RRST: RECURRENT RELATIONAL STOCK TRANSFORMER FOR CROSS-SECTIONAL STOCK RANKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Cross-sectional stock ranking orders hundreds of stocks by their next-day return, and the public benchmarks judge the order two ways: by the correlation of scores with returns over the whole universe, and by the return of the few stocks at the top. We present the Recurrent Relational Stock Transformer (RRST), one recurrent cell that steps over the last days of a window and carries two states per stock: a stock state that carries each stock's representation from one day to the next through a per-channel retention spread over timescales, so that recent days dominate while longer memory is kept, and a relation state holding each stock's loadings on latent factors, trained by an auxiliary loss to explain the next day's cross-section and consumed by the scoring layer as risk exposures. A scoring layer assembles one score for both kinds of metric. On the two public benchmark families, RRST is above all 24 methods reported on NASDAQ and NYSE in cumulative return and Sharpe ratio (SR 4.48 and 7.02 against the best published 2.32 and 2.35) and above StockMixer on all four of its metrics on the same markets, with every hyperparameter chosen by a validation-only rule; the ablation locates the consistent top-of-ranking gain in the relation state used as exposures. We also prove that the benchmarks' price normalisation, by each stock's maximum over the whole series, encodes the future, measure what that column is worth, and report every result with and without the level inputs: NASDAQ and NYSE stay above every published top-of-ranking result in both settings, and S&P500 gains in IC and ICIR with the level inputs.

1 INTRODUCTION

Every trading day a cross-sectional model looks at a few hundred stocks and orders them. Two families of public benchmarks judge that order: one by the cumulative return and Sharpe ratio of buying the five best-ranked stocks each day (Feng et al., 2019; Wang et al., 2025), the other by the daily correlation of scores with returns together with top-of-ranking precision (Fan & Shen, 2024). Twenty-odd architectures have been evaluated on them (Sawhney et al., 2021; Xia et al., 2024; Li et al., 2024b; Wang et al., 2025), and three things stand out. First, in the methods evaluated on them, the relations between stocks are either supplied in advance as fixed graphs or recomputed from scratch every day by attention over the cross-section; neither carries an estimate of the structure from one day to the next and revises it. Second, where recurrence over days is used at all, its state is largely overwritten at each step; on a cell of that kind we measure that the gradient reaching the state after the first step is 0.3% of that at the last step, against 6% for ours. Third, the benchmarks judge one score two ways, by its correlation with returns and by the very top of the ranking, and the two do not reward the same forecast.

RRST (Figure 1) is built around these three points. A single recurrent cell steps over the last T_c days of a window and carries a *stock state* S_t and a *relation state* $R_t \in \mathbb{R}^{N\times r}$, each stock's loadings on r latent factors. The stock state has a per-channel retention initialised over a spread of timescales, so that its temporal Jacobian keeps a term that no decoder can remove; the relation state is trained by a *relation loss* to explain the next day's cross-section, biases the cross-stock attention, and is consumed by the scoring layer as the risk exposures the forecast is adjusted for. Each step reads a sub-window that is historical as of its own day, with a risk model refitted on it, and is trained to forecast the day after it; both states are re-initialised for every prediction window, so the recurrence carries structure

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.

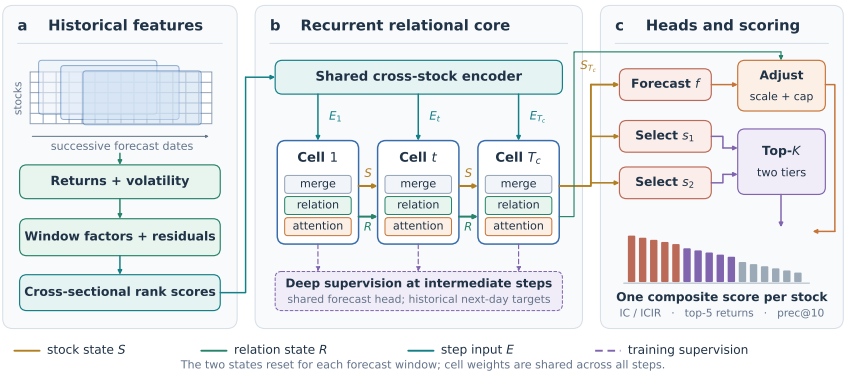

Figure 1: RRST overview. (a) Overlapping historical sub-windows produce risk features and cross-sectional ranks. (b) Shared cells carry stock and relation states; dashed paths indicate training supervision. (c) Final states feed the heads and exposure-adjusted composite score.

within a window rather than indefinitely across dates. A scoring layer assembles one score for the two kinds of metric from the exposure-adjusted forecast and two selection heads.

