StockMixer: A Simple Yet Strong MLP-Based Architecture for Stock Price Forecasting

Jinyong Fan, Yanyan Shen

Shanghai Jiao Tong University

The AAAI Conference on Artificial Intelligence

上海交通大學

SHANGHAI JIAO TONG UNIVERSITY

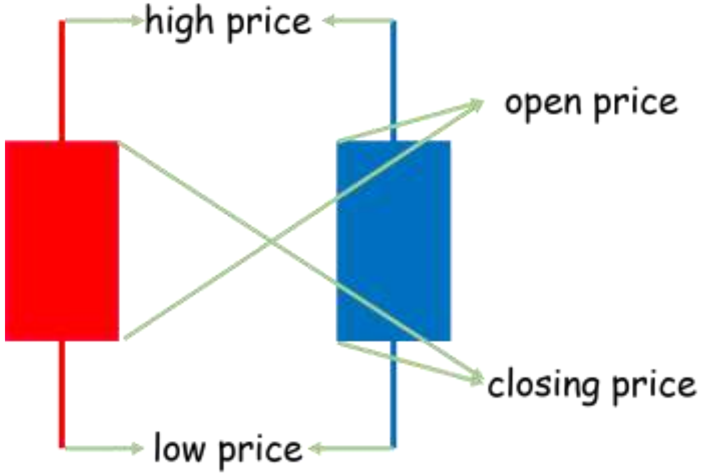
Outline

- Background
- Motivation
- Method
- Experiments
- Conclusion

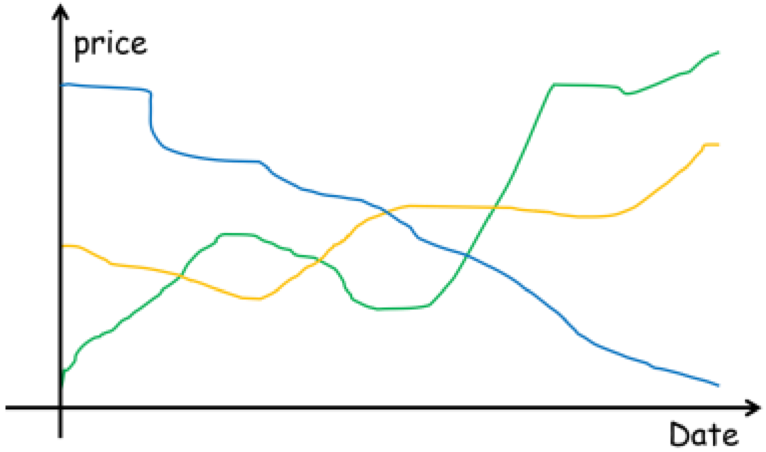
Background

- **Stock price forecasting** is a fundamental task in the field of quantitative investment.
- As the stock market is highly *volatile* and *chaotic*, achieving high forecasting accuracy remains an open question.
- There exist three correlations in stock price data:

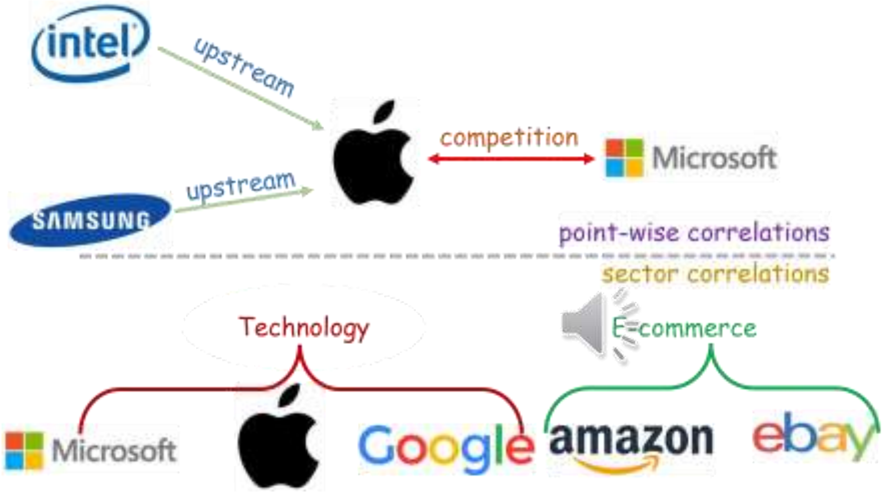
Indicator correlation



Temporal correlation



Stock correlation



Motivation

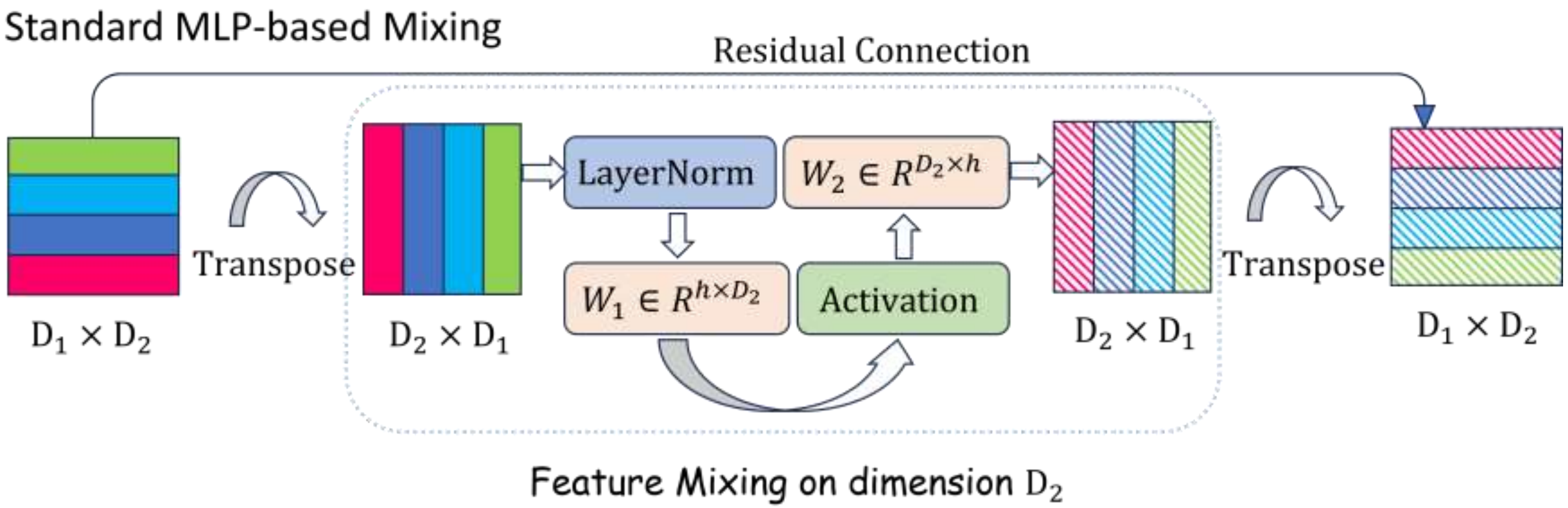
Existing deep learning methods

(1) take parts of the three correlations into account.

(2) adopt hybrid neural architectures, which increases the model complexity and may further hurt the model's generalization ability.

Could we develop a simple neural architecture that is easy to optimize and enjoys strong predictive performance by modeling the above-mentioned correlations effectively?

Motivation: MLP-based Mixing



For inputs $x \in R^{D_1 \times D_2}$, we calculate the embedding $\hat{x}$ mixed on dimension D_2:

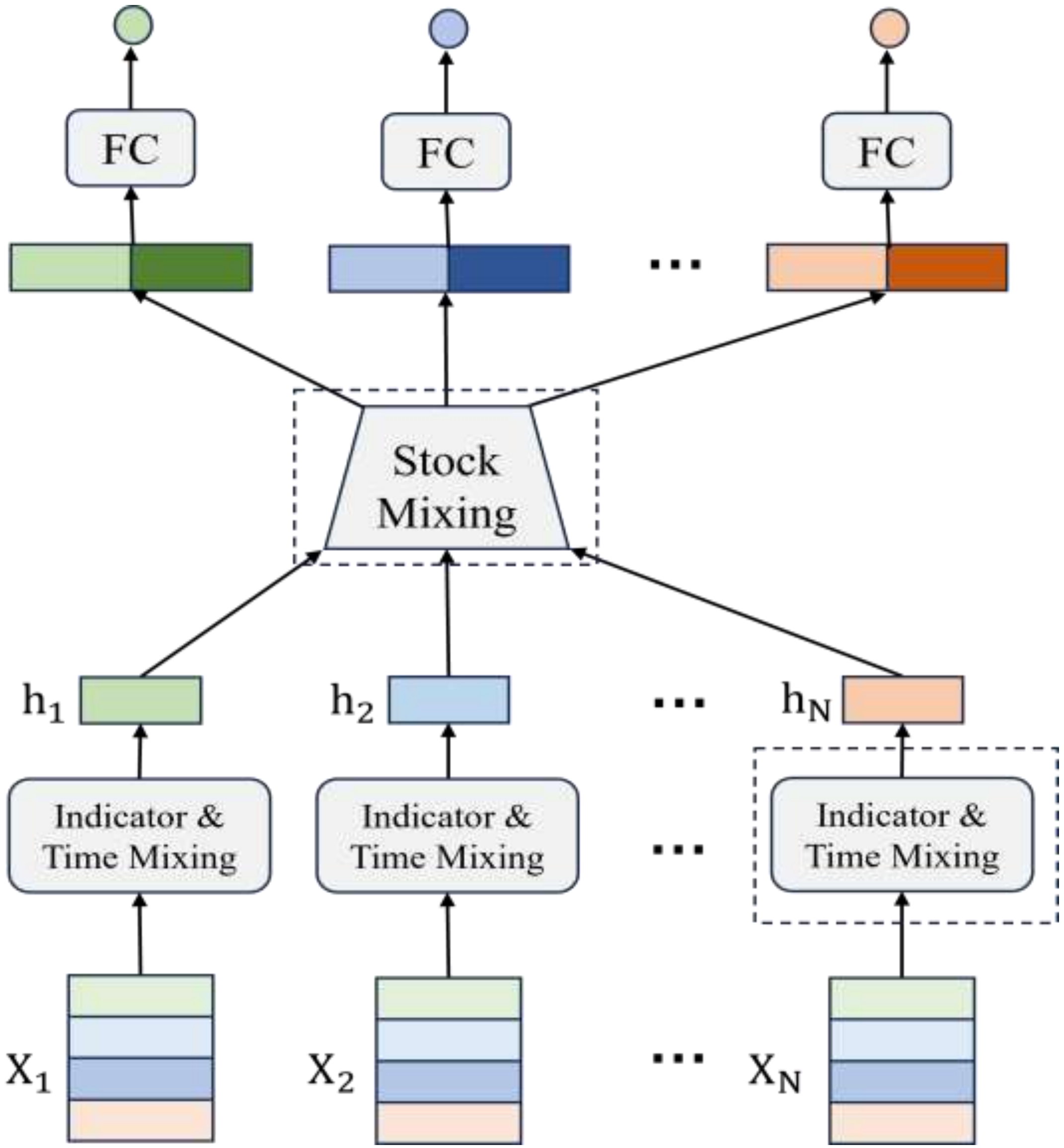
$$\hat{x}^T = x^T + W_2 \sigma(W_1 \mathrm{LayerNorm}(x^T))$$

Insight: The MLP-based mixing

(1) could exchange information in any dimension by matrix transposition.

(2) possess architectural simplicity and linear computational efficiency.

