
STOCK PREDICTION VIA A DUAL RELATION FUSION NETWORK INCORPORATING STATIC AND DYNAMIC RELATIONS

Long Chen¹ **Huixin Bai^{*1}** **Mingxin Wang¹** **Xiaohua Huang²** **Ying Liu²** **Jie Zhao³**
Ziyu Guan⁴

¹Shaanxi Key Laboratory of Information Communication Network and Security, Xi'an University of Posts and Telecommunications, Xi'an, China

²School of Information Science and Technology, Northwest University, Xi'an, China

³School of Data Science and Artificial Intelligence, Chang'an University, Xi'an, China

⁴School of Computer Science and Technology, Xidian University, Xi'an, China

^{*}Corresponding author

Correspondence:

{chenlong0, 05221170@stu., Mingxin Wang@stu.}xupt.edu.cn,
{huangxiaohua, liuying}@stumail.nwu.edu.cn,
jiezhao@chd.edu.cn,
zyguan@xidian.edu.cn

ABSTRACT

Accurate modeling of inter-stock relationships is critical for stock price forecasting. However, existing methods predominantly focus on single-state relationships, neglecting the essential complementarity between dynamic and static inter-stock relations. To solve this problem, we propose a Dual Relation Fusion Network (DRFN) to capture the long-term relative stability of stock relation structures while retaining the flexibility to respond to sudden market shifts. Our approach features a novel relative-static relation component that models time-varying long-term patterns and incorporates overnight informational influences. We capture dynamic inter-stock relationships through distance-aware mechanisms, while evolving long-term structures via recurrent fusion of dynamic relations from the prior day with the pre-defined static relations. Experiments demonstrate that our method significantly outperforms the baselines across different markets, with high sensitivity to the co-movement of relational strength and stock price. Code: <https://github.com/bhx-E/DRFN>.

1 Introduction

Reliable stock price predictions facilitate investors in making profitable investment decisions and aid financial regulators in identifying potential financial risks. Early research on stock prediction focused on enriching the diversity of input information by introducing more market indicators [1] or combining texts and indicators [2, 3]. The subsequent studies focus on learning the temporal patterns of time-series historical data [4, 5]. However, these studies often overlook inter-stock relationships, which are critical drivers of stock price fluctuations. Accurately modeling these relationships in real-world markets remains challenging due to their inherent complexity and diversity. Specifically, they span: industry-sector correlations, vertical supply chain linkages, competitive dynamics, and cross-asset contagion effects. Existing studies learning inter-stock correlations typically adopt single-regime models, focusing exclusively on short-term fluctuations [6] or long-term dependencies [7]. This leads to dual limitations: purely static models fail to capture evolving relations, while solely dynamic approaches may undermine model stability. During significant events (e.g., global crises or policy shifts), previously trained models often fail abruptly as relationships shift beyond their adaptive capacity. To this end, dynamic modeling frameworks incorporate static structural anchors to mitigate volatility. He et al. [8] proposed a static-dynamic graph neural network for stock recommendations. They used a data-driven graph learner to capture static/dynamic stock relations and a graph interactor to fuse multi-graph information, followed by graph convolution and ranking modules for stock recommendations. However, this study omits three critical issues: (1) News

impact neglect: The effect of news on both short- and long-term stock relationships is unaddressed; (2) Time-varying structure: Stock relations evolve dynamically with new information (e.g., Apple-OpenAI’s competitive-to-collaborative structural shift through strategic AI integration); and (3) Nonlinear interaction: Dynamic and static relationships exhibit complex, uncertain couplings, and simple concatenation cannot capture market complexity.

To tackle these critical shortcomings, we propose a Dual-Relation Fusion Network for stock price prediction. The key idea is to elucidate the complementarity between static and dynamic relationships, thereby enabling the model to maintain stable relationship patterns acquired over time and to rapidly adapt to abrupt changes. Specifically, to address market inefficiencies, behavioral biases, and lagged event responses — all phenomena requiring complementary multi-source information — we align news and market indicator embeddings via contrastive learning, followed by bilinear fusion. This captures the nuanced interactions between qualitative sentiment and quantitative signals, yielding robust stock representations with an enhanced ability to anticipate price movements driven by market inefficiencies. Next, we design a novel relation fusion module to capture static-dynamic complementarity through three core components: 1) Dynamic Relation Learning: Model synergistic (positive) and adversarial (negative) correlations via cross-attention with distance modulation, capturing short-term mutual influences between stocks. 2) Relative Static Relation modeling: Enhance predefined static relationships by fusing prior-day dynamic features with weighted static features, capturing time-varying long-term structures and overnight information impacts. 3) Dynamic-Relative Static Fusion: Fuse dynamic and relative-static features using bidirectional cross-attention. After relation modeling, we integrate the learned relations with the original news and market embeddings via residual connections to predict the next-day stock returns.

The main contributions are as follows: 1) Dual Relation Fusion framework: Integrates short-term volatility dynamics with relative long-term dependencies, simultaneously capturing structural stability while maintaining market responsiveness. 2) Distance-Modulated Correlation Modeling: Quantifies synergistic-adversarial stock dynamics through distance-aware mechanisms to capture time-varying inter-stock patterns. 3) Adaptive Static Relation Construction: Models evolve long-term structures via recurrent fusion of prior-day dynamic relations with the pre-defined static relations.

