FINKAN: KAN-BASED PREDICTIVE STRUCTURE LEARNING FOR STOCK FORECASTING UNDER MARKET SHIFTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Heterogeneous stock forecasting requires learning stable predictive structures from historical periods and generalizing to unseen future periods under market shifts. Existing distribution-aware, multi-frequency, and graph methods address parts of this task, but still leave two issues: stable temporal representations and reliable relational aggregation. We propose **FinKAN**, a KAN-based framework with stability-selected predictive Markov-blanket stock aggregation. FinKAN pretrains a Data Shift Aligner only on temporally separated subdomains inside the training period, extracts nonlinear multi-frequency patterns with frequency-aware KAN modules, and builds a predictive Markov-blanket inclusion prior through sparse conditional neighborhood selection with temporal resampling. The prior gates direct, indirect, and residual cross-stock message passing without claiming interventional causality. Experiments on real-world stock datasets show that FinKAN improves prediction accuracy and investment metrics over competitive baselines and its backbone variant. Ablations and visualizations further show that predictive Markov-blanket aggregation yields a more robust and interpretable relational structure for stock forecasting.

1 INTRODUCTION

Stock forecasting is a core task in quantitative finance, supporting portfolio construction, risk management, and trading decisions. Given historical market observations, it predicts future stock returns. Unlike generic time-series forecasting, it operates in a non-stationary, interactive market where stock patterns and cross-stock dependencies evolve together. Thus, a model must do more than fit past prices: it should learn a reliable predictive structure under market shifts, including stable stock representations and trustworthy aggregation relations.

Existing studies improve different parts of this structure. For representation learning, MLPs and Transformers model historical prices (Fan & Shen, 2024; Vaswani et al., 2017; Li et al., 2024), while contrastive, meta-learning, and adaptation methods improve generalization under market change (Zhao et al., 2023; Wang et al., 2023; Chen et al., 2025). Yet they usually handle distribution adaptation, temporal modeling, and frequency fusion separately, leaving shifted representations, daily/hourly/minute observations, and frequency-dependent nonlinear patterns weakly integrated. For relation modeling, graph-based and relation-aware methods use predefined links, attention, correlation graphs, or learnable adjacency matrices (Kipf & Welling, 2022; Zhang et al., 2024). However, market co-movements may come from common shocks, sector exposure, liquidity effects, direct predictive links, or indirect paths. Treating them as simple correlations or free learned edges can spread spurious relations and hurt out-of-sample reliability.

These limits motivate a reliable predictive-structure view of heterogeneous stock forecasting under market shifts. This view highlights two challenges. The first is **non-stationary multi-frequency market dynamics**: market patterns change over time, while daily, hourly, and minute-level observations reflect different trading mechanisms and nonlinear behaviors. The second is **cross-stock dependency ambiguity**: observed co-movements may mix direct predictive links, indirect paths, and spurious correlations from common shocks. These challenges interact: unstable stock repre-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

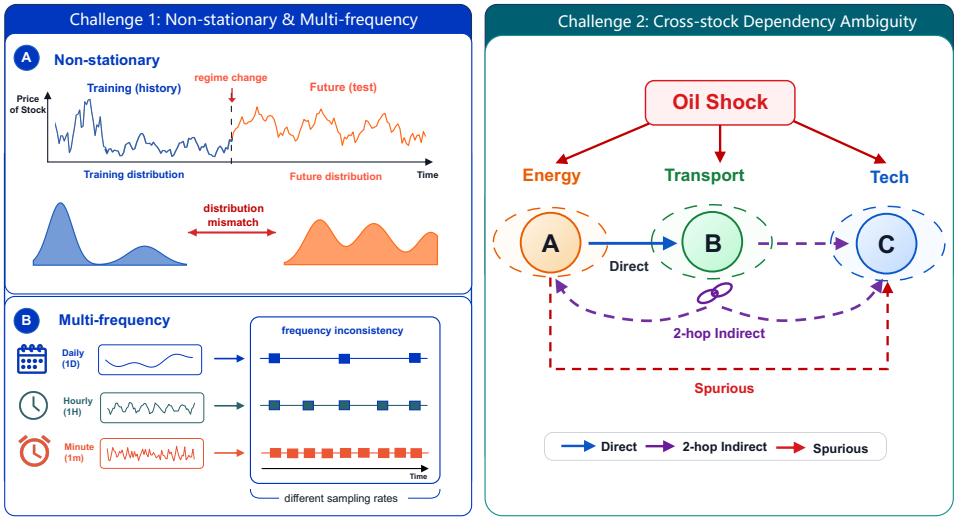


Figure 1: Two coupled challenges in heterogeneous stock forecasting: i) non-stationary multi-frequency market dynamics, and ii) cross-stock dependency ambiguity with direct, indirect, and spurious relations.

sentations can distort relation learning, while ambiguous aggregation can amplify noisy temporal signals.

To address these challenges, we propose **FinKAN**, a KAN-based framework with predictive Markov-blanket stock aggregation. FinKAN jointly models both parts of the predictive structure under a strictly chronological protocol. For non-stationary multi-frequency dynamics, it combines a *Data Shift Aligner* and a *Frequency-Temporal Branch*. The aligner is pretrained using temporally separated subdomains constructed only within the training period, the branch organizes daily, hourly, and minute-level observations into frequency-aware temporal representations, and KAN modules model frequency-dependent nonlinear patterns. Together, alignment improves temporal stability, frequency-aware modeling separates sampling scales, and KANs strengthen nonlinear representation.

For cross-stock dependency ambiguity, FinKAN introduces a *Predictive-Structural Relational Branch*. It estimates a stability-selected predictive Markov-blanket prior from pre-decision observations, builds direct relations, derives two-hop paths, and fuses them with a residual graph. This reduces reliance on unstable correlations while preserving flexibility.

In summary, this paper makes the following contributions.

- We study predictive-structure learning for heterogeneous stock forecasting under market shifts. This task requires stable stock representations and effective aggregation structures, shaped by non-stationary multi-frequency dynamics and cross-stock dependency ambiguity.
- We propose **FinKAN**, a KAN-based framework with stability-selected predictive Markov-blanket stock aggregation. FinKAN aligns shifted market representations, extracts nonlinear multi-frequency temporal patterns, and builds a direct–indirect predictive-structure prior for cross-stock aggregation.
- We evaluate FinKAN on four real-world stock datasets under forecasting and investment-oriented settings. FinKAN outperforms competitive baselines and controlled variants, and ablations plus graph visualizations verify the proposed representation and aggregation components.

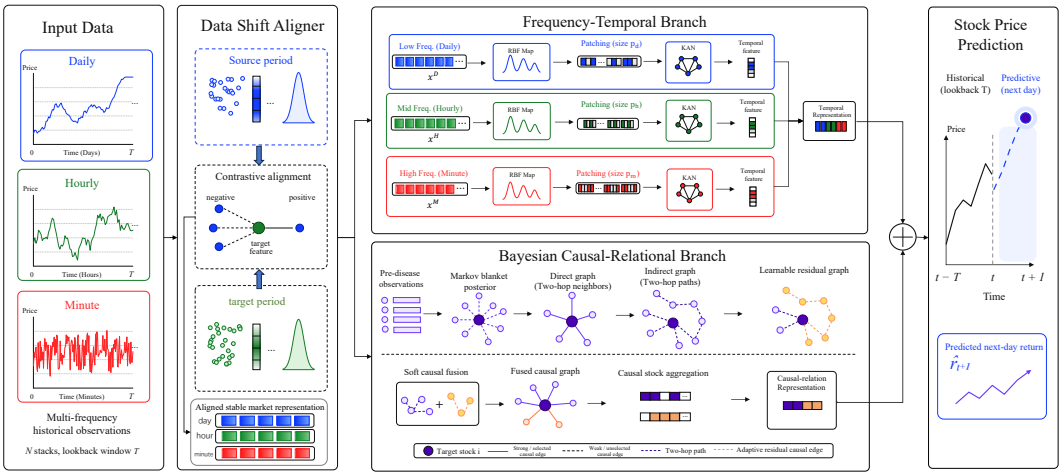


Figure 2: Overview of the proposed FinKAN.

