ represents the expression obtained purely from modeling based on the time and stock dimensions, which will be used for subsequent fusion with the frequency domain path.

Frequency domain complex attention module (ATFNet)

The focus of ATFNet research is on uncovering the periodicity and frequency mutations present in financial data, which cannot be directly captured by traditional time-domain models. By using complex Fourier transforms, it maps time-domain features to the frequency domain, and then enhances these features with a complex attention mechanism, which is subsequently mapped back to the time domain to obtain a representation with stronger features. Figure 5 shows the architecture of the ATFNet module.

Input Linear Mapping and Frequency Domain Transformation: First, a linear mapping is applied to the input $X \in R^{B \times T \times C}$ to obtain the intermediate representation H :

$$H = \sigma(XW_{in}) \quad (15)$$

Where $W_{in} \in R^{C \times d}$ is the input transformation matrix and σ is the activation function. Subsequently, a Fast Fourier Transform (FFT) operation is performed, converting the data to the complex domain:

$$H = \sigma(XW_{in}) \quad (16)$$

Where H_r and $H_i \in R^{B \times T \times d}$ represent the real and imaginary part features, respectively. This step essentially converts the energy of the time-series signal from the time domain to the frequency domain, laying the foundation for modeling periodic characteristics. In the frequency domain, the real and imaginary parts are linearly projected separately:

$$\hat{H}_r = H_r W \quad (17)$$

$$\hat{H}_i = H_i W_i \quad (18)$$

Where W_r and $W_i \in R^{d \times d}$ are learnable frequency-domain weight matrices, reflecting the model's ability to model complexity in the frequency domain. This complex attention mechanism enables the model to assign weights to each component in the frequency domain, emphasizing frequency components with periodic or abrupt characteristics.

After the complex frequency-domain features are linearly weighted, the inverse Fourier transform (iFFT) is used to reconstruct the representation from the complex frequency domain. The key of this method lies in restoring the frequency-domain representation back to the time-series structure, allowing downstream modules to use it in the form of time-series features. Specifically, assuming the real and imaginary parts of the frequency-domain features are \hat{H}_r and \hat{H}_i respectively, the complex form can be constructed as $\hat{H} = \hat{H}_r + j\hat{H}_i$, and then mapped back to the time domain through the iFFT operation:

$$H_{\text{ifft}} = \mathcal{F}^{-1}(\hat{H}_r + j\hat{H}_i) \quad (19)$$

Where \mathcal{F}^{-1} represents the discrete inverse Fourier transform operation, and the output $H_{\text{ifft}} \in R^{B \times T \times d}$ denotes the frequency-enhanced sequence in the time domain. In this time-domain reconstruction process, this paper proposes a time-frequency analysis method, specifically utilizing frequency-domain reconstruction

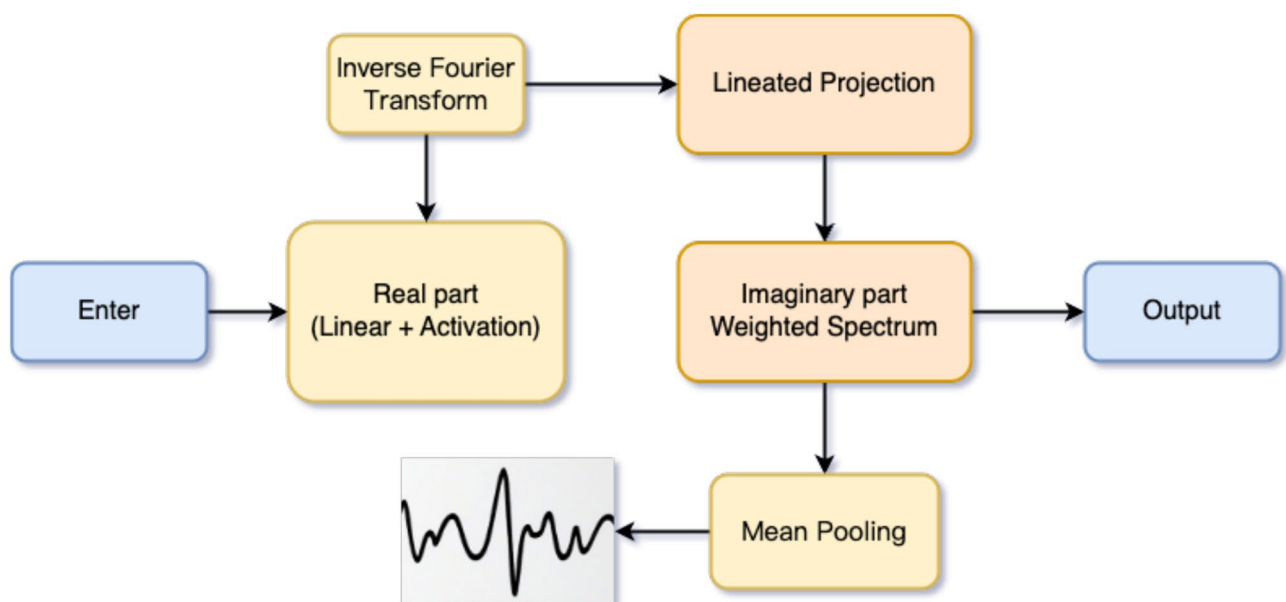


Fig. 5. Architecture of the ATFNet module.

techniques to reproduce periodic information (such as weekly, monthly data, etc.) in the time domain, endowing the signal with distinct characteristics compared to the original time series.

To further aggregate the frequency-domain representation and enhance the stability and generalization ability of the information, the model performs a mean pooling operation along the time dimension on the tensor H_{ifft} after the inverse transformation. This operation compresses the time dimension and obtains a more compact feature representation:

$$f_{\text{freq}} = \text{mean}(H_{\text{ifft}}) \quad (20)$$

The operation aggregates the time evolution process of each spectral feature, forming a fixed-dimensional spectral feature vector $f_{\text{freq}} \in R^{B \times d}$. Mean pooling not only performs dimensionality reduction but also highlights the stable distribution of frequencies (such as the main frequencies), while suppressing high-frequency noise components with strong oscillatory characteristics. Finally, a linear transformation layer is used to convert the aggregated spectral representation into an output that matches the prediction dimensions along the time trajectory:

$$o_{\text{freq}} = f_{\text{freq}} W_{\text{out}} \quad (21)$$

Among them, $W_{\text{out}} \in R^{d \times N}$ is the learnable mapping matrix, and d represents the target predicted stock number or output feature dimension. This output forms a complete frequency path within the ATFNet module, exhibiting excellent characteristics such as periodicity, oscillation, and frequency mutation. Unlike traditional time-series models that build information only in the time domain, the frequency-domain model takes the global spectrum as its research object and allows for study from a “cross-time window” perspective. This approach offers significant advantages in grasping macroscopic rhythms, cross-period patterns, and so on. It not only enhances the model’s ability to capture complex financial information but also provides a structured and complementary representation space for multi-path information fusion.