2 Related Work

Earlier studies have used historical data and technical indicators of individual stocks for predictions [9, 10]. Li et al. [2] used online news text and stock price time-series market indicators as input, and they proposed a multimodal event-driven LSTM model to perform the interaction and integration of the multimodal inputs to enhance the performance of stock price prediction. However, these studies ignored correlations between stocks and deemed them independent entities, making it difficult to capture their interactive influence on stock predictions. Subsequent studies have recognized this issue and developed stock prediction methods based on relational modeling. Graph models are widely used in relationship modeling. Ying et al. [11] designed a time-aware relational attention mechanism to capture the time-varying correlation strength between stocks. However, the use of predefined relations lacks an effective mechanism for handling attribute-related influences. The method in [12] infers latent firm relations from market signals and, with its attribute-driven graph attention network, captures attribute-sensitive momentum spillovers and complex market signal interactions. Huynh et al. [13] attempted to capture more sophisticated dynamic relations using industry hypergraphs, wavelet hypergraph convolutions, and temporal generative filters to capture multi-order and internal dynamic relationships. Relation modeling is further improved by capturing momentary and cross-time stock correlations through intra- and inter-stock information aggregation, using a gating mechanism for automatic feature selection based on market information [5]. However, this may not fully capture the long-term invariant relationships between stocks. LSR-IGRU [6] constructed long-short-term relationship matrices using secondary industry and overnight price information, enhancing GRU input to better integrate temporal and relationship information. Extending LSR-IGRU, we propose DRFN to capture persistent relational structures while maintaining flexibility in responding to sudden market shifts.

3 Method

We predict the next-day returns for a set of Z input stocks, formulated as a regression problem. The daily return is computed as the ratio of the interday closing price differential to the prior trading day’s closing price: $r_{t+1} = \frac{c_{t+1} - c_t}{c_t}$. For stock i , the input data $D = \{D_i^\theta\}_{\theta=t-T+1}^t$ contains news texts and market indicators from the past T days; Specifically, on day t , D_i^t comprises Q news texts $\{n_{i,q}^t\}_{q=1}^Q$ and six market indicators: open (p_o^t), close (p_c^t), adjusted close (p_{ac}^t), high (p_h^t), low (p_l^t), and volume (p_v^t), which form the vector $\mathbf{p}_i^t = [p_o^t, p_c^t, p_{ac}^t, p_h^t, p_l^t, p_v^t]$.

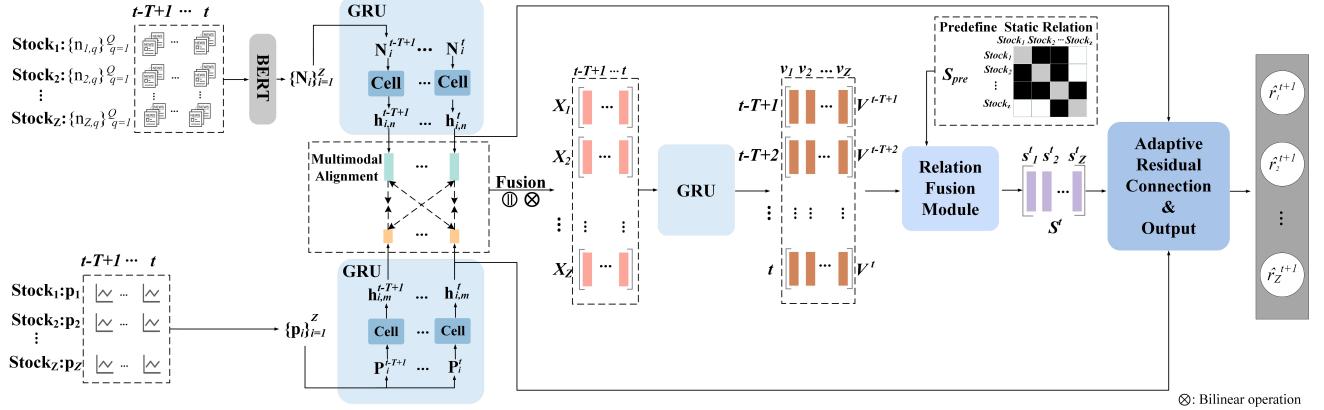


Figure 1: Overview of our framework.

The DRFN framework is illustrated in Figure 1. First, we encode the input textual news and market indicator data using BERT&GRU and GRU, respectively. The obtained temporal hidden states from both modalities are then aligned and fused to generate multimodal fusion vectors. Next, the fusion vectors are fed into the GRU module for extracting temporal patterns. Subsequently, the relation fusion module models long-term static and short-term dynamic inter-stock relations using the temporal multimodal features and the predefined static relations. Finally, the relation features are combined with the residual-propagated temporal hidden states derived from both textual news and market indicators to form unified representations. These representations are then transformed by a nonlinear mapping function for stock price predictions.

3.1 News Text Encoding

News sentiment drives investor behavior, triggering stock price changes. More importantly, news shapes stock correlations. For instance, supply chain news affects upstream and downstream companies; major positive news for one firm can negatively impact its competitors. We encode each news text using a pre-trained BERT model as follows:

$$\mathbf{n}_{i,q}^\theta = \text{BERT}(n_{i,q}^\theta) \in \mathbb{R}^L \quad (1)$$

The Q encoded textual news embeddings in one day are represented by the matrix $\mathbf{N}_i^\theta = [\mathbf{n}_{i,1}^\theta, \mathbf{n}_{i,2}^\theta, \dots, \mathbf{n}_{i,Q}^\theta]$. We then use a GRU to learn temporal patterns from all news over the T -day period:

$$\mathbf{h}_{i,n}^\theta = \text{GRU}(\mathbf{N}_i^\theta, \mathbf{h}_{i,n}^{\theta-1}) \in \mathbb{R}^L \quad (2)$$

where $\mathbf{h}_{i,n}^\theta$ represents the temporal hidden state of news embeddings at time step $\theta \in [t - T + 1, t]$.

3.2 Market Indicator Encoding

Market indicators capture market dynamics, including trends and volatility. We model the market indicator dynamics using a separate GRU:

$$\mathbf{h}_{i,m}^\theta = \text{GRU}(\mathbf{P}_i^\theta, \mathbf{h}_{i,m}^{\theta-1}) \in \mathbb{R}^{L'} \quad (3)$$

where $\mathbf{h}_{i,m}^\theta$ denotes the temporal hidden states for the market indicators over the T -day time window.