2 Related Work

2.1 Heterogeneous Stock Forecasting

Recent deep learning methods improve forecasting by modeling temporal patterns and cross-stock dependencies. STHAN-SR (Sawhney et al., 2021) combines hypergraphs with Hawkes' attention for temporal and inter-stock correlations. MASTER (Li et al., 2024) uses market information for feature selection and correlation modeling. StockMixer (Fan & Shen, 2024) captures indicator-, temporal-, and stock-level dependencies through MLP mixing. However, these methods often rely on predefined links, correlation graphs, or free learned graphs, which may propagate spurious co-movements from market shocks, sector exposure, or liquidity effects.

2.2 Causal-Prior and KAN-Based Modeling

Causal reasoning helps distinguish stable predictive relations from spurious associations. Granger causality (Granger, 1969) tests whether one variable's past helps predict another, and Pearl's framework (Pearl, 2009) formalizes causality beyond correlation. Neural Granger Causality (Tank et al., 2021) extends discovery to nonlinear time series, while dynamic graph shift studies (Zhang et al., 2022) seek invariant relational representations. Recent causal foundation models, including CausalPFN (Balazadeh et al., 2025), Do-PFN (Robertson et al., 2025), and CausalFM (Ma et al., 2025), show the value of structural priors. Graph-conditioned and knowledge-informed models (Reuter et al., 2026; Xu et al., 2026) also show that partial graph knowledge can guide inference. Yet these works mainly target effect estimation or discovery, while the true stock graph is latent, dynamic, and hard to verify by intervention. FinKAN therefore uses a stability-selected predictive Markov-blanket prior to constrain message passing, not a deterministic predictive graph.

Kolmogorov-Arnold Networks (KANs) (Liu et al., 2024) provide expressive nonlinear approximation and have been extended to time-series modeling. TKANs (Genet & Inzirillo, 2024b) combine KANs with recurrent memory, TKAT (Genet & Inzirillo, 2024a) integrates KANs with Transformer-style temporal modeling, and SigKAN (Inzirillo & Genet, 2024) adds learnable path signatures. Still, KAN predictors alone do not solve temporal shift, multi-frequency heterogeneity, or uncertain stock relations. FinKAN uses KANs for frequency-specific representation and temporal aggregation, and uses predictive Markov-blanket aggregation to constrain cross-stock message passing.

3 METHODOLOGY

3.1 PROBLEM DEFINITION

Following (Fan & Shen, 2024; Wang et al., 2023), we formulate stock forecasting as one-step-ahead return prediction. Let $\mathcal{S} = \{s_1, \ldots, s_N\}$ denote N stocks. At prediction day t, each stock has multi-frequency historical observations:

$$X_t = [X_t^D, X_t^H, X_t^M] \in \mathbb{R}^{N \times T \times F}, \tag{1}$$

where T is the lookback window, X_t^D, X_t^H, and X_t^M are daily, hourly, and minute-level features, and F is the feature dimension. The target is the one-day return:

$$y_{i,t} = \frac{p_{i,t} - p_{i,t-1}}{p_{i,t-1}}, \tag{2}$$

where $p_{i,t}$ is the closing price of stock s_i on day t. The model predicts $\hat{Y}_t = f_\theta(X_t, \mathcal{D}_{<t}) \in \mathbb{R}^{N \times 1}$, where $\mathcal{D}_{<t}$ denotes all pre-decision observations. The predictive graph uses only $\mathcal{D}_{<t}$ to prevent leakage.

3.2 OVERVIEW

As shown in Figure 2, FinKAN has three components: a Data Shift Aligner (DSA), a Frequency-Temporal Branch, and a Predictive-Structural Relational Branch. The first two learn stable multi-frequency stock representations; the third builds a predictive Markov-blanket direct–indirect graph for cross-stock aggregation. Their outputs are fused for one-day return prediction.

3.3 DATA SHIFT ALIGNER

Stock market distributions shift over time, creating train-test mismatch. DSA reduces this mismatch through domain-aligned feature extraction. It is pretrained independently and frozen during downstream forecasting.

For frequency $\alpha \in \{D, H, M\}$, the Adaptive Feature Extractor produces frequency-aware representations:

$$f^\alpha = W_\alpha \phi(\|x^\alpha - c^\alpha\|), \tag{3}$$

where W_α is a learnable weight matrix, c^α is a learnable center, and $\phi(\cdot)$ is the Gaussian RBF. Features are reweighted by:

$$F^\alpha = f^\alpha \odot \mathrm{softmax}_\beta(f^\alpha), \tag{4}$$

where β controls weighting sharpness.

Alignment uses Nearest-Neighbor Contrastive Learning (NNCL) (Dwibedi et al., 2021), where the nearest source-domain neighbor of target representation z is the positive sample z^+. The pretraining objective is:

$$\mathcal{L}_{\mathrm{NNCL}} = -\log \frac{\exp(\mathrm{sim}(z, z^+)/\tau)}{\exp(\mathrm{sim}(z, z^+)/\tau) + \sum_{z^-} \exp(\mathrm{sim}(z, z^-)/\tau)}. \tag{5}$$

$\mathcal{L}_{\mathrm{NNCL}}$ is used *only* for DSA pretraining and is not included in the final forecasting objective.

3.4 FREQUENCY-TEMPORAL BRANCH

Daily data capture slow market trends, while higher-frequency data contain short-term fluctuations. FinKAN applies frequency-specific KAN projections using Gaussian RBFs for training stability on volatile data:

$$Z^\alpha = W_\alpha \phi(\|\mathrm{LayerNorm}(X^\alpha) - c^\alpha\|), \quad \alpha \in \{D, H, M\}. \tag{6}$$

Frequency-specific patching is then applied, with larger patches for high-frequency inputs to suppress noise and smaller patches for low-frequency inputs to preserve trends:

$$\bar{Z}^\alpha = \mathrm{Patch}_{p_\alpha}(Z^\alpha). \tag{7}$$

Patched features are concatenated and projected into a unified representation:

$$Z^F = \mathrm{MLP}\big(\mathrm{Concat}(\bar{Z}^D, \bar{Z}^H, \bar{Z}^M)\big)\,. \tag{8}$$

Multi-scale temporal dependencies are captured via dual-resolution aggregation:

$$h_l = \mathrm{MLP}_l(Z^F), \quad h_h = \mathrm{MLP}_h(Z^F), \tag{9}$$

$$u_l = \mathrm{KAN}_l(h_l), \quad u_h = \mathrm{KAN}_h(h_h). \tag{10}$$

The temporal representation is obtained by residual fusion:

$$H_T = \mathrm{MLP}_T(\mathrm{Concat}(u_l + h_l,\, u_h + h_h))\,, \tag{11}$$

encoding both long-term trends and short-term fluctuations.