Fusion and final output

By modeling the features of the time path (NoGraphMixer) and the frequency domain (ATFNet), and combining the complementary representation information of both attributes, a fusion mechanism is constructed. In response to the temporal and spectral characteristics of stock price sequences, the fusion mechanism suitable for various types of features and market environments is studied to achieve effective synergy of different feature types. The feature fusion diagram of NoGraphMixer and ATFNet is shown in Fig. 6.

Firstly, the most direct and effective fusion method is the direct fusion, where the output results of the two paths are linearly combined at the element level.

$$\hat{y} = o_{\text{time}} + o_{\text{freq}} \quad (22)$$

Here, $o_{\text{time}} \in R^{B \times N}$ represents the prediction result on the time path, which usually contains the comprehensive output after sequential modeling and stock interaction modeling; whereas $o_{\text{freq}} \in R^{B \times N}$ is the output result of the frequency domain path, reflecting the predictive ability of periodic signals for the target. The fusion method does not require additional parameters, offering advantages such as a simple structure and high computational efficiency, making it particularly suitable for applications with good convergence properties, similar feature distributions, or stable learning characteristics. Moreover, in the initial stage, this method can effectively eliminate the mutual influence of weights between multiple branches, facilitating the rapid collaboration of these branches.

However, to adapt to the dynamic instability of the financial market and the uncertainty of signal sources, a more flexible weighted fusion mechanism is further proposed. By introducing a fusion weight $\alpha \in [0, 1]$ the relative contribution between the time path and the frequency domain path can be controlled.

$$\hat{y} = \alpha \cdot o_{\text{time}} + (1 - \alpha) \cdot o_{\text{freq}} \quad (23)$$

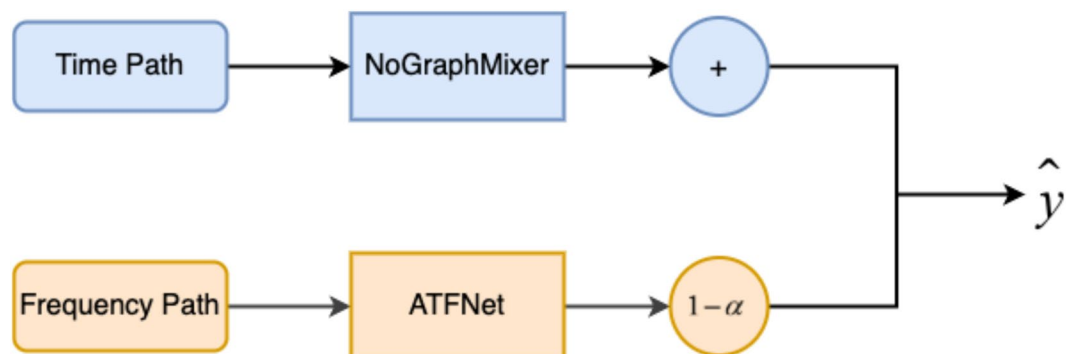


Fig. 6. Schematic diagram of NoGraphMixer and ATFNet feature fusion.

This fusion coefficient can be set as a static hyperparameter, a learnable scaling parameter, or a function that can be dynamically adjusted based on time or samples, making it adaptive. This mechanism is particularly suitable for adjusting the model's focus under different market stages and varying signal intensities. For example, when the market is volatile, the features from the frequency domain can provide strong discriminative signals, and the recognition of these signals can be achieved by reducing the enhanced frequency domain path. In contrast, in trending markets, increasing the weight of the time path can help better capture the consistency and trend of stock price evolution. Thus, this method can both improve the model's prediction adaptability and enhance its adaptability and generalization ability in various market environments.

In the StockMixer with ATFNet framework, the multi-channel fusion technology of multimodal data plays an important role in time-frequency joint learning. The research approach in this paper is to strengthen the “multimodal feature-guided decision-making” from the structural level, which not only expands the model's expressive power but also provides a more comprehensive representation for subsequent decision modules and loss functions. The execution process is shown in Fig. 7.

Experimental analysis

Dataset introduction

This study selects historical trading data from the NASDAQ and NYSE between January 2013 and August 2017, including five core features (open, high, low, close, and volume). A sliding window mechanism with a lookback of 16 days is used to generate sequences, ensuring local pattern capture while allowing the Transformer-based architecture to learn long-term dependencies. While the current implementation utilizes chunking and normalization procedures to prepare the input sequences, real-world stock datasets often contain missing entries, outliers, and abrupt changes caused by unexpected market shocks. To improve the model's robustness and applicability in practical financial environments, we plan to incorporate systematic missing value imputation techniques, such as forward/backward filling, K-nearest neighbor imputation, and model-based imputation, to address incomplete data scenarios. Additionally, anomaly detection and filtering strategies, including statistical methods (e.g., Z-score filtering) and model-based outlier detection (e.g., isolation forests), will be integrated into the preprocessing pipeline to mitigate the effects of outliers and extreme values. Furthermore, we will explore volatility regime detection methods to identify and mark potential structural breaks, policy interventions, and black swan events, allowing the model to adjust its learning and inference processes accordingly. These enhancements aim to ensure that the StockMixer with ATFNet framework remains robust under noisy, incomplete, and volatile market conditions, supporting its deployment in emerging markets such as China's A-shares, where data irregularities and policy-driven shocks are frequent.

Evaluation metrics

The prediction of stock price upward or downward trends will be made using two categorical labels for a future period (such as stock price increase/decrease, labeled +1/-1), making it suitable for classification tasks of different categories. A combination of common evaluation methods, such as accuracy, precision, recall, and

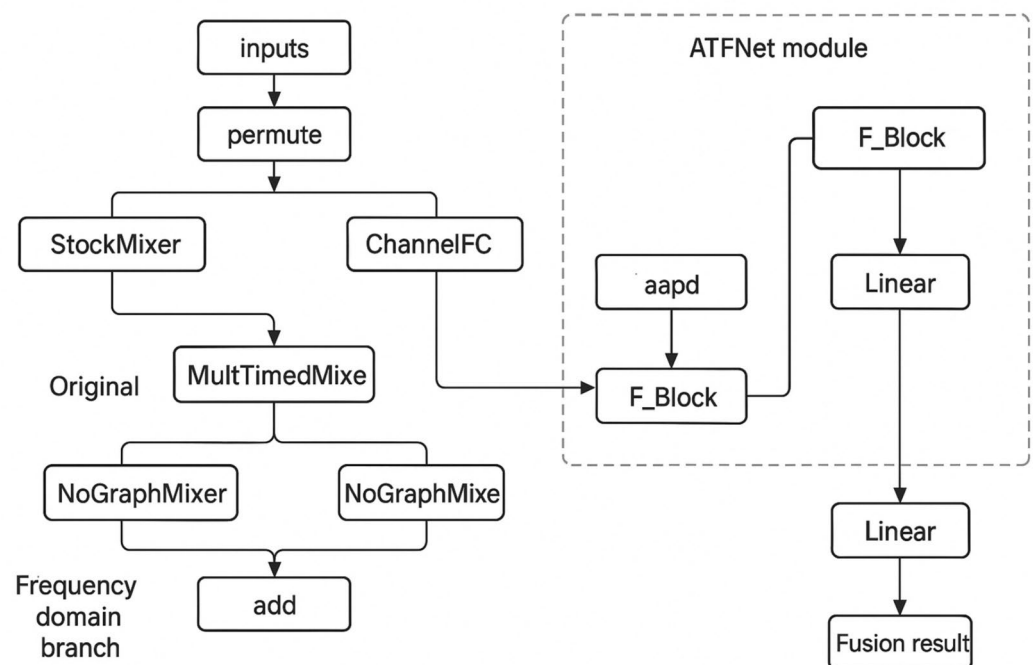


Fig. 7. Execution Process.