3.3 Multimodal Alignment and Fusion

Market indicators and textual news exhibit strong complementarity, primarily grounded in Behavioral Finance. Investor decisions are systematically influenced by sentiment propagated through news narratives. Consequently, while quantitative market indicators reflect collective trading actions and outcomes, textual news reveals the qualitative narratives, sentiments, and behavioral drivers underpinning these actions. This provides a critical qualitative context that is absent from the quantitative data alone. For Z stocks, we apply contrastive learning with InfoNCE loss [14] to

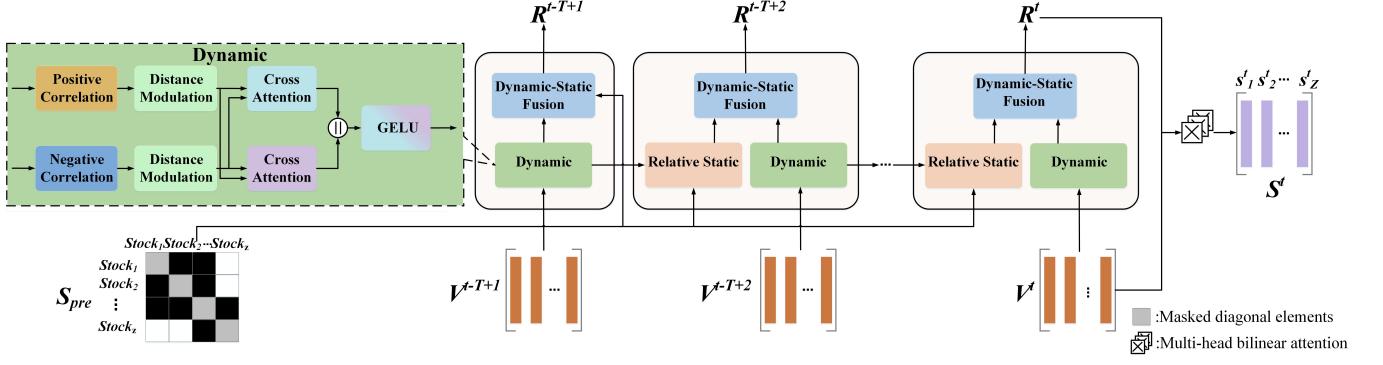


Figure 2: Relation Fusion Module

align the temporal hidden states of the market indicators and textual news:

$$\mathcal{L}_{\text{Align}} = -\frac{1}{T} \cdot \frac{1}{Z} \sum_{\theta=t-T+1}^t \sum_{i=1}^Z \log \frac{\exp(\text{sim}(\mathbf{h}_{i,n}^\theta, \mathbf{h}_{i,m}^\theta)/\tau)}{\sum_{j=1}^Z \exp(\text{sim}(\mathbf{h}_{i,n}^\theta, \mathbf{h}_{j,m}^\theta)/\tau)} \quad (4)$$

where $\text{sim}(\cdot)$ is the cosine similarity function; τ denotes the temperature hyperparameter. Cross-modal alignment reduces the distance between these two modalities within the same stock, while increasing the distance across stocks between one stock’s market indicator features and another stock’s textual news features.

Following alignment, we use concatenation and weighted bilinear transformation to fuse the aligned cross-modal features and exploit their information complementarity:

$$\mathbf{x}_i^\theta = \tanh \left(\mathbf{W}_i [\mathbf{h}_{i,m}^\theta \| \mathbf{h}_{i,n}^\theta] + \mathbf{h}_{i,m}^\theta \mathbf{T}_i^{[1:K]} \mathbf{h}_{i,n}^\theta + \mathbf{b}_i \right) \quad (5)$$

where $\mathbf{T}_i^{[1:K]} \in \mathbb{R}^{L \times L' \times K}$ and $\mathbf{W}_i \in \mathbb{R}^{K \times (L+L')}$ are weight matrices, \mathbf{b}_i is the bias term, and these parameters are shared across all time steps. $[\cdot \| \cdot]$ represents the concatenation operation. Note that we perform this fusion across all T days. In fusion, concatenation preserves the original feature integrity, whereas the K -dimensional bilinear tensor product enables a structured cross-modal interaction. The k -th entry is computed using one tensor slice. Bilinear operations surpass linear combinations in modeling complex market phenomena. For instance, the impact of news sentiment on stock prices is often nonlinear, exerting greater influence beyond certain intensity thresholds, a relationship effectively captured by bilinear fusion.

3.4 Temporal Pattern Learning

After multimodal fusion, the sequence of fused vectors $\{\mathbf{x}_i^\theta\}_{\theta=t-T+1}^t$ is fed into a GRU, which outputs all hidden states:

$$\mathbf{v}_i^\theta = \text{GRU} (\{\mathbf{x}_i^\theta\}_{\theta=t-T+1}^t) \in \mathbb{R}^F \quad (6)$$

where F denotes the hidden size of the GRU. This module captures cross-modal temporal collaborative patterns by dynamically adjusting memory retention through its update gate, focusing on the short-term impacts of news events while retaining long-term periodic patterns in market dynamics.

3.5 Relation Fusion Module

Inter-stock relationships are not fixed and are subject to dynamic changes due to macroeconomic factors, industry, market sentiment, and other factors. To capture this complementarity, we propose a novel relation fusion module that combines short-term dynamic relationships with long-term relatively static relationships. The module structure is shown in Figure 2. It comprises five components: Predefined Static Relations, Dynamic Relation Learning (the zoomed-in callout in Figure 2), Relative-Static Relation Modeling, Dynamic-Relative Static Fusion, and Relational-Temporal Interactive Fusion.