3.5 Predictive-Structural Relational Branch

Cross-stock aggregation is important for stock forecasting, but observed co-movement does not necessarily indicate a stable predictive relation. Stocks may move together because of market-wide shocks, sector exposure, or liquidity effects. To avoid relying on arbitrary correlations, FinKAN constructs a predictive Markov-blanket-guided graph from pre-decision observations.

stability-selected predictive Markov-blanket inclusion probability. For each target stock s_i, FinKAN estimates whether stock s_j belongs to its predictive Markov blanket before day t:

$$\Pi_{ij}^t = P(s_j \in MB_i^t \mid \mathcal{D}_{<t}), \tag{12}$$

where MB_i^t denotes the predictive Markov-blanket neighborhood of s_i, and $\mathcal{D}_{<t}$ contains only observations available before day t. The term represents conditional predictive dependence rather than a fully identified causal mechanism. In practice, this posterior is approximated by repeated historical resampling:

$$\Pi_{ij}^t = \frac{1}{B}\sum_{b=1}^{B} \mathbf{1}\left[s_j \in \widehat{MB}_{i,b}^t\right], \tag{13}$$

where B is the number of resampling trials. This inclusion probability provides a stable predictive prior for aggregation rather than an interventional predictive graph. It captures conditional predictive neighborhoods under historical observations and should not be interpreted as causal identification. Appendix E gives the estimator, window, and leakage-control details.

Direct and indirect relational graphs. The direct graph keeps the top-K Markov-blanket neighbors for each stock. We apply a Top-K mask that keeps the K largest entries in each row of Π^t ($K = 10$ in our experiments) and zeros the rest:

$$A_{ij}^{dir,t} = \mathrm{Norm}\big(\mathrm{TopKMask}_K\big(\Pi_{ij}^t\big)\big)\,, \tag{14}$$

where $\mathrm{TopKMask}_K(\cdot)$ keeps only the top-K row entries, and $\mathrm{Norm}(\cdot)$ denotes row-stochastic normalization.

Because influence can pass through intermediate stocks, FinKAN also derives an indirect graph:

$$A^{ind,t} = \mathrm{Norm}\left((A^{dir,t})^2 \odot (1 - A^{dir,t})\right). \tag{15}$$

The squared term captures two-hop paths, while $(1 - A^{dir,t})$ avoids duplicating direct edges. Thus, $A^{dir,t}$ and $A^{ind,t}$ encode first- and second-order relational signals.

Learnable residual graph. A prior-only graph may be too rigid for dynamic markets. FinKAN therefore keeps a learnable residual graph with free edge parameters:

$$A^{learn} = \sigma(\Theta), \tag{16}$$

where $\Theta \in \mathbb{R}^{N\times N}$ is a learnable edge-logit matrix and $\sigma(\cdot)$ is the sigmoid function. Each edge can be optimized from data, while the structural gate in Eq. 18 limits its influence. If an antisymmetric residual graph uses stock embeddings $E_1, E_2 \in \mathbb{R}^{N\times d}$, the valid form is $\mathrm{ReLU}(E_1E_2^\top - E_2E_1^\top) \in \mathbb{R}^{N\times N}$; $\mathrm{ReLU}(E_1^\top E_2 - E_2^\top E_1)$ yields a $d \times d$ matrix and is a notation error. FinKAN uses the free edge-logit form in Eq. 16, matching the implementation.

Table 1: Overall performance on CSI100 and CSI300. Lower MAE, RMSE, and MedAE are better, while higher AR and SR are better. The best results are **bold** and the second-best results are underlined.

Method	CSI100					CSI300				
	MAE	RMSE	MedAE	AR	SR	MAE	RMSE	MedAE	AR	SR
General time-series baselines										
Linear	0.04343	0.05534	0.03292	-1.253	-0.111	0.03234	0.05812	0.03812	-1.591	-0.221
DLinear	0.03845	0.04445	0.02743	10.987	0.343	0.02982	0.04598	0.02981	12.321	0.345
Mamba	0.04012	0.05143	0.03232	5.321	0.254	0.03432	0.04231	0.03871	6.311	0.233
Transformer	0.02812	0.03745	0.02871	20.291	0.899	0.02421	0.02981	0.02872	21.475	0.934
Autoformer	0.03819	0.04543	0.02721	11.910	0.389	0.02879	0.03871	0.03110	11.331	0.398
Informer	0.04342	0.05348	0.04231	-1.451	-0.143	0.03321	0.04312	0.04435	-1.786	-0.291
Non-stationary	0.03212	0.03867	0.02423	17.344	0.651	0.02412	0.02871	0.02981	19.541	0.723
Stock-specific baselines										
Enhancer	0.02871	0.03612	0.02221	20.981	0.887	0.02131	0.02811	0.02495	24.111	0.952
MASTER	0.03070	0.03800	0.02380	19.210	0.742	0.02280	0.02900	0.02620	21.840	0.803
StockMixer	0.03130	0.03920	0.02460	17.950	0.697	0.02320	0.02970	0.02670	21.090	0.772
KAN-based baselines										
TKAN	0.03660	0.04380	0.02760	12.010	0.421	0.02820	0.03600	0.03120	14.470	0.505
MT-KAN	0.03420	0.04100	0.02570	15.340	0.561	0.02580	0.03200	0.02870	18.390	0.668
Causal-discovery baseline										
PCMCI+Linear	0.03810	0.04600	0.02920	9.510	0.324	0.02950	0.03800	0.03250	12.470	0.423
FinKAN family										
FinKAN-Base	0.02324	0.03013	0.01918	27.987	1.134	0.01871	0.02123	**0.01731**	31.532	1.112
FinKAN	**0.02131**	**0.02998**	**0.01896**	**29.788**	**1.141**	**0.01799**	**0.02110**	0.01743	**32.457**	**1.214**

Soft causal fusion. A structural gate relaxes the Bayesian prior toward a uniform baseline (ρ=0.10, η=0.50):

$$G^t = \rho\mathbf{1} + (1-\rho)\cdot \mathrm{Clip}\big(A^{dir,t} + \eta A^{ind,t},\, 0,\, 1\big)\,. \tag{17}$$

The final aggregation matrix combines all relational signals:

$$A^{final,t} = \mathrm{Norm}\big(\alpha A^{dir,t} + \beta A^{ind,t} + \gamma A^{learn} \odot G^t + I\big)\,, \tag{18}$$

where α, β, γ are learnable scalars and I is the identity for self-loops. Cross-stock aggregation is then:

$$H_C = \mathrm{LayerNorm}\big(Z^F + \mathrm{Dropout}\big(A^{final,t} Z^F W_C\big)\big)\,. \tag{19}$$

The final prediction fuses temporal and relational representations:

$$H = \mathrm{InterAgg}(H_T, H_C), \qquad \hat{Y}_t = \mathrm{Linear}(H). \tag{20}$$

3.6 Training Objective

FinKAN uses two stages. **Stage 1**: DSA is pretrained with $\mathcal{L}_{\mathrm{NNCL}}$, then frozen. **Stage 2**: the remaining modules are trained end-to-end with MSE:

$$\mathcal{L}_{task} = \frac{1}{|\Omega_t|}\sum_{i\in\Omega_t} (\hat{y}_{i,t} - y_{i,t})^2, \tag{21}$$

where Ω_t is the set of valid labels. $\mathcal{L}_{\mathrm{NNCL}}$ is not added to $\mathcal{L}_{task}$, so gains can be attributed to the representation and predictive-structural design. Appendix F clarifies DSA data access.

4 Experimental Setup

Datasets and Metrics. We evaluate FinKAN on CSI100, CSI300, NASDAQ, and TSE. CSI100 and CSI300 contain Chinese large-cap stocks with daily, hourly, and minute-level features, each

Table 2: Backbone component ablation on CSI300. This table verifies the effect of the inherited FinKAN-Base modules before introducing Bayesian predictive-relational stock aggregation. The best results are **bold** and the second-best results are underlined.