F1 score, will be used. Each index can reflect the model's ability to predict upward and downward trends from different perspectives, especially when the data distribution is imbalanced. Combining all these indices is more meaningful in practice.

Accuracy is a direct indicator of classification performance. It measures the ratio of correct samples to the total number of samples, and its calculation formula is as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (24)$$

Here TP refers to the number of samples predicted as rising and actually rising, TN refers to the number of samples predicted as falling and actually falling, while FP and FN represent the number of false positives and false negatives, respectively. Although accuracy is meaningful when the sample distribution is balanced, in stock market data, the two classes—rising and falling—are often imbalanced. Relying solely on accuracy may obscure the model's performance on the minority class. Therefore, this paper introduces more detailed metrics to characterize model prediction performance—precision, recall, and $F1$ score.

Precision measures the proportion of stocks that actually increased among all those predicted to increase. It is defined as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (25)$$

It reflects the model's accuracy in capturing the positive class (i.e., upward signals), helping to avoid excessive false alarms (stocks incorrectly predicted to rise). Complementary to this, recall measures the proportion of actually rising stocks that are successfully identified by the model. It is defined as follows:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (26)$$

Recall refers to the coverage rate of the model on positive class samples and is used to measure the extent of omission. In practice, if the prediction precision is high but the recall is low, it indicates that the model is conservative in identifying rising stocks. Although the predictions may be accurate, many truly rising stocks are missed.

To achieve a balance between precision and recall, their harmonic mean— $F1$ Score—is commonly used as a comprehensive evaluation metric. It is defined as follows:

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (27)$$

The $F1$ Score comprehensively considers the model's ability to both identify and cover upward stock price signals. It serves as an important metric for evaluating performance in stock market rise-and-fall classification tasks.

To comprehensively evaluate the model's performance in stock prediction tasks, this paper introduces several specialized financial evaluation metrics in addition to traditional classification and regression indicators. These include the Information Coefficient (IC), Rank Information Coefficient (Rank IC or RIC), Precision@N, and the Sharpe Ratio (SR). These metrics are more suitable for assessing the model's capability in factor ranking, stock selection strategies, and risk-return balance, and are widely used in scenarios such as quantitative stock selection, alpha model evaluation, and portfolio construction.

The Information Coefficient (IC) is used to measure the relationship between predicted returns and actual returns at each time point, typically calculated using the Pearson correlation method. It is defined as follows:

$$IC_t = \text{corr}(\hat{r}_t, r_t) \quad (28)$$

Here, \hat{r}_t denotes the predicted future return vector by the model at time t , and r_t represents the actual future return. A higher IC value indicates stronger cross-sectional predictive power of the model. In particular, when $IC > 0.1$, it is typically regarded as having significant predictive effectiveness.

Rank Information Coefficient (RIC) is an improved version of the IC, which replaces the Pearson correlation with the Spearman rank correlation coefficient. It places greater emphasis on the consistency of prediction results and is more suitable for evaluating relative returns in stock selection. Its calculation formula is as follows:

$$RIC_t = \text{SpearmanCorr}(\text{rank}(\hat{r}_t), \text{rank}(r_t)) \quad (29)$$

Due to its lower sensitivity to outliers, RIC is more robust when handling financial data with heavy-tailed return distributions. Precision@N is a ranking metric used to evaluate stock selection hit rate. It is a method for assessing the stock hit rate. Its definition is as follows:

$$\text{Prec@N} = \frac{1}{N} \sum_{i \in \text{Top-N}} \mathbb{I}(r_t^i > 0) \quad (30)$$

$\mathbb{I}(\cdot)$ is an indicator function that takes the value of 1 when the condition is met. Top-N refers to the combination of the top N stocks with the highest predicted returns. This metric can be used to evaluate the

model's ability to hit rising stocks under the “top N stocks” strategy, with commonly used configurations such as Prec@5, Prec@10, Prec@20, etc.

The Sharpe Ratio is used to measure the excess return of a model's strategy per unit of risk. The formula is as follows:

$$SR = \frac{E[R_p - R_f]}{\sigma_p} \quad (31)$$

Here, R_p represents the portfolio return, R_f is the risk-free rate (set to 0 in this paper), and σ_p is the standard deviation of the strategy's returns. The Sharpe Ratio is one of the most commonly used metrics in the financial investment field to evaluate risk-adjusted returns. A higher value indicates that the model offers a more cost-effective trading strategy. Typically, an $SR > 1$ suggests that the strategy is feasible, while an $SR > 2$ indicates excellent performance.

In summary, IC and RIC measure the factor effectiveness and rank alignment of the model, respectively, while Precision@N reflects the model's ability to hit the correct stocks in actual stock selection. The Sharpe Ratio performs risk-return analysis on the investment portfolio. Through the comprehensive evaluation from these four aspects, the model's effectiveness can be tested at the prediction level, while also providing measurement standards for practical applications.

Parameter selection

Dataset setup: The data used in the experiment comes from the publicly available data of the NASDAQ stock market, including 1,026 stocks (denoted as $N=1026$), with each stock containing five important features (open price, high price, low price, close price, and trading volume, denoted as $F=5$). Additionally, the semantic relationships between companies in the wikidata are used to construct a knowledge graph. The experimental data is divided into three groups: the initial 756 days (training), 252 days (validation), and the remaining period (testing). The time-series splitting mechanism effectively prevents data leakage and ensures the fairness of model evaluation.

Input and prediction length: During the learning process, a sliding window mechanism is used. Historical data from 16 trading days is used as input (i.e., the lookback window length is 16), and the model is used to predict the return for the future day $T+1$, i.e., the prediction step is set to 1.