Predefined Static Relations. Predefined static relationships are long-term, stable business relationships that exist among firms, derived from known market structures and enterprise connections. The predefined static relation matrix \mathbf{S}_{pre} is constructed from company entities and relationships provided by Wikidata [15]. It is a binary matrix where 1 denotes related stocks and 0 denotes unrelated stocks, serving as the static relation structure over the past T -day time window. We apply a diagonal mask to \mathbf{S}_{pre} to exclude self-correlation.

Dynamic Relation Learning. Stock correlations evolve dynamically under external influences. For instance, the new energy and traditional energy sectors may shift from a positive correlation during policy stimuli to a negative correlation during capital reconfiguration. To capture these time-varying inter-stock dynamics, we integrate dual-correlation modeling with distance modulation:

$$\mathbf{A}^{\theta,(a)} = \frac{\mathbf{V}^\theta (a \cdot \mathbf{V}^\theta)^\top}{\sqrt{F}} \in \mathbb{R}^{Z \times Z} \quad (7)$$

$$D_{ij}^{\theta,(a)} = \|\mathbf{v}_i^\theta - a \cdot \mathbf{v}_j^\theta\|_1 \in \mathbb{R}^{Z \times Z} \quad (8)$$

$$\tilde{\mathbf{A}}^{\theta,(a)} = \mathbf{A}^{\theta,(a)} \odot \exp(-\mathbf{D}^{\theta,(a)}) \quad (9)$$

where $\mathbf{V}^\theta = [\mathbf{v}_1^\theta, \mathbf{v}_2^\theta, \dots, \mathbf{v}_Z^\theta] \in \mathbb{R}^{Z \times F}$ denotes the hidden state matrix from GRU. The attention matrix $\mathbf{A}^{\theta,(a)}$ models the relational structures among stocks at every time, with the scalar $a \in \{+1, -1\}$ switching between positive and negative correlation modes. The term \sqrt{F} serves as a scaling factor to normalize the attention scores. $\mathbf{D}^{\theta,(a)}$ quantifies pairwise behavioral discrepancies, while $\tilde{\mathbf{A}}^{\theta,(a)}$ fuses directional alignment and distance-based modulation via Hadamard product \odot .

$$\mathbf{F}^{\theta,(a)} = \text{softmax} \left(\frac{\tilde{\mathbf{A}}^{\theta,(a)} \cdot (\tilde{\mathbf{A}}^{\theta,(-a)})^\top}{\sqrt{Z}} \right) \cdot \tilde{\mathbf{A}}^{\theta,(-a)} \quad (10)$$

$$\mathbf{D}_{dynamic}^\theta = \text{GELU} \left([\mathbf{F}^{\theta,(+1)} \| \mathbf{F}^{\theta,(-1)}] \cdot \mathbf{W}_d \right) \quad (11)$$

Next, cross-attention in Eq. 10 performs cross-perspective fusion by: 1) Identifying stock pairs exhibiting strong positive but weak negative correlations (or vice versa) through interactions between $\tilde{\mathbf{A}}^{\theta,(a)}$ and $\tilde{\mathbf{A}}^{\theta,(-a)}$; 2) Assigning cross-perspective importance weights via softmax to highlight dominant association patterns. Then a nonlinear projection with a learnable parameter matrix $\mathbf{W}_d \in \mathbb{R}^{2Z \times Z}$ integrates the positive and negative correlation fusion matrices $\mathbf{F}^{\theta,(+1)}$ and $\mathbf{F}^{\theta,(-1)}$ to produce the dynamic relation matrix $\mathbf{D}_{dynamic}^\theta \in \mathbb{R}^{Z \times Z}$, which dynamically evolves with market conditions.

Relative-Static Relation Modeling. Overnight information is released intensively after the opening of the next trading day and has a significant impact on stock price movements [16]. Nevertheless, predefined static relations solely capture long-term relation patterns while neglecting this ultra-short-term information. To address this limitation, we integrate the predefined static relations with the dynamic relation from the preceding day, resulting in a final static relation termed the relative-static relation. Specifically, we employ linear weighting to combine the dynamic relation of the previous time step $\mathbf{D}_{dynamic}^{\theta-1} \in \mathbb{R}^{Z \times Z}$ and the predefined static relation $\mathbf{S}_{pre} \in \mathbb{R}^{Z \times Z}$. The computational formula is as follows:

$$\mathbf{S}_{relative}^\theta = \alpha \cdot \mathbf{D}_{dynamic}^{\theta-1} + (1 - \alpha) \cdot [\mathbf{W}_s \odot \mathbf{S}_{pre}] \quad (12)$$

where $\mathbf{S}_{relative}^\theta \in \mathbb{R}^{Z \times Z}$ is the output at time step θ . We add a learnable parameter matrix \mathbf{W}_s that is reused across all time steps to dynamically adjust \mathbf{S}_{pre} . This adjustment enables the predefined static relation to capture the time-varying characteristics of the relationship strength among stocks while maintaining a stable long-term relation structure. Besides, we introduce a learnable scalar parameter α that determines the relative contributions of dynamic and static relationships. Note that at the initial time step $t - T + 1$, \mathbf{S}_{pre} serves as the static relation component since prior dynamic relations are unavailable.