Method: Overview of the proposed StockMixer



- **Inputs:** $X = \{X_1, \dots, X_N\},\ X_i \in R^{T\times F}$
- **Indicator & Time Mixing:**

$$h_i = \text{Indicator\&TimeMixing}(X_i)$$

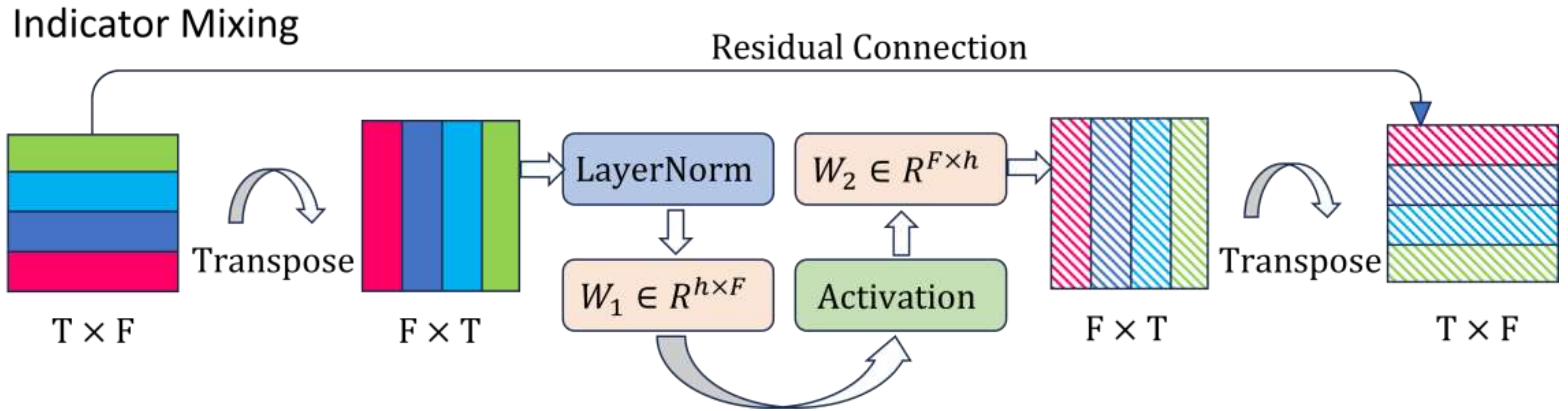
- **StockMixing:** $H = \{h_1, \dots, h_N\}$

$$\widehat{H} = \text{concat}(H, \text{StockMixing}(H))$$

- **Loss function:**

$$L = L_{\text{MSE}} + \alpha \sum_{i=1}^{N} \sum_{j=1}^{N} \max(0, -(\hat{r}_i^t - \hat{r}_j^t)(r_i^t - r_j^t))$$

StockMixer: Indicator Mixing

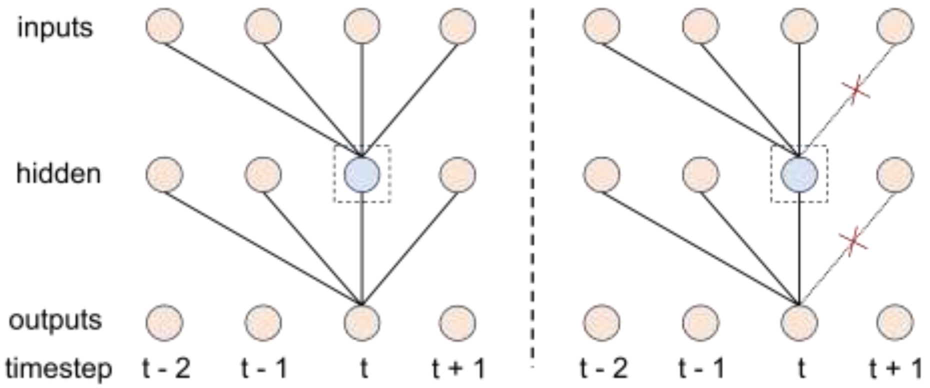


Indicator Mixing is consistent with standard MLP-based mixing.