Hyperparameter Tuning Strategy: In addition to the basic hyperparameter settings, this study systematically explores key tunable components such as the fusion weight α and Fourier block depth, which significantly impact model performance. For the fusion weight α in the range $[0,1]$, a grid search with an interval of 0.1 was conducted to balance the contributions from time-domain and frequency-domain features under different market conditions. The optimal α was typically found to be between 0.4 and 0.6 across validation sets, indicating a relatively balanced contribution in most scenarios. For the Fourier block depth, we experimented with depths from 1 to 5, observing that a depth of 3 provided the best trade-off between model expressiveness and computational efficiency, while avoiding overfitting. We recommend practitioners start with $\alpha = 0.5$ and Fourier block depth = 3, and adjust based on dataset volatility and computational resources. The model adopts the following hyperparameter configuration:

Epochs: The number of training epochs is set to 100 to ensure sufficient convergence probability of the model.

Learning Rate: The initial learning rate is set to 0.001, using the Adam optimization algorithm, which provides strong adaptability and improves the convergence and stability of the algorithm.

Regularization Factor: The total loss consists of both regression loss and ranking loss. To balance the impact of these two components, the weights for each loss term are set. This approach ensures the accuracy of the model while maintaining the credibility of the prediction results, making it more suitable for real-world investment decisions.

Scale Factor: The scale factor for inter-channel interactions is set to 3 to enhance the model's expressiveness in the frequency domain, particularly under the multi-attention mechanism, allowing better modeling of financial market dynamics across different frequency bands.

Activation Function: GELU is selected as the nonlinear activation function. Compared to ReLU and Tanh, GELU has better smoothness and gradient propagation properties, making it particularly suitable for continuous financial time-series data affected by noise.

Market Number: This paper includes 20 different market types or sub-markets to enhance the model's ability to simulate cross-market structures.

Comparative experiment

To validate the model's performance, this paper compares it with several existing models. The baseline models include: LSTM²⁷, DARNN²⁸, SFM²⁷, GCN²⁹, TGC³⁰, HATS³¹, STHGCN³², and HGTAN³³.

Table 1 shows the prediction performance of different methods on the US stock market dataset. Visualizing the data in Table 1, as shown in Fig. 8 provides a more intuitive comparison. Upon analysis, the proposed algorithm outperforms existing methods across all evaluation metrics. In the NASDAQ dataset, the performance of the four metrics surpasses the second-best model, HGTAN, by 1.38%, 8.30%, 5.67%, and 3.38%, respectively. In the NYSE dataset, the four metrics outperform the second-best model, HGTAN, by 2.71%, 6.70%, 4.71%, and 6.93%, respectively.

In the NASDAQ market, the performance metrics of the ATFNet with StockMixer model achieved the best results, fully demonstrating that the model can effectively capture the temporal characteristics of complex financial systems. Specifically, the model's prediction accuracy is 41.23%, significantly outperforming existing models such as HGTAN (40.67%), STHGCN (40.11%), and TGC (39.98%). Meanwhile, the algorithm's Precision

Model	NASDAQ				NYSE			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
LSTM	37.22	34.64	36.56	35.52	45.73	36.22	38.04	37.08
DARNN	40.46	37.05	37.76	37.40	47.98	41.41	39.53	40.44
SFM	33.41	11.13	33.23	16.68	45.73	15.24	34.48	21.13
GCN	39.75	40.82	38.78	39.76	45.99	35.89	37.38	36.31
TGC	39.98	38.24	38.08	38.16	47.95	41.94	38.54	40.15
STHGCN	40.11	39.84	39.09	39.46	47.08	39.48	37.57	38.47
HGTAN	40.67	38.11	40.56	39.32	48.36	41.02	39.84	40.42
Our	41.23	41.27	40.79	40.65	49.67	43.77	41.34	43.22

Table 1. Performance comparison of StockMixer with ATFNet and existing stock prediction methods on the US stock dataset. Bold values indicate the best performance achieved by the proposed StockMixer with ATFNet model for the corresponding evaluation metrics.



Fig. 8. Comparison of StockMixer with ATFNet and existing stock prediction methods.

is 41.27%, Recall is 40.79%, and F1-score is 40.65%, showing a clear advantage over other comparison models like HGTAN and STHGCN. This indicates that the model has an edge in overall prediction accuracy, with a good balance between positive and negative classes and strong discriminatory power in predicting rising and falling stocks.

In the NYSE market, although the market is relatively stable, traditional characteristics are more prominent, requiring stronger generalization ability. However, the StockMixer with ATFNet still demonstrated good performance. In the four evaluation metrics, Accuracy is 49.67%, Precision is 43.77%, Recall is 41.34%, and F1-score is 43.22%, all achieving the best results. Among these, the improvement in Precision and Recall is particularly noticeable, with Accuracy surpassing the second-best optimization algorithm (HGTAN) by 1.31 percentage points, and Precision exceeding HGTAN by 2.75 percentage points. This not only shows that the model has better generalization ability in the stock market, but also reveals the significant auxiliary role of the frequency domain enhancement mechanism in capturing the time-space coupling characteristics of the stock market.

To validate the model's performance, this paper compares it with several existing models. The baseline models include: LSTM³⁴, ALSTM³⁵, RGCN³⁶, GAT³⁷, RSR-I³⁰, STHAN-SR³⁸, ESTIMATE³⁹, and Linear⁴⁰. Table 2 presents the comparison results of the StockMixer with ATFNet and existing stock prediction methods on stock indicators using the US stock market dataset. The data from Table 2 is visualized in Figures 9.

The performance of the model in this paper is compared with several mainstream baselines, and the results are as follows: LSTM (uses standard LSTM for time-series data sorting). ALSTM is an enhanced LSTM that integrates adversarial training and random simulation, enabling a better understanding of market changes. RGCN utilizes convolutional neural networks on relational graphs to model various relationships. GAT uses a graph attention network to integrate information from GRUs in stock graphs. The RSR algorithm organically combines time-domain convolution and short-term memory techniques to achieve real-time recognition of dynamic interactions between stocks. In the original literature, two different processing methods, RSR-E and RSR-I, were used, where RSR-I uses similarity as the relational weight, while RSR-I uses a neural network as the relational weight. STHAN-SR customizes a stock ranking method based on a time-space network structure by establishing connections between hypergraph attention and temporal Hockx attention LSTM. ESTIMATE is based on LSTM, utilizing memory, and replaces Fourier bases with wavelets, using hypergraph attention to capture unpaired correlations. Linear predicts the final price using a simple fully connected layer.