Model Variant	CSI300			
	IC	ICIR	RankIC	RankICIR
w/o DSA	0.0196	0.3418	0.0174	0.3349
w/o Frequency Mixer	0.0478	0.6608	0.0418	0.7387
w/o Temporal Aggregation	0.0442	0.7034	0.0451	0.7236
w/o Stock Aggregation	0.0274	0.4360	0.0227	0.4395
DSA + Stock Aggregation	0.0444	0.6405	0.0456	0.5820
FinKAN-Base	**0.0656**	**0.8835**	**0.0560**	**0.8912**

Table 3: Ablation of the Data Shift Aligner (DSA) and Nearest-Neighbor Contrastive Learning (NNCL) in FinKAN-Base on NASDAQ and TSE. Higher SR and IRR are better.

Dataset	Model Variant	SR	IRR
NASDAQ	FinKAN-Base	1.47	0.59
	w/o DSA	0.78	0.29
	w/o NNCL	1.32	0.41
TSE	FinKAN-Base	1.34	0.71
	w/o DSA	0.67	0.34
	w/o NNCL	1.25	0.66

with open, close, high, low, and volume. We predict one-day returns and report MAE, RMSE, and MedAE for accuracy, plus AR and SR for investment performance. On NASDAQ and TSE, models rank stocks by predicted return and are evaluated by SR and IRR. Backbone ablations also report IC, ICIR, RankIC, and RankICIR for cross-sectional correlation and temporal stability.

Baselines. We compare FinKAN with Linear, DLinear (Zeng et al., 2023), Mamba (Gu & Dao, 2023; Dao & Gu, 2024), Transformer (Vaswani et al., 2017), Autoformer (Wu et al., 2021), Informer (Zhou et al., 2021), Non-stationary Transformer (Liu et al., 2022), Enhancer, MASTER (Li et al., 2024), StockMixer (Fan & Shen, 2024), TKAN (Genet & Inzirillo, 2024b), MT-KAN, and PCMCI+Linear. We also include **FinKAN-Base**, which keeps the same temporal-frequency backbone but replaces predictive-structure relational aggregation with unconstrained learnable stock aggregation, isolating the proposed graph module.

Graph Variants. To ablate the predictive-structural design, we build five variants with the same representation backbone: **FinKAN-Base** (unconstrained aggregation), **FinKAN-CorrGraph** (Pearson graph), **FinKAN-DirectMB** (direct Markov-blanket graph), **FinKAN-DirectIndirectMB** (direct plus two-hop paths), and full **FinKAN** (soft fusion of direct, indirect, and residual graphs). They form a chain from unconstrained to causally structured aggregation.

Implementation Details. All models use PyTorch, a 7-day lookback window, hidden dimension 256, and Adam optimization with learning rates in $10^{-4}, 10^{-5}, 10^{-6}$. Early stopping is used for training. The graph uses only pre-decision history to prevent leakage. Details are provided in the appendices.

5 Experiments

We evaluate **FinKAN** around three research questions:

- **RQ1**: Does FinKAN improve forecasting and investment-oriented performance over competitive baselines and its backbone variant?
- **RQ2**: Which representation and graph components contribute to the performance improvement?
- **RQ3**: What mechanisms and limitations can be observed from the learned representations and predictive-relational graph?

5.1 Overall Performance (RQ1)

Table 1 shows that FinKAN outperforms all baselines on most CSI100 and CSI300 metrics, including stock-specific, KAN-based, and causal-discovery methods. Compared with FinKAN-Base, CSI100 MAE drops from 0.02324 to 0.02131 and AR rises from 27.987 to 29.788; CSI300 SR rises from 1.112 to 1.214. Since both models share the same backbone and objective, the gain mainly

Table 4: Graph variant ablation on CSI100 and CSI300. All variants share the same FinKAN-Base backbone and differ only in the stock aggregation graph. Lower MAE, RMSE, and MedAE are better, while higher AR and SR are better. The best results are **bold** and the second-best results are underlined.

Model Variant	CSI100					CSI300				
	MAE	RMSE	MedAE	AR	SR	MAE	RMSE	MedAE	AR	SR
FinKAN-Base	0.02324	0.03013	0.01918	27.987	1.134	0.01871	0.02123	**0.01731**	31.532	1.112
FinKAN-CorrGraph	0.02212	0.03113	0.02119	24.231	0.998	0.01967	0.02454	0.01943	27.901	1.002
FinKAN-DirectMB	0.02448	0.03311	0.02632	25.789	0.977	0.01978	0.02541	0.02133	27.431	1.001
FinKAN-DirectIndirectMB	0.02423	0.03297	0.02454	26.976	0.997	0.01897	0.02313	0.02044	29.123	1.023
FinKAN	**0.02131**	**0.02998**	**0.01896**	**29.788**	**1.141**	**0.01799**	**0.02110**	0.01743	**32.457**	**1.214**

Table 5: Graph variant ablation on NASDAQ and TSE. All variants share the same FinKAN-Base backbone. Higher SR and IRR are better. The best results are **bold**.

Model Variant	NASDAQ		TSE	
	SR	IRR	SR	IRR
FinKAN-Base	1.47	0.59	1.34	0.71
FinKAN-CorrGraph	1.43	0.52	1.33	0.64
FinKAN-DirectMB	1.41	0.49	1.32	0.68
FinKAN-DirectIndirectMB	1.42	0.51	1.35	0.70
FinKAN	**1.52**	**0.63**	**1.36**	**0.74**

comes from predictive-structure relational aggregation. Improvements are larger on AR and SR than on MedAE, suggesting better ranking and portfolio construction rather than uniform point-wise error reduction. The only exception is CSI300 MedAE, where FinKAN-Base is slightly better.

Table 8 (Appendix A) further validates FinKAN on NASDAQ and TSE under investment-oriented evaluation. FinKAN improves SR and IRR over FinKAN-Base on both datasets: on NASDAQ, SR increases from 1.47 to 1.52 and IRR from 0.59 to 0.63; on TSE, SR increases from 1.34 to 1.36 and IRR from 0.71 to 0.74. These moderate but consistent gains show that the predictive relational module also works beyond China, where ranking and portfolio selection matter more than point-wise accuracy.

5.2 Ablation Study (RQ2)

We conduct three ablation groups for RQ2: backbone components, distribution alignment (DSA & NNCL), and predictive graph variants. Appendix D provides stepwise details.

Backbone component ablation. Table 2 shows that each FinKAN-Base module contributes. Removing DSA causes the largest drop across all metrics, confirming that temporal alignment is critical for later temporal and relational modeling. Removing Stock Aggregation also sharply reduces IC and RankIC, showing the value of cross-stock information. Removing the Frequency Mixer or Temporal Aggregation further weakens the model, confirming that both are needed in the temporal-frequency backbone.

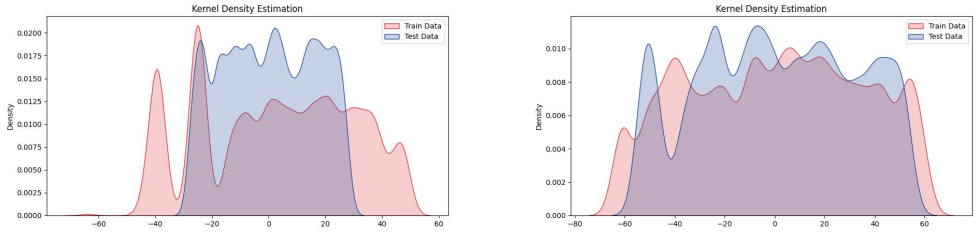
Distribution alignment ablation. Table 3 studies distribution alignment in the investment setting. Removing DSA causes the largest drop: on NASDAQ, SR falls from 1.47 to 0.78 and IRR from 0.59 to 0.29; on TSE, SR falls from 1.34 to 0.67 and IRR from 0.71 to 0.34. Removing only NNCL has a smaller effect, showing that NNCL adds local alignment. DSA stabilizes features, NNCL strengthens them, and both support the predictive graph.