Dynamic-Relative Static Fusion. To capture the complementarity between dynamic and static relationships, we design a dual-relation fusion mechanism that generates comprehensive representations reflecting both short-term fluctuations and long-term stability. Real market conditions reveal no fixed proportionality between these relationships; instead, they present complex and uncertain interactions. Therefore, we implement bidirectional cross-attention for adaptive fusion, enabling dynamic-static integration through bidirectional information flow:

$$\begin{aligned} \mathbf{R}^\theta = \text{GELU} & \left(\left[\text{Attention}(\mathbf{S}_{relative}^\theta, \mathbf{D}_{dynamic}^\theta, \mathbf{D}_{dynamic}^\theta) \right. \right. \\ & \left. \left. \| \text{Attention}(\mathbf{D}_{dynamic}^\theta, \mathbf{S}_{relative}^\theta, \mathbf{S}_{relative}^\theta) \right] \cdot \mathbf{W}_{DRS} \right) \end{aligned} \quad (13)$$

$$\text{Attention}(Q, K, V) = \text{Softmax} \left(\frac{QK^\top}{\sqrt{d_k}} \right) V \quad (14)$$

where \mathbf{R}^θ is the fused representation at step θ , and $\mathbf{W}_{DRS} \in \mathbb{R}^{2Z \times Z}$ is a learnable weight matrix. Unlike traditional fusion methods, our bidirectional cross-attention mechanism adaptively controls the degree of fusion between relationships through its inherent adaptability, thereby effectively exploiting their complementary synergy.

Relational-Temporal Interactive Fusion. Building on the stock relations, we further fuse their temporal multimodal features by applying a multi-head bilinear attention mechanism to the relation matrix \mathbf{R}^t and the temporal multimodal feature matrix \mathbf{V}^t :

$$\mathbf{S}^t = \prod_{b=1}^B \tanh \left(\left[\mathbf{R}^{t,[b]} \cdot (\mathbf{V}^t \mathbf{W}_{TR}^{[b]}) \right] \right) \in R^{Z \times BF'} \quad (15)$$

where $\mathbf{S}^t = [\mathbf{s}_1^t, \mathbf{s}_2^t, \dots, \mathbf{s}_Z^t]$ denotes the final fused representation at time step t , the terminal point of the T -day time window. $\mathbf{W}_{TR}^{[b]} \in R^{F' \times F'}$ denotes the interaction weight matrix of the b -th attention head. This attention mechanism guides the model in learning the interaction between relationships and time by exploring multiple feature subspaces. For instance, a stock may be competitive or complementary to other stocks, and these relationships may be reflected in time series as synergies or differences in characteristics such as stock price volatility and trading volume changes.

3.6 Adaptive Residual Connection and Output

To capture the immediate impact of news and indicators, we employ residual connections and adaptively fuse temporal embeddings with relation features to generate a unified representation. The fused representation is subsequently transformed via a nonlinear projection to predict stock returns. The final output is computed as:

$$\hat{r}_i^{t+1} = \tanh \left(\mathbf{W}_o \cdot [(\mathbf{s}_i^t)^\top + \alpha_i^{(1)} \cdot (\mathbf{W}_n \cdot (\mathbf{h}_{i,n}^t)^\top + \mathbf{b}_n) + \alpha_i^{(2)} \cdot (\mathbf{W}_m \cdot (\mathbf{h}_{i,m}^t)^\top + \mathbf{b}_m)] + b_o \right) \quad (16)$$

where $\mathbf{W}_o \in \mathbb{R}^{1 \times BF'}$, $\mathbf{W}_n \in \mathbb{R}^{BF' \times L}$, and $\mathbf{W}_m \in \mathbb{R}^{BF' \times L'}$ denote weight matrices, with corresponding biases $b_o \in \mathbb{R}$, $\mathbf{b}_n \in \mathbb{R}^{BF'}$, and $\mathbf{b}_m \in \mathbb{R}^{BF'}$. The scalars $\alpha_i^{(1)}$ and $\alpha_i^{(2)}$ are learnable parameters. This adaptive fusion addresses two key challenges: 1) Feature Heterogeneity: Input features are diverse, making simple concatenation noisy. 2) Dynamic Sensitivity: Stocks exhibit varying sensitivities (e.g., policy-sensitive stocks rely more on news, while energy stocks focus more on crude oil prices). Learnable weights enable dynamic parameter adjustment, thus balancing the contributions of different information sources.

3.7 Objective Function

We combine Mean Squared Error (MSE) loss and cross-modal alignment loss as the overall objective function:

$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \lambda \mathcal{L}_{\text{Align}} \quad (17)$$

$$\mathcal{L}_{\text{MSE}} = \frac{1}{Z} \sum_{i=1}^Z (r_i^{t+1} - \hat{r}_i^{t+1})^2 \quad (18)$$

where r_i^{t+1} and \hat{r}_i^{t+1} denote the ground-truth and predicted returns of stock i at time $t + 1$, respectively, and λ is a balancing hyperparameter.

4 Experiments

4.1 Dataset and Experiment Settings

We use datasets from China (CMIN-CN) and the US (CMIN-US) markets [17] for evaluation. CMIN-US contains 105 stocks over 1,006 trading days; CMIN-CN comprises 227 stocks across 968 days¹. Both datasets are chronologically split 8:1:1 into training, validation, and test sets. We use the Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) as evaluation metrics. *Due to market volatility, each baseline is trained 10 times with random initialization, and the average performance is reported.*

¹Samples from missing trading days are removed.

Model	Relation	Text	Price	CMIN-US		CMIN-CN	
				RMSE (Avg.)	MAE (Avg.)	RMSE (Avg.)	MAE (Avg.)
LSTM	✗	✓	✓	0.0232	0.0180	0.0359	0.0285
GRU	✗	✓	✓	0.0230	0.0177	0.0356	0.0277
GCN	✓	✓	✓	0.0221	0.0168	0.0347	0.0264
GAT	✓	✓	✓	0.0217	0.0164	0.0343	0.0257
MASTER	✓	✗	✓	0.0205	0.0156	0.0331	0.0245
SAMBA	✗	✗	✓	0.0201	0.0153	0.0325	0.0237
LSR-IGRU	✓	✗	✓	0.0192	0.0141	0.0317	0.0228
DRFN	✓	✓	✓	0.0166	0.0117	0.0270	0.0191

Table 1: Main comparison results

Cross-market hyperparameter tuning produced identical configurations on both validation sets, demonstrating strong cross-market generalization. Data specifications: maximum daily news count $Q = 30$ (40 tokens/text); BERT-encoded text vectors $L = 768$ (GRU preserves dimension); GRU-encoded temporal vectors of market indicators $L' = 64$. Optimal model parameters: time window $T = 5$ ($T \in [2, 10]$), attention heads $B = 6$ ($B \in \{2, 4, 6, 8, 10\}$) using even counts for symmetric modeling. Dimensions: GRU $F = 128$ and attention $F' = 256$ (both $\in \{32, 64, 128, 256, 512\}$), balance parameter $\lambda = 1$ ($\lambda \in \{0.1, 0.5, 1, 1.5, 2\}$). The model was trained using the Adam optimizer with a learning rate of 2×10^{-3} , implemented on a single NVIDIA Tesla P100 GPU (16G), with a total runtime of approximately 11 hours for convergence.