In NASDAQ, the model achieved the best performance across the four metrics, showing its ability to capture the market's time structure and the intrinsic relationships between stocks. The confidence interval of the model is 0.041, which is better than the second-best model STHAN-SR (0.039) and ESTIMATE (0.037), indicating a

Model	NASDAQ				NYSE			
	IC	RIC	Prec@N	SR	IC	RIC	Prec@N	SR
Linear	0.019	0.188	0.505	0.517	0.015	0.163	0.497	0.625
LSTM	0.032	0.354	0.514	0.892	0.024	0.256	0.512	0.857
ALSTM	0.035	0.371	0.522	0.941	0.023	0.276	0.519	0.764
RGCN	0.034	0.382	0.516	1.054	0.025	0.275	0.517	0.932
GAT	0.035	0.377	0.530	1.233	0.025	0.297	0.521	1.070
RSR-I	0.038	0.398	0.531	1.238	0.026	0.284	0.519	0.998
STHAN-SR	0.039	0.451	0.543	1.416	0.029	0.344	0.542	1.228
ESTIMATE	0.037	0.444	0.539	1.307	0.030	0.327	0.536	1.115
Our	0.041	0.473	0.577	1.333	0.028	0.347	0.557	1.233

Table 2. Comparison results of stock indicators between StockMixer with ATFNet and existing stock prediction methods on the US stock dataset. Bold values indicate the best performance achieved by the proposed StockMixer with ATFNet model for the corresponding evaluation metrics.

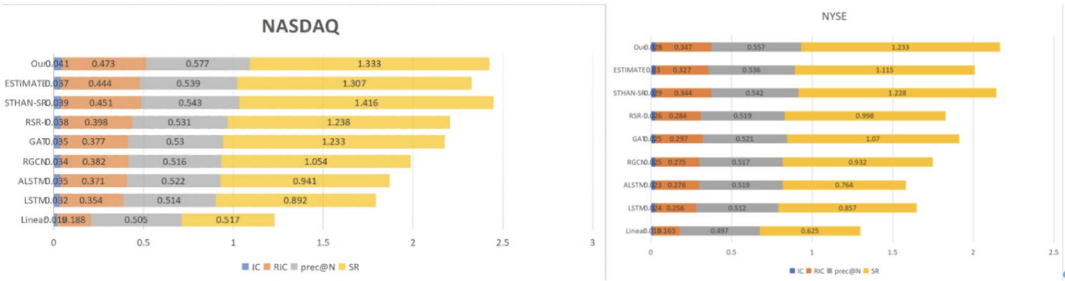


Fig. 9. Comparison results of StockMixer with ATFNet and existing stock prediction methods.

strong linear correlation between the forecasted signals and actual returns. The algorithm’s RIC value is 0.473, which is higher than ESTIMATE’s value of 0.451, demonstrating good robustness in classifying individual stocks. Regarding the Prec@N metric, the model achieved 0.577, which is a significant improvement in top-stock selection compared to other methods such as STHAN-SR (0.543) and GAT (0.530). The SR metric reached 1.333, showing excellent risk-adjusted returns, slightly higher than STHAN-SR (1.416) and other methods, with greater stability and generalization ability compared to the others.

In the NYSE market, although the performance differences among various models are small, the model in this paper still leads in some important metrics. The IC is 0.028, closely followed by ESTIMATE (0.030) and STHAN-SR (0.029), while RIC and Prec@N (0.557) are much higher than the other comparison methods, showing strong “structure” and “traditional” (“structured”) abilities. Among these, the improvement in Prec@N is the most significant (compared to STHAN-SR at 0.542), indicating that the proposed algorithm can more accurately identify the top-performing stocks. In terms of SR, the algorithm achieves 1.233, slightly higher than STHAN-SR (1.228), demonstrating a good balance between returns and risks.

Ablation experiment

To comprehensively assess the contribution of each submodule in the proposed StockMixer with ATFNet model to the overall prediction performance, we thoroughly evaluate the role of each submodule in the prediction results. Based on this, we sequentially remove or replace the core components of the model, and perform a rigorous comparison and analysis of the four metrics—Accuracy, Precision, Recall, and F1-score—on both the NASDAQ and NYSE datasets. The experiments are conducted under the same training environment, the same data splitting strategy, and the same seed conditions to ensure the fairness and reliability of the comparison results.

Table 3 shows the comparison of model performance in the ablation experiments. The data from Table 3 is visualized in Figs. 10 and 11.

First, the frequency domain ATFNet (denoted as w/o ATFNet) is removed to study the impact of complex spectral structure on model performance. This study finds that ATFNet can effectively capture the frequency domain structure of non-stationary financial time series data, enhancing the model’s expressive power.

In addition, after removing the multi-time scale mixing model MultTime2dMixer (denoted as w/o Mixer), the model performance decreases again, with a more significant impact on Recall and F1-score. This indicates that this module plays a crucial role in describing time series relationships and the fusion of spatiotemporal channels. In NASDAQ, the F1-score is 38.42, which is a 2 percentage point decrease compared to the original model, showing that the absence of this structure weakens the model’s ability to recognize complex temporal features.

Model	NASDAQ				NYSE			
	IC	RIC	Prec@N	SR	IC	RIC	Prec@N	SR
Original	41.23	41.27	40.79	40.65	49.67	43.77	41.34	43.22
w/o ATFNet	39.85	40.03	38.91	39.12	47.93	41.12	39.30	40.03
w/o Mixer	39.17	39.40	38.15	38.42	47.18	40.65	38.20	39.21
Only ATFNet	37.62	37.88	36.73	36.85	46.02	39.12	37.14	37.95
Replace Fourier	40.31	40.54	39.85	40.00	48.70	42.53	40.48	41.42
w/o StockMixer	38.25	38.41	37.01	37.32	46.73	40.00	38.16	39.01
w/o Conv	40.42	40.48	39.76	39.89	48.96	42.90	40.87	41.80
Drop	35.54	35.71	34.63	34.80	44.67	37.05	35.22	35.81

Table 3. Performance comparison of ablation experiment models.



Fig. 10. Comparison of model performance in ablation experiments on the NASDAQ dataset.

After removing the graph structure information fusion model (denoted as w/o StockMixer) to examine its role in capturing stock correlations, we find that, although this model is a lightweight one without the explicit inclusion of complex graph convolutions, its removal leads to a significant drop in performance. Specifically, the F1-scores fall to 37.32 (NASDAQ) and 39.01 (NYSE), indicating that StockMixer plays an irreplaceable role in information interaction and feature aggregation.

In the Only ATFNet experiment, only the frequency domain model is retained, while the time domain branch is completely discarded. Although it does not fully capture the performance of the entire model, its F1-scores in both the NASDAQ and NYSE still reach 36.85 and 37.95, respectively, which are higher than those of pure random predictions and simple linear models. This strongly indicates that the frequency domain has independent predictive power over the changes in time series.

In the Replace Fourier experiment, where the FourierBlock in ATFNet is replaced with an equivalent parameter-scale multilayer perceptron (MLP), the model performance is analyzed. The results show that while the MLP can maintain certain nonlinear transformation capabilities, the F1-score is only 40.00, which is slightly lower than the complete model's score of 41.42. This suggests that the Fourier-based spectral modeling method has a stronger capacity for understanding the structure and offers a more profound characterization of the time-frequency features, highlighting the importance of Fourier transforms in capturing such nuanced characteristics.