Table 6: Ablation of graph prior construction.

Graph prior	IC	SR
CorrGraph	0.041 ± 0.004	0.927 ± 0.07
LagCorr-Stability	0.047 ± 0.004	0.921 ± 0.07
Predictive-MB	0.053 ± 0.005	0.960 ± 0.07
Predictive-MB + indirect + residual	0.059 ± 0.005	1.107 ± 0.08

Table 7: Shift-conditioned investment performance.

Shift level	FinKAN	CorrGraph	FinKAN-Base
Low	1.067	0.907	0.966
Medium	1.097	0.922	0.996
High	1.137	0.897	0.976
Full	1.107	0.927	1.006



(a) Before DSA

(b) After DSA

Figure 3: Training and test feature distributions before and after DSA, visualized by t-SNE with kernel density estimation.

Predictive graph variant ablation. Tables 4 and 5 compare graph designs under the same FinKAN-Base backbone. FinKAN achieves the best AR, SR, and IRR across all four datasets, validating the direct–indirect–residual design for both forecasting and investment settings. The progression from FinKAN-CorrGraph to FinKAN-DirectMB to FinKAN-DirectIndirectMB to FinKAN shows that correlation priors are insufficient, direct Markov-blanket edges are too restrictive, and two-hop paths help but still need a learnable residual graph. Robust stock aggregation therefore needs direct predictively plausible edges, indirect paths, and residual flexibility.

5.3 Mechanism and Robustness Analysis (RQ3)

Temporal distribution alignment. Figure 3 visualizes the training and test feature distributions before and after DSA. The original representations exhibit a clear distribution gap, whereas DSA substantially reduces the mismatch. This suggests that DSA provides a more stable feature space for subsequent predictive-relational aggregation.

Predictive structure under market shifts. We further examine whether the improvement arises from predictive structure rather than graph stabilization alone. The predictive Markov-blanket prior provides more informative neighborhoods than correlation-based alternatives, while direct–indirect–residual fusion preserves flexibility under changing market conditions. The advantage becomes more pronounced under stronger distribution shifts, suggesting that predictive neighborhoods are less sensitive to unstable co-movements. Additional analyses of adaptive feature weights and graph-component edge distributions are provided in Appendix B. These visualizations characterize predictive dependencies and should not be interpreted as evidence of interventional causality.

6 Conclusion

This paper proposes FinKAN, a predictive-structure framework for heterogeneous stock forecasting under market shifts. It combines a Data Shift Aligner, a Frequency-Temporal Branch, and a Predictive-Structural Relational Branch for temporal shifts, multi-frequency signals, and direct–indirect–residual stock relations. Residual fusion preserves flexibility beyond the predictive prior. Experiments on CSI100, CSI300, NASDAQ, and TSE show improved forecasting and investment-oriented performance over competitive baselines and FinKAN-Base. Ablations and visualizations validate the relational design.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, concision, organization, and formatting of the manuscript. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the scientific content, claims, and conclusions of this work.

REPRODUCIBILITY STATEMENT

Reproducibility details are provided in Sections 3–4 and Appendices E–I, including predictive-graph construction, data-access and leakage controls, the investment simulation protocol, statistical testing, dataset splits, and hyperparameter settings. All primary evaluations follow chronological splits, and model and graph construction use only training or pre-decision information.

ETHICS STATEMENT

This work does not involve human-subject experiments or the collection of personally identifiable information. The reported financial forecasting and investment metrics are intended for research evaluation rather than deployment or financial advice. Real-world use may be affected by distribution shifts, data bias, transaction costs, and other market risks. The authors adhere to the ICLR Code of Ethics.

REFERENCES

Vahid Balazadeh, Hamidreza Kamkari, Valentin Thomas, Benson Li, Junwei Ma, Jesse C Cresswell, and Rahul G Krishnan. CausalPFN: Amortized causal effect estimation via in-context learning, 2025. URL https://arxiv.org/abs/2506.07918.

Weijun Chen, Shun Li, Xipu Yu, Heyuan Wang, Wei Chen, and Tengjiao Wang. Enhancer: A distribution-aware framework with temporal-relational meta-learning for stock prediction. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *International Conference on Machine Learning (ICML)*, 2024.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9588–9597, 2021.

Jinyong Fan and Yanyan Shen. Stockmixer: A simple yet strong mlp-based architecture for stock price forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8389–8397, 2024.

Remi Genet and Hugo Inzirillo. A temporal kolmogorov-arnold transformer for time series forecasting. *arXiv preprint arXiv:2406.02486*, 2024a.

Remi Genet and Hugo Inzirillo. Tkan: Temporal kolmogorov-arnold networks. *arXiv preprint arXiv:2405.07344*, 2024b.

Clive W J Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438, 1969.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Hugo Inzirillo and Remi Genet. Sigkan: Signature-weighted kolmogorov-arnold networks for time series. *arXiv preprint arXiv:2406.17890*, 2024.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2022.

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, pp. 162–170, 2024.

Yong Liu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Non-stationary transformers: Exploring the stationarity in time series forecasting. *Advances in Neural Information Processing Systems*, 35:9881–9893, 2022.

Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark. Kan: Kolmogorov-arnold networks. *arXiv preprint arXiv:2404.19756*, 2024.

Yuchen Ma, Dennis Frauen, Emil Javurek, and Stefan Feuerriegel. Foundation models for causal inference via prior-data fitted networks, 2025. URL https://arxiv.org/abs/2506.10914.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2nd edition, 2009.

Arik Reuter, Anish Dhir, Cristiana Diaconu, Jake Robertson, Ole Ossen, Frank Hutter, Adrian Weller, Mark van der Wilk, and Bernhard Schölkopf. Use what you know: Causal foundation models with partial graphs, 2026. URL https://arxiv.org/abs/2602.14972.

Jake Robertson, Arik Reuter, Siyuan Guo, Noah Hollmann, Frank Hutter, and Bernhard Schölkopf. Do-PFN: In-context learning for causal effect estimation, 2025. URL https://arxiv.org/abs/2506.06039.

Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, Tyler Derr, and Rajiv Ratn Shah. Stock selection via spatiotemporal hypergraph attention network: A learning to rank approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 497–504, 2021.

Alex Tank, Ian Covert, Nicholas Foti, Ali Shojaie, and Emily B Fox. Neural granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):4267–4279, 2021. doi: 10.1109/TPAMI.2021.3065601.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Mingjie Wang, Feng Chen, Jianxiong Guo, and Weijia Jia. Improving stock trend prediction with multi-granularity denoising contrastive learning. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–10. IEEE, 2023.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

Wenbo Xu, Yue He, Yunhai Wang, Xingxuan Zhang, Kun Kuang, Yueguo Chen, and Peng Cui. A knowledge-informed pretrained model for causal discovery, 2026. URL https://arxiv.org/abs/2603.20842.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11121–11128, 2023.

Zeyang Zhang, Xin Wang, Ziwei Zhang, Haoyang Li, Zhou Qin, and Wenwu Zhu. Dynamic graph neural networks under spatio-temporal distribution shift. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Zhenwei Zhang, Linghang Meng, and Yuantao Gu. Sageformer: Series-aware framework for long-term multivariate time series forecasting. *IEEE Internet of Things Journal*, 2024.

Lifan Zhao, Shuming Kong, and Yanyan Shen. Doubleadapt: A meta-learning approach to incremental learning for stock trend forecasting. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3492–3503, 2023.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.