4.2 Baselines and Main Comparisons

LSTM and GRU. Long Short-Term Memory (LSTM) [18] and Gated Recurrent Unit (GRU) [19] are classic sequence models widely used for time-series prediction tasks. We employ a 2-layer LSTM and 2-layer GRU as sequence-model-based baseline methods.

GAT. Graph Attention neTwork [20] assigns varying weights to different nodes within a neighborhood. We use GAT with two masked self-attention layers.

GCN. The Graph Convolutional Network [21] is commonly used to capture stock relations. We employ a GCN with two convolutional layers as a baseline. The attributes of adjacent nodes are linearly aggregated to the central node using normalized stock relationships.

SAMBA. [22] proposed SAMBA, a framework that combines the Mamba architecture and graph neural networks (GNNs). SAMBA employs a bidirectional Mamba module to capture long-term dependencies in historical price data and to learn inter-stock-feature dependencies via adaptive graph convolution.

MASTER. [5] proposed a Market-Guided Stock TransformER model to capture momentary and cross-time correlations among stocks using market indicators.

LSR-IGRU. [6] proposed an Improved GRU to capture the Long Short-term Relations of stocks. It uses secondary industry information and overnight price changes to construct long- and short-term stock relation representations, and then the improved GRU fuses the original input with these representations via concatenating and weighting unit hidden states.

DRFN. This is our method.

Note that LSTM and GRU use concatenation of technical indicators and text features as inputs. GCN and GAT employ indicator and text feature combinations as graph nodes and connect them with predefined static relations.

The comparison results for the two markets are similar, shown in Table 1. LSTM and GRU achieve the lowest performance because they capture only fundamental temporal patterns, which are insufficient for modeling underlying relationships in multimodal data. GAT and GCN surpass the two basic sequence models because graph-based architectures effectively model the stock relationships. In comparison, MASTER achieves a better performance by capturing fine-grained stock relations, both intra- and inter-stock, along with asymmetric and dynamic inter-stock correlations. SAMBA outperforms MASTER, due to its enhanced capture of long-term dependencies via bidirectional time-series modeling and its effective modeling of intra-stock feature dependencies through adaptive structural modeling. LSR-IGRU outperforms SAMBA through its incorporation of primary/secondary industry data and overnight prices in relation modeling. This approach effectively captures cross-industry dependencies and short-term price fluctuation

Model	CMIN-US		CMIN-CN	
	RMSE (Avg.)	MAE (Avg.)	RMSE (Avg.)	MAE (Avg.)
Static	0.0221	0.0162	0.0337	0.0250
Relative Static	0.0208	0.0151	0.0326	0.0240
Dynamic	0.0196	0.0145	0.0315	0.0233
w/o Modality Alignment	0.0191	0.0139	0.0310	0.0225
w/o adaptive weighted residual mechanism	0.0174	0.0123	0.0280	0.0204
DRFN	0.0166	0.0117	0.0270	0.0191

Table 2: Ablation study

patterns, yielding superior results through richer market-signal extraction. By leveraging complementary learning of static and dynamic stock relations, our method achieves optimal performance across both markets.

4.3 Relation Modeling Variants and Ablation Study

Considering the centrality of relation modeling to our framework, we evaluate the relation fusion module against its variants to demonstrate its effectiveness, with complementary ablation studies of other components. The results are shown in Table 2. Replacing the relation fusion module with Predefined Static Relations (*Static*) causes a significant performance deterioration. This demonstrates that stock relations are inherently dynamic. *Relative Static* employs the Relative Static Relating Modeling, outperforming *Static* by capturing time-varying patterns of the stocks’ relations in long-term trading ranges and incorporating overnight information. *Dynamic* (using only Dynamic Relation Learning) performs best among the three variants, proving that short-term dynamic stock relations correlate more strongly with price fluctuations than long-term static relations. Nevertheless, it consistently underperforms our static-dynamic fusion method because static structural anchors are essential in dynamic modeling frameworks to mitigate market volatility. Further ablation studies highlight modality alignment’s necessity and the direct influence of input embeddings on stock dynamics.

4.4 Relation Attention Sensitivity to Price Fluctuations

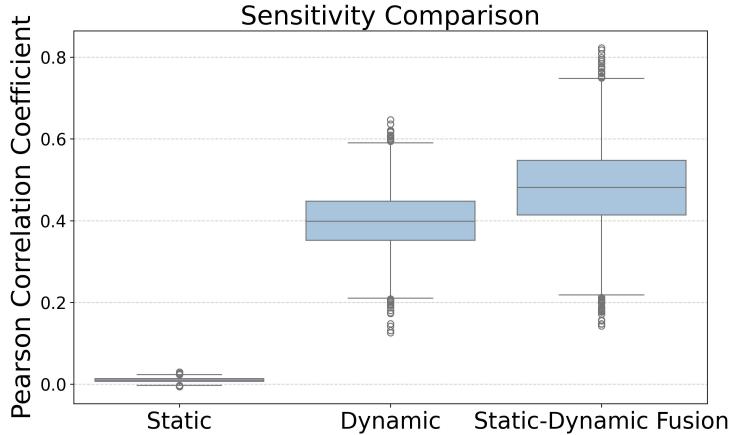


Figure 3: Sensitivity analysis

Modeling stock relationships is the main contribution of our work. We evaluate whether attention intensity between stocks in the relational module responds to price fluctuations. Specifically, we assess the sensitivity of three relational modeling approaches to market turbulence on the test set²: 1) the Static Relation method employed in GAT [20], 2) the Dynamic Relation Learning proposed in LSR-IGRU [6], and 3) our Static-Dynamic Fusion. To measure stock relationship sensitivity to price fluctuations, we compute the Pearson correlation between relational strength changes

²Given consistent cross-market comparison results, we analyze the full 90-trading-day test set of US market data in CMIN-US.