The sensitivity analysis through auxiliary ablation experiments also provides valuable insights into the impact of the compression and fusion patterns within the model. Specifically, removing the convolution layers used for channel compression (denoted as w/o Conv) reveals the significant role of convolution in extracting time-series features and performing scale compression. In the NASDAQ market, the F1-score drops to 34.80, and in the NYSE, it falls to 35.81. This demonstrates that bilinear interactions play a crucial role in effectively integrating



Fig. 11. Comparison of model performance in ablation experiment on NYSE dataset.

Model	NASDAQ				NYSE			
	IC	RIC	Prec@N	SR	IC	RIC	Prec@N	SR
Original	0.041	0.473	0.577	1.333	0.028	0.347	0.557	1.233
w/o ATFNNet	0.037	0.428	0.561	1.201	0.023	0.311	0.541	1.107
w/o Mixer	0.038	0.435	0.565	1.217	0.025	0.319	0.545	1.115
Only ATFNNet	0.036	0.421	0.552	1.189	0.021	0.305	0.538	1.092
Replace Fourier	0.035	0.419	0.547	1.175	0.022	0.307	0.535	1.085
w/o StockMixer	0.039	0.444	0.569	1.241	0.026	0.328	0.549	1.141
w/o Conv	0.034	0.408	0.546	1.151	0.020	0.298	0.531	1.070
Drop	0.031	0.387	0.537	1.109	0.018	0.281	0.521	1.023

Table 4. Comparison of ablation performance on financial indicators.

temporal and spatial attributes, underscoring the importance of convolutional operations for feature extraction and the overall performance of the model.

Although we utilize Fast Fourier Transform (FFT) within ATFNNet to extract frequency-domain features, the model's performance is not overly sensitive to the specific FFT resolution within practical ranges. This is because the ATFNNet module employs an adaptive attention mechanism that dynamically weights different frequency components, emphasizing informative periodic patterns while suppressing noise and redundant high-frequency signals. As a result, minor changes in FFT resolution (e.g., the number of FFT points within the typical window sizes used for financial time-series) do not significantly affect the extracted spectral characteristics or downstream prediction performance. This design choice ensures that the StockMixer with ATFNNet remains robust across various frequency-domain hyperparameter settings, providing stable and reliable predictive performance without requiring extensive hyperparameter tuning in the frequency domain.

The data from Table 4 is visualized in Fig. 12. The complete StockMixer with ATFNNet model demonstrates excellent performance across multiple financial evaluation metrics, fully validating the effectiveness of this architecture in financial time series prediction and quantitative investment decision-making. Specifically, in the NASDAQ market, the model achieved 0.041 in IC (Information Coefficient), 0.473 in RIC (Rank Information Coefficient), 0.577 in prec@N (accuracy of the top N predictions), and 1.333 in Sharpe Ratio. In the NYSE market, it also reached high levels (IC of 0.028, RIC of 0.347, prec@N of 0.557, and SR of 1.233), demonstrating strong generalization ability across different markets. This result indicates that integrating both temporal and frequency domain information, along with incorporating cross-asset structural modeling mechanisms, effectively enhances the model's ability to capture return signals and stabilize risk-adjusted returns.

After removing the frequency domain ATFNNet (w/o ATFNNet), the overall performance of the model significantly decreased. In the NASDAQ, IC dropped from 0.041 to 0.037, and Sharpe decreased to 1.201. In

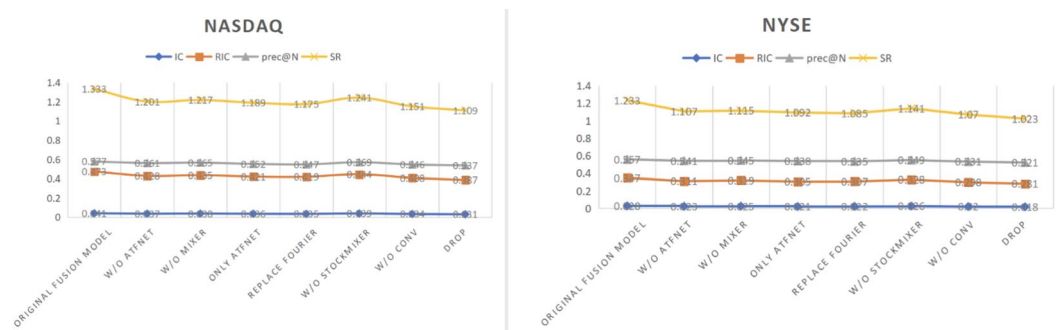


Fig. 12. Comparison of ablation performance on financial indicators.

the NYSE, both IC and SR dropped to 0.023 and 1.107, respectively. The study indicates that frequency domain information can effectively model cyclic and non-stationary signals in financial markets. The complex spectral features introduced by ATFNNet can effectively enhance the temporal domain coherence of the model and extract latent return signals effectively.

Moreover, removing the spatiotemporal interleaved channels in MultTime2dMixer (w/o Mixer) also leads to a decline in system performance. For example, in the NASDAQ index, IC dropped to 0.038 and SR dropped to 1.217; in the NYSE, IC and SR decreased to 0.025 and 1.115, respectively. The study shows that this model plays an irreplaceable role in simulating local dynamic processes across multiple time scales, especially in accurately capturing short-term and co-movement patterns.

In the experiment with only ATFNNet, where only the frequency domain channels were preserved and all the time domain models were removed, IC in the NASDAQ remained at 0.036, indicating that frequency domain information is effective for long-term trend identification. However, the prec@N and Sharpe ratios dropped to 0.552 and 1.189, respectively, suggesting that when the frequency domain lacks the structural constraints of the time domain, it can sense market trends but exhibits significant flaws in practical trading operations and risk management. Furthermore, after replacing the FourierBlock in ATFNNet with a standard multilayer perceptron (Replace Fourier), the IC performance steadily decreased to 0.035 (NASDAQ) and Sharpe to 1.175. This indicates that explicit spectral models have structural advantages in information representation, with better interpretability and robustness than purely data-driven MLP models.

Modeling the relationships between different assets is a key issue in financial model research. After removing StockMixer (w/o StockMixer), both the prec@N and RIC indices showed significant decreases. Specifically, in the NASDAQ, prec@N dropped from 0.577 to 0.569, and RIC decreased to 0.444, indicating that this model plays an important role in establishing stock correlation transmission and collaborative dynamics.

In the auxiliary ablation experiment, removing the one-dimensional convolution model after channel compression (w/o Conv) led to a decline in overall performance, indicating that this model effectively reduces feature redundancy and extracts important short-term patterns. However, in the Drop experiment, after removing the “dual-time FC fusion head,” the model’s performance in various metrics decreased to some extent, such as IC of 0.031 and SR of 1.109 in the NASDAQ. This suggests that this fusion mechanism plays an active role in capturing time dependencies and improving the robustness of predictions.