We calculate the new embedding of a single stock $x \in R^{T\times F}$ by:

$$\hat{x}^T = x^T + W_2\sigma(W_1 \mathrm{LayerNorm}(x^T))$$

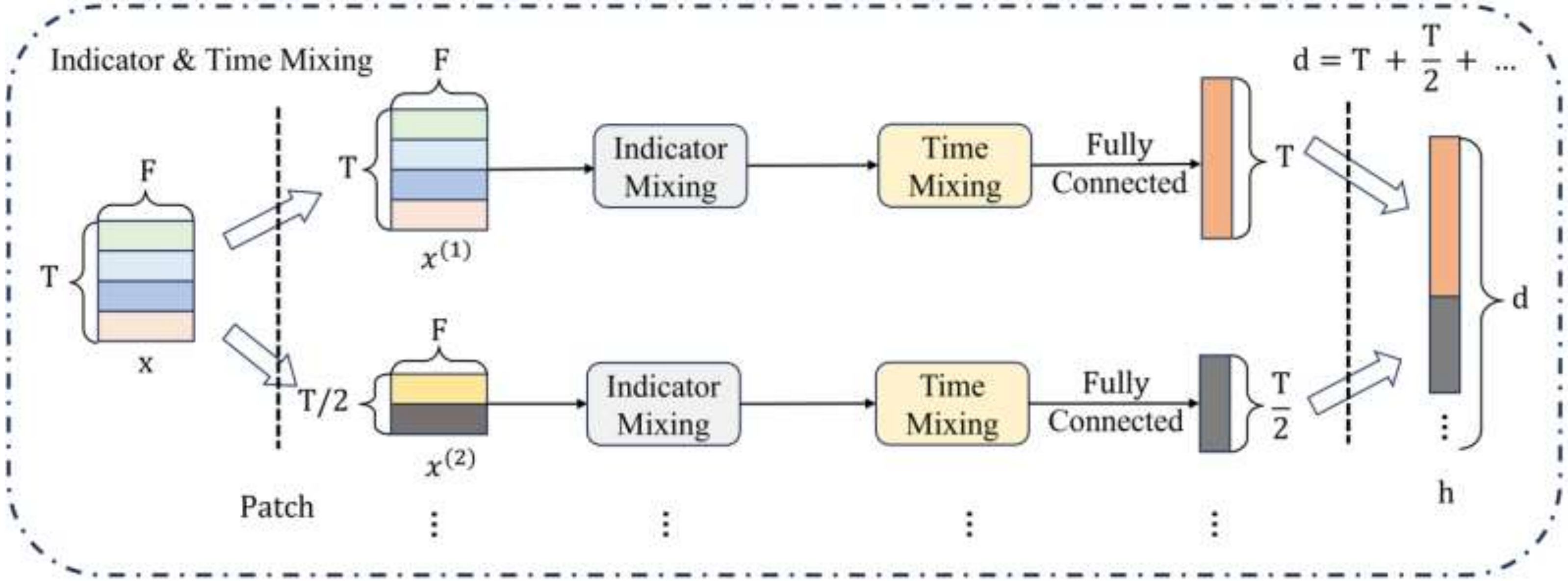
StockMixer: Time Mixing



Standard Mixing (left) vs Time Mixing (right)

Replace the weights with upper triangular matrix U_1, U_2 realizes the process:

$$h = \hat{x} + \mathbf{U}_2\sigma(\mathbf{U}_1\text{LayerNorm}(\hat{x})),$$



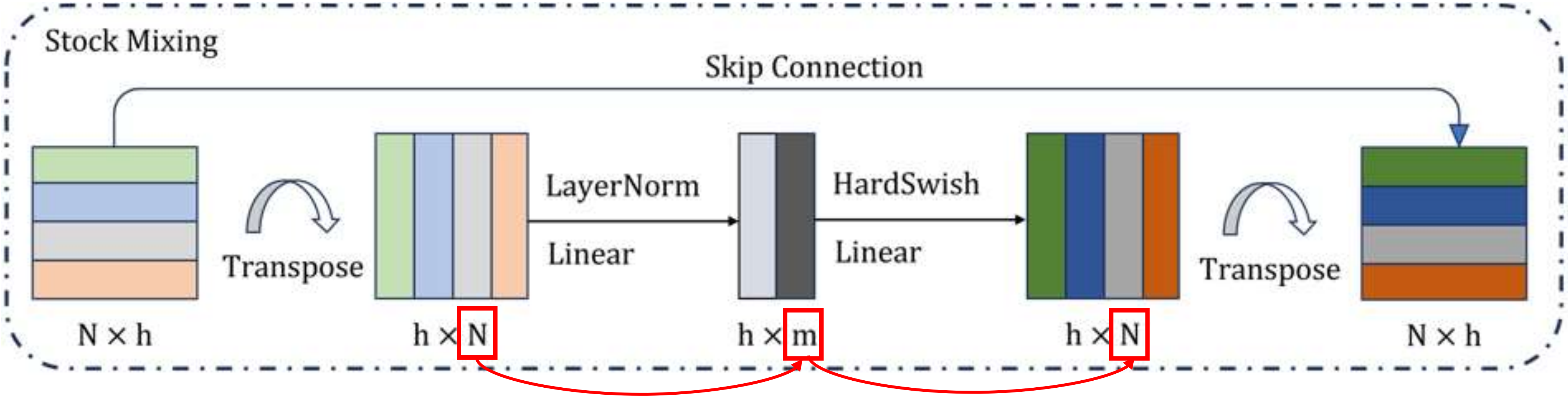
Segment time sequence into subsequence-level patches and mix features at k scales.

$$x^{(k)} = \text{Avgpool}(x)_{\text{kernel}=k}, k \in \{\frac{T}{2}, \frac{T}{4}, \ldots, 1\},$$

$$h^{(k)} = \text{TimeMixing}(\text{IndicatorMixing}(x^{(k)})),$$

$$h = \text{FC}(\text{concat}(h^{(k)})), k \in \{\frac{T}{2}, \frac{T}{4}, \ldots, 1\}.$$

StockMixer: Stock Mixing



- Direct information exchange among stocks → Stock-to-Market and Market-to-Stock
- Replace the hidden dimension of standard mixing related to stocks with a hyperparameter *m* :

$$\hat{H} = H + \mathbf{M}_2 \sigma(\mathbf{M}_1 \mathrm{LayerNorm}(H))$$

Experiments: Setup

- **Datasets:** NASDAQ, NYSE, S&P500
- **Comparison methods:**

  1.RNN-based: LSTM, ALSTM

  2.GNN-based: RGCN, GAT, RSR-I

  3.HGNN-based: STHAN-SR, ESTIMATE

  4.MLP-based: Linear

- **Evaluation metrics:**

  1. Rank-based: Information Coefficient (**IC**), Rank Information Coefficient (**RIC**)

  2. Accuracy-based: **Precision@N**

  3. Return-based: Sharpe Ratio(**SR**)

	NASDAQ	NYSE	S&P500
# Stocks	1026	1737	474
Start Time	13-01-02	13-01-02	16-01-04
End Time	17-12-08	17-12-08	22-05-25
Train Days	756	756	1006
Val Days	252	252	253
Test Days	273	273	352

Table 1: Statistics of datasets.

Experiments: Comparison Performance

- Our method achieves the best results across metrics IC, RIC and SR and fetches an average performance gain of 7.6%, 10.8% and 10.9%.

Model	NASDAQ				NYSE				S&P500			
	IC	RIC	prec@N	SR	IC	RIC	prec@N	SR	IC	RIC	prec@N	SR
LSTM	0.032	0.354	0.514	0.892	0.024	0.256	0.512	0.857	0.031	0.186	0.531	1.332
ALSTM	0.035	0.371	0.522	0.941	0.023	0.276	0.519	0.764	0.029	0.181	0.532	1.298
RGCN	0.034	0.382	0.516	1.054	0.025	0.275	0.517	0.932	0.028	0.175	0.528	1.359
GAT	0.035	0.377	0.530	1.233	0.025	0.297	0.521	1.070	0.034	0.191	0.541	1.484
RSR-I	0.038	0.398	0.531	1.238	0.026	0.284	0.519	0.098	0.033	0.200	0.542	1.437
STHAN-SR	0.039	0.451	0.543	1.416	0.029	0.344	**0.542**	1.228	0.037	0.227	0.549	1.533
ESTIMATE	0.037	0.444	0.539	1.307	**0.030**	0.327	0.536	1.115	0.035	0.241	**0.553**	1.547
Linear	0.019	0.188	0.505	0.517	0.015	0.163	0.497	0.625	0.016	0.156	0.520	0.674
StockMixer	**0.043**	**0.501**	**0.545**	**1.465**	0.029	**0.351**	0.539	**1.454**	**0.041**	**0.262**	0.551	**1.586**