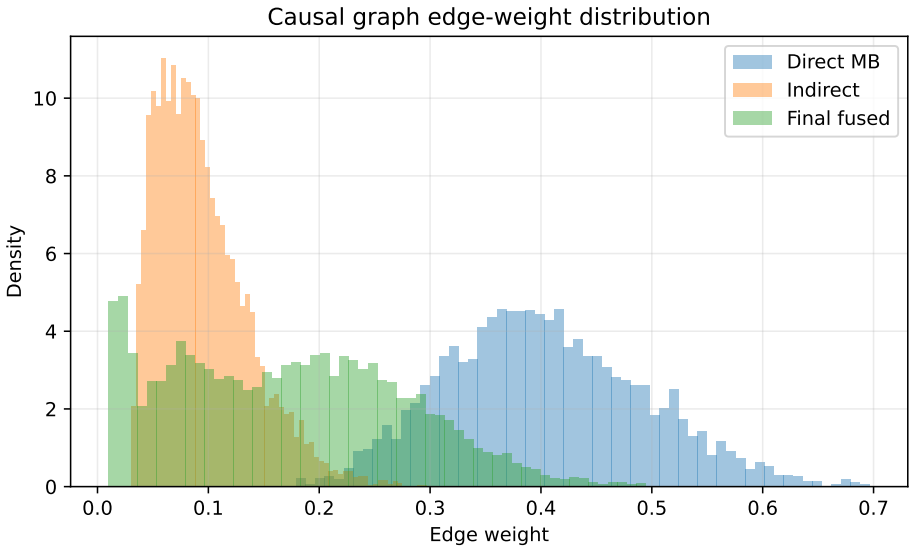


Figure 4: Edge-weight distributions of the direct predictive Markov-blanket graph, indirect two-hop graph, and final fused graph.

A Investment-Oriented Performance on NASDAQ and TSE

Table 8: Investment-oriented performance on NASDAQ and TSE. Higher SR and IRR are better. The best results are in **bold** and the second-best results are underlined.

Method	NASDAQ		TSE	
	SR	IRR	SR	IRR
General time-series baselines				
Linear	0.75	0.12	0.53	0.17
DLinear	1.08	0.25	0.89	0.42
Mamba	0.98	0.20	0.86	0.31
Transformer	1.36	0.41	1.23	0.51
Autoformer	1.17	0.32	1.05	0.55
Informer	0.74	0.13	0.46	0.15
Non-stationary	1.27	0.36	1.18	0.62
Stock-specific baseline				
Enhancer	1.31	0.41	1.12	0.53
FinKAN family				
FinKAN-Base	1.47	0.59	1.34	0.71
FinKAN	**1.52**	**0.63**	**1.36**	**0.74**

Table 8 evaluates investment-oriented performance on NASDAQ and TSE. FinKAN achieves the best SR and IRR on both datasets. Over FinKAN-Base, it improves NASDAQ SR from 1.47 to 1.52 and IRR from 0.59 to 0.63; on TSE, SR rises from 1.34 to 1.36 and IRR from 0.71 to 0.74. These moderate but consistent gains show that the predictive-relational module transfers beyond China and supports ranking-based investment decisions.

B Additional Mechanism Analysis

Predictive-Relational Graph Structure

Figure 4 compares the edge-weight distributions of FinKAN's relational components. The direct predictive Markov-blanket graph concentrates on stronger weights, corresponding to first-order pre-

Table 9: Error subset analysis on the test set. The table reports the number of cases in each error subset, the proportion of the total test set, the number corrected by FinKAN relative to baseline errors, and the dominant failure mode associated with each subset.

Error subset	Cases	Ratio (%)	Corrected #	Corrected ratio (%)	Common distribution / failure mode
Shared FP	80	9.8	78	97.5	Sparse, high-volatility, boundary-near samples
Shared FN	98	10.2	83	84.7	Long-tail, weak-trend, delayed-reversal samples
Strong-baseline errors	15	4.5	7	46.7	Hard cases even for strong baselines
All-baseline shared hard cases	45	55.0	34	75.6	Common historical challenge across model families

dictively plausible relations selected by the stability-based inclusion score. The indirect graph places more mass on weaker weights, consistent with its role in representing two-hop paths. The fused graph covers a broader range through the residual component, providing flexibility beyond the structural prior. These distributions illustrate how FinKAN separates direct, indirect, and residual predictive dependencies; they do not establish interventional causal effects.

C Failure Case Analysis

FinKAN corrects many shared false positives (97.5%) and shared false negatives (84.7%), showing that the predictive-relational graph recovers errors missed by baselines. For all-baseline hard cases, the correction ratio remains high at 75.6%, suggesting good generalization to difficult instances. However, it drops to 46.7% on strong-baseline errors, so some samples remain hard. These failure modes—sparse high-volatility, boundary-near, and delayed-reversal samples—have highly localized signals that a historical pre-decision prior cannot fully capture. This motivates regime-aware Bayesian graph updates under high-volatility or reversal-dominated periods.

D Supplementary Ablation Details

This appendix provides stepwise analysis omitted from the main text for space.

Backbone ablation details. Table 2 shows that removing DSA degrades all four metrics most, confirming temporal alignment as a prerequisite for temporal and relational modeling. Removing Stock Aggregation causes a large IC/RankIC drop, showing that cross-stock information is essential. Removing Frequency Mixer or Temporal Aggregation also weakens performance. This ablation links FinKAN-Base and FinKAN, shows that each backbone component targets a distinct aspect, and motivates predictive-relational Stock Aggregation.

Distribution alignment details. Table 3 quantifies alignment under investment settings. Removing DSA causes sharp SR/IRR drops on both datasets (NASDAQ: 1.47→0.78, 0.59→0.29; TSE: 1.34→0.67, 0.71→0.34). Removing only NNCL while keeping DSA causes milder degradation, showing that NNCL adds local alignment. Together, DSA and NNCL provide a stable feature space for the predictive graph.

Graph variant details. Tables 4 and 5 give full comparisons. FinKAN-CorrGraph is generally weaker, showing that correlation priors are insufficient. FinKAN-DirectMB uses only direct Markov-blanket edges and underperforms the final model, suggesting these edges are too restrictive. FinKAN-DirectIndirectMB improves over DirectMB in several cases, especially on investment metrics, but still trails FinKAN. Thus, the learnable residual graph is needed for relations not covered by the Bayesian prior. The progression from FinKAN-Base to FinKAN supports direct predictively plausible edges, indirect paths, and residual flexibility for robust aggregation.

E Causal Graph Construction Details

This appendix details the predictive Markov-blanket prior used by the predictive-structural relational branch. It complements Section 3 and focuses on the estimator, temporal window, and residual-graph control.

Markov-blanket posterior estimator. The quantity $\Pi_{ij}^t = P(s_j \in MB_i^t \mid \mathcal{D}_{<t})$ in Eq. (13) is implemented as a bootstrap posterior-inclusion score over lagged return dependencies. For a training window before evaluation, we form a return matrix $R \in \mathbb{R}^{T_g \times N}$, where rows are trading days and columns are stocks. For each source–target pair (j, i), the source is the one-day lagged return $R_{1:T_g-1,j}$ and the target is the next-day sequence $R_{2:T_g,i}$. Both sequences are cross-sectionally demeaned by day and standardized. In bootstrap trial b, we resample time indices with replacement and compute

$$S_{ij}^{(b)} = \left| \mathrm{Corr}\left(R_{\mathcal{I}_b,j}^{lag}, R_{\mathcal{I}_b,i}^{now} \right) \right|, \tag{22}$$

with the diagonal set to zero. The estimated blanket for target i in trial b is then

$$\widehat{MB}_{i,b} = \mathrm{TopK}_{K_{MB}} \left(\{S_{ij}^{(b)}\}_{j=1}^{N} \right), \tag{23}$$

and the posterior inclusion matrix is

$$\Pi_{ij} = \frac{1}{B} \sum_{b=1}^{B} \mathbf{1}[j \in \widehat{MB}_{i,b}]. \tag{24}$$

The default setting is $K_{MB} = 10$, $B = 50$, and random seed 3. Missing, infinite, or degenerate scores are set to zero. This estimator is a linear lagged-dependence proxy for Markov-blanket inclusion, not a nonlinear conditional-independence discovery algorithm. It uses no explicit conditional-independence test or p-value threshold; row-wise Top-K gives sparsity, and bootstrap frequency gives stability. The matrix is therefore a causal-plausibility prior for constrained message passing, not evidence of interventional causality.