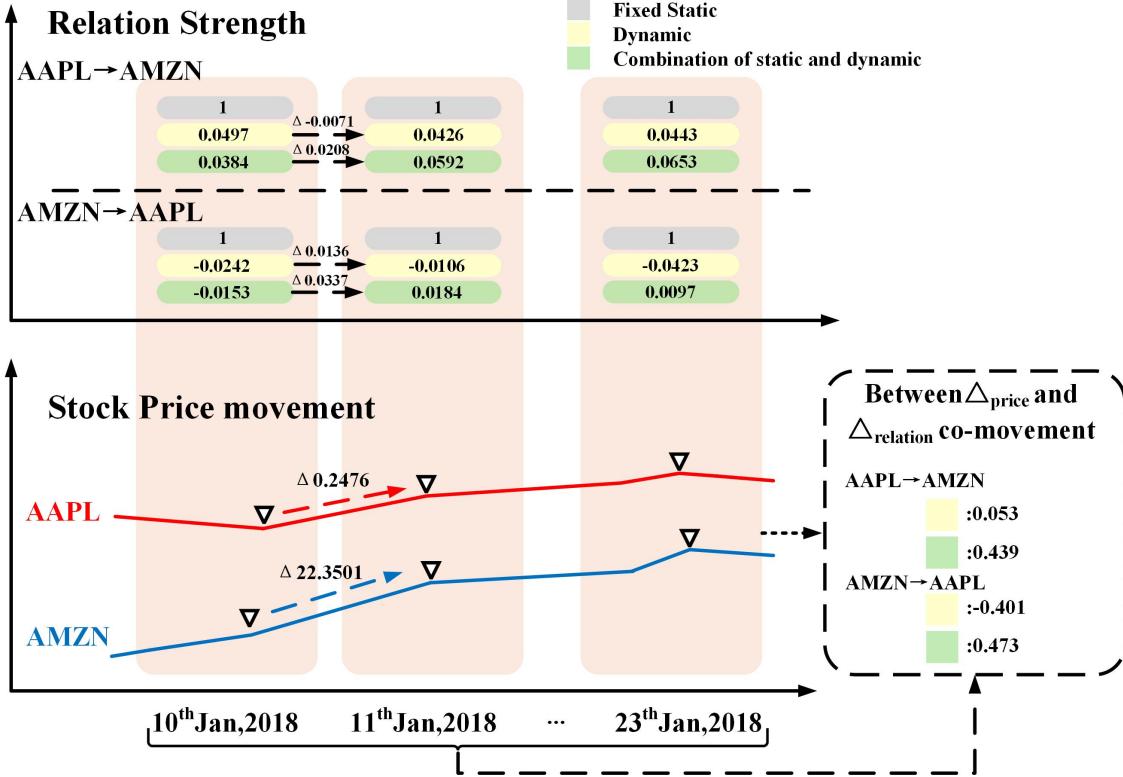


Figure 4: Comparative sensitivity of different relation modeling methods to AAPL and AMZN price movements

(ΔS) and price changes (ΔC):

$$\rho = \frac{\sum_{\theta=1}^{T'} (\Delta S^{(\theta)} - \bar{\Delta S})(\Delta C^{(\theta)} - \bar{\Delta C})}{\sqrt{\sum_{\theta=1}^{T'} (\Delta S^{(\theta)} - \bar{\Delta S})^2} \cdot \sqrt{\sum_{\theta=1}^{T'} (\Delta C^{(\theta)} - \bar{\Delta C})^2}} \quad (19)$$

where $\Delta S^{(\theta)}$ and $\Delta C^{(\theta)}$ are the predicted changes in relational strength and stock price from $\theta = 1$ to θ , with $\bar{\Delta S}$ and $\bar{\Delta C}$ their respective means over 90-day windows ($T' = 90$). The results are shown in Figure ???. Static relation modeling with predefined stock relations inherently lacks price sensitivity, owing to its fixed connectivity. While modeling dynamic stock relations adapts inter-stock connectivity to market fluctuations, its relational sensitivity to price volatility remains minimal. This is why LSR-IGRU underperforms our method because it models stock relationships through overnight price similarity and historical patterns but fails to capture real-time relationship dynamics driven by price fluctuations, a fundamental limitation that causes performance deterioration. Our Static-Dynamic Fusion module achieves superior sensitivity to relation-price co-movement. By recurrently integrating historical and current dynamics, we accumulate cross-period relations, thereby enhancing co-movement sensitivity.

We further compare the sensitivity of different relation modeling methods to market shifts by visualizing the relational strength and stock prices for AAPL and AMZN (January 10–23, 2018) in Figure ???. Long-term analysis reveals a weak correlation between the dynamic module and actual price fluctuations (coefficients: 0.053 and -0.401), particularly in directional consistency. In stark contrast, our Static-Dynamic Fusion module achieves significantly stronger correlations (0.439 and 0.473). A Short-term analysis of January 10–11 is a representative case: During synchronous AAPL-AMZN price surges, our Static-Dynamic Fusion module closely tracked price movements in both direction and amplitude, confirming strong short-term synergy; the dynamic module, however, failed to capture this co-movement. Building on our quantitative analysis, the case study visually illustrates enhanced relation-to-price sensitivity in our static-dynamic fusion module, achieved through temporal superposition effects of dynamic relations.