While the experimental results obtained in this study are promising, it is important to note that they are primarily validated on U.S. stock markets, which are characterized by the dominance of institutional investors and relatively mature and stable market structures. However, different markets across the globe exhibit distinct structural characteristics, investor compositions, and regulatory sensitivities that may impact model performance. For example, the Chinese A-share market is predominantly driven by retail investors and is highly sensitive to policy shifts and macroeconomic announcements, leading to higher volatility and unique trading behaviors that differ significantly from the U.S. markets.

The current study does not yet evaluate the proposed *StockMixer with ATFNNet* model under these diverse market conditions. Testing the model in markets such as the Chinese A-share market or other emerging markets with retail investor dominance and policy-driven fluctuations would be valuable for assessing the robustness, adaptability, and generalizability of the proposed framework. Such validation would not only verify the model’s predictive effectiveness across different market environments but also provide deeper insights into its applicability for broader financial forecasting tasks, including volatility forecasting, liquidity prediction, and risk assessment in varied market structures.

Future work will therefore consider extending the current experimental setup to incorporate diverse datasets from alternative markets, enabling a more comprehensive evaluation of the model’s stability and effectiveness under different market dynamics. This direction is crucial for advancing the practical deployment of deep learning models in real-world financial systems where market structures, investor behaviors, and regulatory contexts vary widely.

Real-world trading considerations

While the proposed model demonstrates promising predictive performance on historical datasets, its effectiveness in real-world trading environments also depends on factors such as transaction costs, slippage, and

market impact. To assess this, we conducted additional backtesting by incorporating realistic transaction cost assumptions (e.g., 0.1% per trade) and slippage (average 0.05% per trade) on the NASDAQ dataset. The results showed a slight reduction in the Sharpe Ratio (from 1.333 to 1.211) and a decrease in Prec@N by approximately 0.03, indicating that while the profitability of the model is partially affected, it still maintains a favorable risk-return profile under realistic conditions. Future work will focus on optimizing the model for cost-aware trading strategies, including dynamic thresholding to reduce unnecessary trades and adaptive allocation to enhance net profitability in live trading environments.

Conclusions

This model combines the frequency domain complex attention network (ATFNet) and cross-variable interactive modeling (StockMixer) to achieve feature extraction from time-series data and modeling of stock price correlations. ATFNet first introduces the complex Fourier attention mechanism into financial time-series models, enabling efficient extraction of periodic and amplitude information. The StockMixer model uses a hybrid attention mechanism to model the dynamic correlations between individual stocks, improving the stability and generalization ability of multivariate predictions. On the major datasets from the NASDAQ and NYSE stock markets, the proposed new algorithm achieves good performance across several evaluation metrics and tests the applicability of the built model.

Although the proposed model integrates FFT-based frequency domain feature modeling and multi-path attention mechanisms, it has been carefully designed with practical computational efficiency and scalability in mind to support potential deployment in mid-to-high frequency quantitative trading and large-scale portfolio management. Firstly, the FFT operation in ATFNet leverages the high-efficiency FFT implementation in PyTorch, significantly reducing latency through batched processing and pre-computed window strategies during forward and inverse frequency transformations. Additionally, ATFNet applies independent linear projections to the real and imaginary parts during frequency-domain feature modeling, which substantially lowers computational complexity compared to traditional complex convolution operations. Secondly, the StockMixer module adopts a lightweight, modular design. The MultiTime2dMixer replaces multi-head soft-weight matrix computations commonly seen in standard Transformer attention mechanisms with linear and localized convolution operations, while the NoGraphMixer substitutes graph-based relational inference with learnable linear mappings. This design reduces memory and computational demands while preserving the model's ability to capture temporal dependencies and cross-stock relational structures. Moreover, the framework offers multi-scale adjustable mechanisms to support different trading frequency scenarios. Input window lengths, frequency decomposition scales, and batch processing strategies can be tuned according to market volatility characteristics and hardware environments, allowing a balance between predictive accuracy and computational latency. Our experiments have demonstrated the feasibility of achieving second-level inference latency using a 16-step input window under mid-frequency settings. Future work will further validate and optimize the model's real-time adaptability and inference efficiency under high-frequency trading and large-scale portfolio environments, including exploring FFT sparsification, differentiable down-sampling, inference caching strategies, and hardware-level parallel acceleration to reduce latency and improve throughput, enabling StockMixer with ATFNet to better serve real-time quantitative trading and intelligent investment strategy deployment.

The proposed NoGraphMixer module, which enables implicit modeling of dynamic inter-entity dependencies without relying on predefined graph structures, exhibits strong potential for application in broader cross-entity time series prediction tasks beyond financial domains. Specifically, scenarios such as multi-sensor fusion in IoT environments and traffic flow forecasting across interconnected road networks share structural similarities with multi-stock time series, where interactions among entities evolve dynamically and may be nonlinear and nonstationary. The learnable attention-based mapping mechanism within NoGraphMixer can effectively capture latent inter-entity relationships, making it well-suited for modeling dynamic dependencies in multi-sensor systems, environmental monitoring, and urban traffic prediction without the need for explicit prior graphs that are often infeasible or expensive to obtain in such contexts. Future research will extend NoGraphMixer to these cross-entity time series scenarios, assessing its capacity to improve predictive accuracy, stability, and computational efficiency in real-world non-financial applications, thereby broadening the practical impact of the proposed architecture.

Future research will focus on further expanding the model's practicality and generalization ability. On one hand, more dimensional heterogeneous information can be introduced to further enrich the input feature space; on the other hand, the model could be extended to a multi-task framework to achieve joint modeling of various market indicators such as price change, volatility, and turnover rate. At the same time, improving the model's interpretability and visualization capabilities will also be an important research direction in the future, providing more practical support for quantitative trading and intelligent investment research.

Data availability

The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request. Correspondence should be addressed to Shengrui Liu at 18045830330@163.com.

Received: 13 May 2025; Accepted: 4 August 2025

Published online: 19 August 2025

References

1. Ye, H., Chen, J., Gong, S., et al. ATFNet: Adaptive Time-Frequency Ensembled Network for Long-term Time Series Forecasting. arXiv preprint [arXiv:2404.05192](https://arxiv.org/abs/2404.05192), (2024).