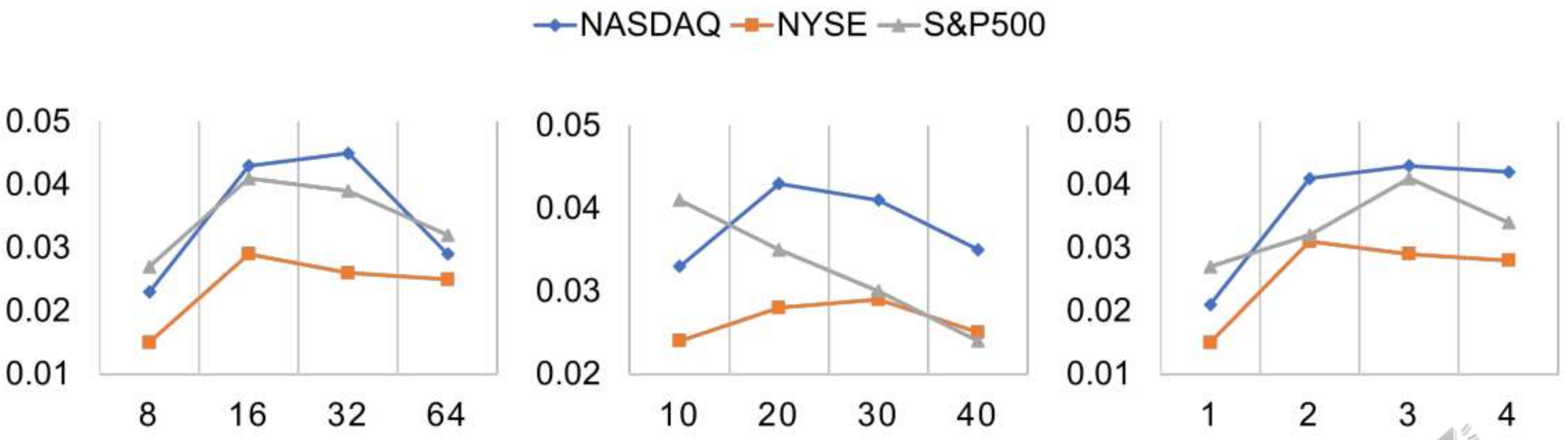
Experiments: Ablation Study

- Ablation study for indicator, time and stock mixings.
- Three mixings jointly contribute to the performance and the time mixing matters most.

Ablation	NASDAQ		NYSE	
Model Component	IC	RIC	IC	RIC
LSTM	0.032	0.354	0.024	0.256
w.o.Indicator Mixing	0.040	0.465	0.027	0.291
w.o.Time Mixing	0.018	0.164	0.016	0.161
w.o.Stock Mixing	0.037	0.376	0.026	0.285
LSTM + Stock Mixing	0.041	0.476	**0.030**	0.307
STHAN-SR	0.039	0.451	0.029	0.344
StockMixer	**0.043**	**0.501**	0.029	**0.351**

Experiments: Hyperparameter Sensitivity

- We focus on the most important hyperparameters and select IC as metric.



(a) Window length (b) Market (c) Scale factor

Conclusion

- We propose **StockMixer**, a simple yet strong architecture with enhanced MLP blocks for stock price forecasting.
- We devise the time mixing to exchange multi-scale time patch information and realize the stock mixing by exploiting stock-to-market and market-to-stock influences explicitly.
- Code: https://github.com/SJTU-Quant/StockMixer

- Email: weizhili@sjtu.edu.cn

Thank you!