Temporal construction of Π. The notation Π^t emphasizes that the causal-prior graph must be available before prediction day t. In the reported experiments, the graph is built once from the training split before validation and testing. For CSI100/CSI300, it uses observations from 2014-01-01 to 2018-09-30 and is then fixed; no validation or test labels are used. A rolling version could use an expanding or sliding window ending at $t-1$, but we did not use it in the main tables to keep baselines under the same split and avoid repeated test-period adaptation.

Fusion coefficients and residual-graph control. The soft-fusion coefficients in Eq. 18 are implemented through unconstrained logits $\lambda = (1.0, 0.5, 0.5)$ followed by a softmax:

$$(\alpha, \beta, \gamma) = \mathrm{softmax}(\lambda). \tag{25}$$

This keeps the mixture weights non-negative and summing to one during training. The residual edge logits Θ in $A^{learn} = \sigma(\Theta)$ are initialized to zero. Their influence is controlled by the structural gate

$$G = \rho \mathbf{1} + (1-\rho)\mathrm{Clip}(A^{dir} + \eta A^{ind}, 0, 1), \tag{26}$$

with $\rho = 0.10$ and $\eta = 0.50$ by default. The implementation also applies row normalization, self-loops, graph-layer dropout of 0.10, and validation-based early stopping. The main objective adds no L1 sparsity or temporal-smoothing penalty to A^{learn}. Since the graph is fixed per split rather than updated daily, temporal smoothness of A^{learn} is not applicable.

F DSA Pretraining and Data Access

DSA pretraining uses training-period features as the source domain and unlabeled target-period features as the target domain for NNCL. The MinMax scaler is fitted only on the training range and then applied to both domains, so target distributions do not affect scaling. NNCL uses no target labels, portfolio returns, or future labels; it uses target features as queries and training features as

keys. Thus, the reported DSA setting is an unsupervised transductive domain-adaptation protocol. For the main evaluation, we adopt a strictly inductive chronological protocol: DSA is pretrained only within the training period by constructing temporally separated source and target subdomains. After pretraining, DSA parameters are frozen and no validation- or test-period features are accessed during representation learning or forecasting. The transductive configuration is treated only as an offline adaptation upper-bound setting and is not mixed with the primary inductive benchmark. The w/o DSA and w/o NNCL ablations measure this stage. Under strict inductive deployment, DSA can be pretrained without target queries, or NNCL can be disabled.

G Investment Simulation Protocol

The investment metrics are computed from daily cross-sectional rankings. Each test day, the model ranks stocks by predicted return and forms an equal-weight long-only portfolio from the Top-30 stocks. The portfolio is rebalanced daily without a turnover cap. The setting is frictionless: transaction costs, turnover constraints, slippage, short-sale constraints, and financing costs are omitted, and the risk-free rate is zero. Thus, SR and IRR are diagnostic ranking metrics, not a fully executable cost-adjusted backtest. Under proportional transaction costs, the realized daily portfolio return would be

$$r_{p,t}^{cost} = r_{p,t} - c \cdot \mathrm{Turnover}_t, \tag{27}$$

so higher-turnover methods would be penalized more. We leave cost-sensitive evaluation with turnover reporting to future work.

H Statistical Testing and Hyperparameter Sensitivity

Statistical testing protocol. Supplementary experiments use random seed 3 for reproducibility, and the paper does not claim formal significance for every pairwise comparison. When seed-level logs or daily prediction arrays are available, paired tests can be used because all models share the same trading days. For metric m, a seed-level 95% confidence interval is $\bar{m} \pm t_{0.975,n-1} s_m/\sqrt{n}$ over n seeds. For daily ranking metrics such as IC, RankIC, or portfolio returns, a paired bootstrap over test days gives a nonparametric interval for the difference between FinKAN and each baseline. A paired t-test or Wilcoxon signed-rank test over daily differences can serve as a complementary check. This avoids treating stock-day observations as independent when they share a market day.

Sensitivity of graph hyperparameters. The main text reports sensitivity to lookback window T and global vector length L, while Table 10 lists graph-specific grids. The main causal-graph hyperparameters are the Markov-blanket budget K_{MB}, indirect weight η, and gate relaxation ρ. Smaller K_{MB} gives a conservative graph but may miss sector-mediated relations; larger K_{MB} improves coverage but may admit spurious co-movements. The relaxation ρ controls how much the residual graph can bypass the prior: very small ρ makes aggregation close to a hard prior, while large ρ weakens causal gating. The indirect weight η controls two-hop paths. The reported setting uses $K_{MB} = 10$, $\eta = 0.50$, and $\rho = 0.10$. Robustness checks use $K_{MB} \in \{5, 10, 20, 30, 50\}$, $\eta \in \{0.3, 0.5, 0.7\}$, and $\rho \in \{0.05, 0.1, 0.2\}$; we expect a plateau near the default, since overly sparse or dense priors reduce ranking stability.

I Hyperparameter Settings

This appendix summarizes all hyperparameter settings in one table. The table is organized by Category: training settings define FinKAN optimization and representation; frequency and temporal settings define the multi-frequency KAN branches; GraphNet and BMBCG settings define baseline and predictive-relational aggregation; DSA settings describe pretraining and inference; dataset entries report splits and stock universes; and sensitivity entries list hyperparameter grids.

The Category column keeps the grouping from the configuration sheet. Each Category cell is vertically merged across its parameter rows, and horizontal rules separate Categories. The Sensitivity Grid column follows the original CSV.

Table 10: Aggregated hyperparameter settings grouped by Category.

Category	Parameter	Symbol	Default / Value	Unit / Range	Sensitivity Grid	Used In / Notes
FinKan Training	Lookback window	T	7	days	7, 14, 30 (Fig.3a); optional 4, 21	All main experiments; Fix L=16 when sweeping T
	Global vector length	L	16	–	4, 8, 16 (Fig.3b)	GraphNet / predictive graph layer; Fix T=7 when sweeping L
	Feature dimension	d_feat	270	features	–	All; day[0:5] + hour[5:30] + min[30:270]
	Hidden dimension	d_model	256	–	–	All
	Loss weight beta	beta	10 (CSI300) / 5 (others)	–	–	All; higher beta on CSI300 only
	Learning rate	lr	1e-5	–	–	FinKan/main.py; main1.py uses 1e-6 for graph ablation
	Training epochs	n_epoch	20	epochs	–	All
	Random seed	seed	3	–	–	Reported supplementary experiments use seed 3 for reproducibility
	Early-stop threshold	train_stop_loss_thred	0.95	–	–	All; validation-based stopping
	GPU device	GPU	0	–	–	All
Frequency Mixer	Day branch hidden dim	-	2	–	–	Frequency branch
	Day branch output dim	-	5	–	–	Frequency branch
	Day batch_num	-	1	–	–	Frequency branch
	Hour branch hidden dim	-	32	–	–	Frequency branch
	Hour branch output dim	-	64	–	–	Frequency branch
	Hour batch_num	-	5	–	–	Frequency branch
	Minute branch hidden dim	-	30	–	–	Frequency branch
	Minute branch output dim	-	128	–	–	Frequency branch
	Minute batch_num	-	8	–	–	Frequency branch
Temporal Aggregation	KAN seq_len	-	window_size (T)	–	–	Temporal branch; tied to T
	KAN d_state	-	32	–	–	Temporal branch
	KAN dconv	-	3	–	–	Temporal branch
	KAN e_fact	-	2	–	–	Temporal branch
GraphNet (vanilla)	Neighbor count k	k	10	–	–	FinKAN baseline; matches K_MB budget
	Hidden dim	hidden_dim	256	–	–	FinKAN baseline
	Graph depth	graph_depth	1	–	–	FinKAN baseline
	MLP depth	mlp_depth	1	–	–	FinKAN baseline
	GC alpha	gc_alpha	1	–	–	FinKAN baseline
BMBCG / Markov Blanket	Markov blanket Top-K	K_MB	10	neighbors	5, 10, 20, 30, 50	Causal-Fi / bmbcg; default matches GraphNet k=10
	Bootstrap count	B	50	resamples	–	Graph construction; training split only
	Indirect graph weight	eta	0.50	–	0.3, 0.5, 0.7	Causal-Fi bmbcg; usually fixed at default
	Edge gate relaxation	rho	0.10	–	0.05, 0.1, 0.2	Causal-Fi bmbcg; gate = rho + (1-rho)*prior_mass
	Graph type	graph_type	bmbcg	–	–	Causal-Fi full model; vanilla/corr/direct/direct_indirect/bmbcg
	Dropout (graph layer)	dropout	0.10	–	–	Causal graph layers
DSA Pretraining	Batch size	-	64	samples	–	DSA / NNCL
	Learning rate	-	0.001	–	–	DSA / NNCL
	Epochs	-	20	epochs	–	DSA / NNCL