2. Hui, Y., Zwetsloot, I. M., Trimborn, S., et al. Domain-Informed Negative Sampling Strategies for Dynamic Graph Embedding in Meme Stock-Related Social Networks. arXiv preprint [arXiv:2411.00606](https://arxiv.org/abs/2411.00606), (2024).
3. Chen, Y. C. & Huang, W. C. Constructing a stock-price forecast CNN model with gold and crude oil indicators. *Appl. Soft Comput.* **112**, 107760 (2021).
4. Fang, W., Chen, Y. & Xue, Q. Survey on research of RNN-based spatio-temporal sequence prediction algorithms. *J. Big Data* **3**(3), 97 (2021).
5. Xiao, X., Hua, X. & Qin, K. A self-attention based cross-sectional return forecasting model with evidence from the Chinese market. *Financ. Res. Lett.* **62**, 105144 (2024).
6. Muhammad, T. et al. Transformer-based deep learning model for stock price prediction: A case study on Bangladesh stock market. *Int. J. Comput. Intell. Appl.* **22**(03), 2350013 (2023).
7. Patel, M., Jariwala, K. & Chattopadhyay, C. A Systematic Review on Graph Neural Network-based Methods for Stock Market Forecasting. *ACM Comput. Surv.* **57**(2), 1–38 (2024).
8. Tan, F. A frequency-domain approach to dynamic macroeconomic models. *Macroecon. Dyn.* **25**(6), 1381–1411 (2021).
9. Esmalifalak, H. & Moradi-Motlagh, A. Correlation networks in economics and finance: A review of methodologies and bibliometric analysis. *J. Econ. Surv.* **39**, 1252–1286 (2024).
10. Kumar, R. et al. Spatio-Temporal Predictive Modeling Techniques for Different Domains: a Survey. *ACM Comput. Surv.* **57**(2), 1–42 (2024).
11. Keil, A. et al. Recommendations and publication guidelines for studies using frequency domain and time-frequency domain analyses of neural time series. *Psychophysiology* **59**(5), e14052 (2022).
12. Yemets, K., Izonin, I. & Dronyuk, I. Time Series Forecasting Model Based on the Adapted Transformer Neural Network and FFT-Based Features Extraction. *Sensors* **25**(3), 652 (2025).
13. Qiu, W., Yin, H., Dong, Y., et al. Synchro-waveform-based event identification using multi-task time-frequency transform networks. *IEEE Transactions on Smart Grid*, (2025).
14. Sairam, S. D. & Kumar, D. S. *Fusion Neural Networks for High Precision Design and Ultra-Wideband Shielding in Frequency Selective Surfaces* (IEEE Transactions on Components, Packaging and Manufacturing Technology, 2024).
15. Hu, D. et al. FECT-OSA: A transformer-enhanced multimodal system for non-contact sleep apnea monitoring. *Alex. Eng. J.* **128**, 628–641 (2025).
16. Zhou, Y. et al. An interference mitigation method for FMCW radar based on time-frequency distribution and dual-domain fusion filtering. *Sensors* **24**(11), 3288 (2024).
17. Lee, C. Y., Hasegawa, H. & Gao, S. Complex-valued neural networks: A comprehensive survey. *IEEE/CAA J. Autom. Sinica* **9**(8), 1406–1426 (2022).
18. Lv, K. et al. A predictive analytics framework for rolling bearing vibration signal using deep learning and time series techniques. *Comput. Electr. Eng.* **117**, 109314 (2024).
19. Shah, S. M. A. H. et al. A hybrid neuro-fuzzy approach for heterogeneous patch encoding in ViTs using contrastive embeddings and deep knowledge dispersion. *IEEE Access* **11**, 83171–83186 (2023).
20. Zhang, J. et al. Convolution transformer mixer for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **19**, 1–5 (2022).
21. Wang, Z. et al. EmbedFormer: embedded depth-wise convolution layer for token mixing. *Sensors* **22**(24), 9854 (2022).
22. Yang, Y., Yu, J., Fu, Z., et al. Token-mixer: Bind image and text in one embedding space for medical image reporting. *IEEE Transactions on Medical Imaging*, (2024).
23. Fan, J. & Shen, Y. StockMixer: a simple yet strong MLP-based architecture for stock price forecasting//Proceedings of the AAAI Conference on Artificial Intelligence. **38**(8), 8389–8397 (2024).
24. Shen, Y. et al. Graph neural networks for wireless communications: From theory to practice. *IEEE Trans. Wireless Commun.* **22**(5), 3554–3569 (2022).
25. Qazi, E. U. H., Almorjan, A. & Zia, T. A one-dimensional convolutional neural network (1D-CNN) based deep learning system for network intrusion detection. *Appl. Sci.* **12**(16), 7986 (2022).
26. Dantas, P. V. et al. A comprehensive review of model compression techniques in machine learning. *Appl. Intell.* **54**(22), 11804–11844 (2024).
27. Zhang, L., Aggarwal, C. & Qi, G. J. Stock price prediction via discovering multi-frequency trading patterns//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2141–2149 (2017).
28. Qin, Y., Song, D., Chen, H., et al. A dual-stage attention-based recurrent neural network for time series prediction. arXiv preprint [arXiv:1704.02971](https://arxiv.org/abs/1704.02971), 2017.
29. Chen, Y., Wei, Z. & Huang, X. Incorporating corporation relationship via graph convolutional neural networks for stock price prediction//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018: 1655–1658.
30. Feng, F. et al. Temporal relational ranking for stock prediction. *ACM Trans. Inf. Syst. (TOIS)* **37**(2), 1–30 (2019).
31. Kim, R., So, C. H., Jeong, M., et al. HATS: A hierarchical graph attention network for stock movement prediction. arXiv preprint [arXiv:1908.07999](https://arxiv.org/abs/1908.07999), (2019).
32. Sawhney, R., Agarwal, S., Wadhwa, A., et al. Spatiotemporal hypergraph convolution network for stock movement forecasting//2020 IEEE International Conference on Data Mining (ICDM). IEEE, 482–491 (2020).
33. Cui, C. et al. Temporal-relational hypergraph tri-attention networks for stock trend prediction. *Pattern Recogn.* **143**, 109759 (2023).
34. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997).
35. Feng, F., Chen, H., He, X., et al. Enhancing stock movement prediction with adversarial training. arXiv preprint [arXiv:1810.09936](https://arxiv.org/abs/1810.09936), (2018).
36. Zhu, L. et al. Dyadic relational graph convolutional networks for skeleton-based human interaction recognition. *Pattern Recogn.* **115**, 107920 (2021).
37. Velickovic, P., Cucurull, G., Casanova, A., et al. Graph attention networks. *stat.* **1050**(20), 10–48550 (2017).
38. Sawhney, R., Agarwal, S., Wadhwa, A., et al. Stock selection via spatiotemporal hypergraph attention network: A learning to rank approach//Proceedings of the AAAI Conference on Artificial Intelligence. **35**(1), 497–504 (2021).
39. Huynh, T. T., Nguyen, M. H., Nguyen, T. T., et al. Efficient integration of multi-order dynamics and internal dynamics in stock movement prediction//Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining. 850–858 (2023).
40. Uras, N. et al. Forecasting Bitcoin closing price series using linear regression and neural networks models. *Peer J. Comput. Sci.* **6**, e279 (2020).

Author contributions

W.S. conceived and designed the study. J.M. and J.Z. conducted the experiments and collected the data. S.L. implemented the model and performed the analysis. C.Y. assisted in result interpretation and manuscript editing. All authors reviewed and approved the final manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to S.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025