5 Conclusions

In this study, we propose a dual-relation fusion network to achieve complementary fusion between static and dynamic stock relations, resulting in the learning of the long-term relative stability of the stock relation structure while maintaining

the model's flexibility in responding to sudden market fluctuations. Our method demonstrates consistent superiority in comparative experiments, with the relation fusion module exhibiting enhanced sensitivity to temporal stock price fluctuations.

6 Acknowledgments

This paper is supported by the National Natural Science Foundation of China (62203354, 62472340, 6230336, 62133012, 62425605, 62306235, 62502051), the Key Research and Development Program of Shaanxi (Program Nos. 2024GX-YBXM-122, 22025CY-YBXM-041, 2024CY2-GJHX-15, 2022ZDLGY01-10), the Natural Science Basic Research Program of Shaanxi (Program Nos. 2025JC-QYXQ-040, 2025JC-YBQN-900, 2025CY-YBXM-067), Guangdong Basic and Applied Basic Research Foundation (Program No.2025A1515010267), the Postdoctoral Research Project Sustentation Fund of Shaanxi (No.2024BSHSDZZ112), the Postdoctoral Fellowship Program of CPSF (Grant No. GZC20251101) and the Key Scientific Research Projects of the Shaanxi Provincial Education Department (No. 24JR152).

References

- [1] Sidra Mehtab, Jaydip Sen, and Abhishek Dutta. Stock price prediction using machine learning and lstm-based deep learning models. In *Machine Learning and Metaheuristics Algorithms, and Applications: Second Symposium, SoMMA 2020, Chennai, India, October 14–17, 2020, Revised Selected Papers 2*, pages 88–106, Singapore, 2021. Springer, Singapore.
- [2] Qing Li, Jinghua Tan, Jun Wang, and Hsinchun Chen. A multimodal event-driven lstm model for stock prediction using online news. *IEEE Transactions on Knowledge and Data Engineering*, 33(10):3323–3337, 2020.
- [3] Ruize Gao, Shaoze Cui, Hongshan Xiao, Weigu Fan, Hongwu Zhang, and Yu Wang. Integrating the sentiments of multiple news providers for stock market index movement prediction: A deep learning approach based on evidential reasoning rule. *Information Sciences*, 615:529–556, 2022.
- [4] Burak Gülmez. Stock price prediction with optimized deep lstm network with artificial rabbits optimization algorithm. *Expert Systems with Applications*, 227:120346, 2023.
- [5] Tong Li, Zhaoyang Liu, Yanyan Shen, Xue Wang, Haokun Chen, and Sen Huang. Master: Market-guided stock transformer for stock price forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 162–170, Vancouver, 2024. AAAI.
- [6] Peng Zhu, Yuante Li, Yifan Hu, Qinyuan Liu, Dawei Cheng, and Yuqi Liang. Lsr-igru: Stock trend prediction based on long short-term relationships and improved gru. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 5135–5142, Boise ID USA, 2024. Association for Computing MachineryNew YorkNYUnited States.
- [7] Zhiluo Chen, Zeyu Huang, and Yukang Zhou. Predicting stock trend using gnn. *Highlights in Science, Engineering and Technology*, 39:816–822, 2023.
- [8] Yanshen He, Qiutong Li, Feng Wu, and Jianliang Gao. Static-dynamic graph neural network for stock recommendation. In *Proceedings of the 34th international conference on scientific and statistical database management*, pages 1–4, Copenhagen, Denmark, 2022. AAAI Press.
- [9] Jigar Patel, Sahil Shah, Priyank Thakkar, and Ketan Kotecha. Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert systems with applications*, 42(1):259–268, 2015.
- [10] David MQ Nelson, Adriano CM Pereira, and Renato A De Oliveira. Stock market's price movement prediction with lstm neural networks. In *2017 International joint conference on neural networks (IJCNN)*, pages 1419–1426, Anchorage, AK, USA, 2017. Ieee, ieee.
- [11] Xiaoting Ying, Cong Xu, Jianliang Gao, Jianxin Wang, and Zhao Li. Time-aware graph relational attention network for stock recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2281–2284, Online, 2020. ACM.
- [12] Rui Cheng and Qing Li. Modeling the momentum spillover effect for stock prediction via attribute-driven graph attention networks. In *Proceedings of the AAAI Conference on artificial intelligence*, volume 35, pages 55–62, Online, 2021. AAAI Press.

- [13] Thanh Trung Huynh, Minh Hieu Nguyen, Thanh Tam Nguyen, Phi Le Nguyen, Matthias Weidlich, Quoc Viet Hung Nguyen, and Karl Aberer. Efficient integration of multi-order dynamics and internal dynamics in stock movement prediction. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pages 850–858, Washington, D.C., USA, 2023. ACM.
- [14] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [15] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.
- [16] Dong Lou, Christopher Polk, and Spyros Skouras. A tug of war: Overnight versus intraday expected returns. *Journal of Financial Economics*, 134(1):192–213, 2019.
- [17] Di Luo, Weiheng Liao, Shuqi Li, Xin Cheng, and Rui Yan. Causality-guided multi-memory interaction network for multivariate stock price movement prediction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12164–12176, 2023.
- [18] S Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 9(8):1735–1780, 1997.
- [19] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.
- [20] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. Graph attention networks. *stat*, 1050(20):10–48550, 2017.
- [21] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, pages –, Toulon, 2017. ICLR.
- [22] Ali Mehrabian, Ehsan Hoseinzade, Mahdi Mazloum, and Xiaohong Chen. Mamba meets financial markets: A graph-mamba approach for stock price prediction. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.