Continued on next page

Table 10: Aggregated hyperparameter settings grouped by Category (continued).

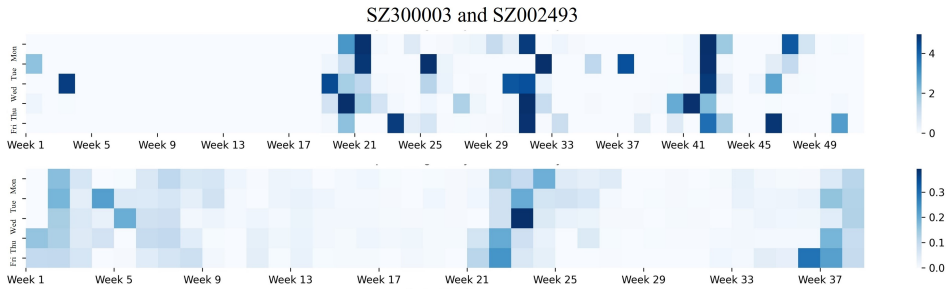
Category	Parameter	Symbol	Default / Value	Unit / Range	Sensitivity Grid	Used In / Notes
	InfoNCE temperature	tau	0.5	–	–	w/o NNCL uses –loss mse; default NNCL loss
	Loss mode	-	infonce	–	–	Ablation Table 4
DSA Inference	DSA batch size	-	4096	samples	–	FeatureIntegrator forward
Dataset Split	CSI100/CSI300 train	–	-	2014-01-01 to 2018-09-30	–	CSI markets; `DATASET_CONFIG`
	CSI100/CSI300 val	–	-	2018-10-01 to 2018-12-31	–	CSI markets
	CSI100/CSI300 test	–	-	2019-01-01 to 2020-09-25	–	CSI markets
	NASDAQ train	–	-	2013-01-01 to 2015-12-31	–	NASDAQ
	NASDAQ val	–	-	2016-01-01 to 2016-12-31	–	NASDAQ
	NASDAQ test	–	-	2017-01-01 to 2017-12-31	–	NASDAQ
	TSE train	–	-	2015-11-01 to 2018-07-31	–	TSE
	TSE val	–	-	2018-08-01 to 2019-06-30	–	TSE
	TSE test	–	-	2019-07-01 to 2020-08-31	–	TSE
	CSI stock count	-	284	stocks	–	CSI100/CSI300
	NASDAQ stock count	-	1026 (auto-detect)	stocks	–	NASDAQ
	TSE stock count	-	95	stocks	–	TSE
Sensitivity (Figure 3)	T sweep grid	-	7;14;30	days	7;14;30	CSI100 + CSI300; IC and RankIC per point
	L sweep grid	-	4;8;16	–	4;8;16	CSI100 + CSI300; IC and RankIC per point
Sensitivity (Optional)	`K_MB` sweep grid	-	5;10;20;30;50	neighbors	5;10;20;30;50	BMBCG on CSI300; demo in `plot_exp3_exp4_causal_visualization.py`

In summary, the main experiments use a seven-day lookback window, global vector length 16, 270 input features, and hidden dimension 256. The predictive-structural relational branch uses Markov-blanket Top-K 10, 50 bootstrap resamples, indirect weight 0.50, gate relaxation 0.10, and graph-layer dropout 0.10. DSA is pretrained for 20 epochs with batch size 64 and InfoNCE temperature 0.5, then used during inference with batch size 4096. Sensitivity analyses vary lookback window T, global vector length L, and optionally Markov-blanket Top-K K_{MB}.

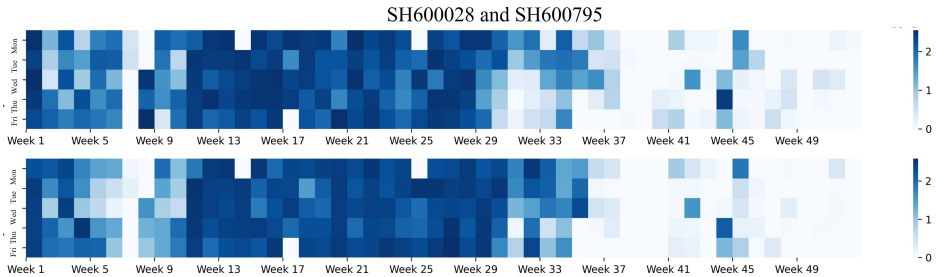
J Adaptive Feature Weight Patterns

Figure 5 shows feature weights learned by the Adaptive Feature Extractor across representative stocks.

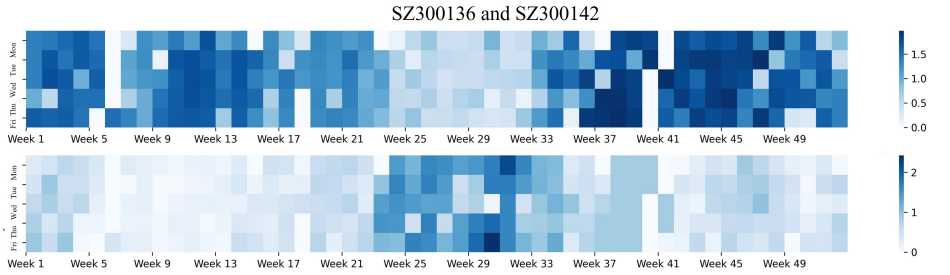
Three patterns emerge. The *cyclical volatility* pattern shows periodic weight changes, possibly reflecting recurring events such as earnings cycles or policy announcements. The *trend-consistent* pattern shows similar weight trajectories across stocks, reflecting shared market exposure. The *difference-complementary* pattern shows heterogeneous weights, suggesting offsetting responses to the same market information. Together, these patterns show that the Adaptive Feature Extractor learns time-varying, stock-specific importance and provides richer representations for cross-stock aggregation.



(a) Cyclical volatility



(b) Trend-consistent



(c) Difference-complementary

Figure 5: Representative feature-weight heatmaps extracted by the Adaptive Feature Extractor, illustrating three qualitatively distinct patterns of stock-level and temporal feature importance.