Three findings frame the paper. RRST is above all 24 methods reported on NASDAQ and NYSE in cumulative return and Sharpe ratio, and above StockMixer on all four of its metrics on the same markets, under a validation-only selection rule. The ablation locates the one gain that is consistent across markets in the relation state used as exposures; cross-stock attention pays on NASDAQ, the leaky cell's advantage there lies within the noise of one test year, on NYSE the arms without them earn more return, and no module removal lowers the IC by more than 0.003. The audit proves that the files' full-series normalisation encodes the future and measures what the column is worth; without it, RRST stays above every published top-of-ranking result on NASDAQ and NYSE, while S&P500 shows no clear predictive gain.

Contributions. (i) *A recurrence over trading days for cross-sectional models*: one cell that attends over stocks within a day and carries N per-stock states through a leaky multi-timescale update, with an exact decomposition of its temporal Jacobian into the retention's own term and what the decoder learns (Theorem 1). (ii) *A relation state with its own objective*, a recurrent estimate of each stock's factor loadings trained to be a forward-looking factor model of the day being predicted and used as the exposures of the scoring layer, where the ablation locates the gain; and *a scoring layer for two kinds of metric*, with the identity that says why they reward different things. (iii) *Results and an audit*: above every published method on both families on NASDAQ and NYSE; a proof of what the future-normalised level reveals, its measured worth, and every result with and without it.

2 Related Work

Cross-sectional stock ranking. Feng et al. (2019) introduced the NASDAQ/NYSE benchmark with temporal graph convolution over industry and Wiki relations and a pairwise ranking loss. Subsequent methods use hypergraph attention (Sawhney et al., 2021; Xia et al., 2024), long–short pattern transformers (Wang et al., 2022), data-axis transformers (Yoo et al., 2021), market-guided transformers (Li et al., 2024a), bilinear ranking (Li et al., 2024b), and specialised pre-training (Wang et al., 2025). StockMixer (Fan & Shen, 2024) and TimeMixer (Wang et al., 2024) are mixing-based references; RRST uses no external relation data.

Recurrent architectures. Repeated Jacobians can attenuate or amplify recurrent gradients (Pascanu et al., 2013). Gated recurrent networks (Hochreiter & Schmidhuber, 1997; Cho et al., 2014) and structured or diagonal state-space models (Gu et al., 2022; Orvieto et al., 2023; Gu & Dao, 2023) provide mechanisms for controlling memory. Feedback and recurrent-memory transformers reuse earlier representations (Fan et al., 2020; Bulatov et al., 2022). The Recurrent Looped Transformer (Zhang et al., 2026) connects a parallel causal encoder and recurrent decoder through a gated state merge. RRST adopts this merge structure across trading days, adds a per-channel leaky stock-state update, and maintains a second recurrent state for relations. Each ingredient has a precedent; RRST's contribution is their integration for a cross-section of stocks, with an analysis and an ablation of what each adds.

Inducing attention, risk models, and leakage. Two-hop attention through learned queries follows the inducing point construction of the Set Transformer (Lee et al., 2019). Forecast neutralisation and factor models are standard tools in portfolio analysis (Grinold & Kahn, 2000; Connor & Korajczyk, 1986; Fama & French, 1993); reversal and momentum are established cross-sectional signals (Jegadeesh, 1990; Jegadeesh & Titman, 1993). Nonlinear factor exposures also appear in autoencoder asset pricing models (Gu et al., 2021). Data leakage and backtest overfitting are longstanding evaluation concerns (Kaufman et al., 2012; Bailey et al., 2014; López de Prado, 2018). Section 7 gives a precise instance.

3 Problem setting

Let $X_{t,i} \in \mathbb{R}^F$ contain the observations of stock i on day t, with a validity flag, closing price $P_{t,i}$, and return $y_{t,i} = P_{t,i}/P_{t-1,i} - 1$. A model maps the W days ending on day $t-1$ to scores $s_t \in \mathbb{R}^N$; *non-anticipation* requires that s_t depend on days $\le t-1$ only, which row indexing alone does not ensure when preprocessing uses future data.

The benchmarks report mean daily Pearson correlation $\mathrm{IC} = \mathrm{mean}_t \mathrm{IC}_t$, its ratio to daily standard deviation $\mathrm{ICIR} = \mathrm{mean}_t \mathrm{IC}_t / \mathrm{std}_t \mathrm{IC}_t$, and precision among the ten highest-scored eligible stocks, where a hit is $y_{t,i} \ge 0$. For the daily equal-weight top-5 return ρ_t, they report $\mathrm{IRR} = \prod_t (1+\rho_t) - 1$ and $\mathrm{SR} = \sqrt{252}\,\mathrm{mean}_t \rho_t / \mathrm{std}_t \rho_t$. The evaluators' own conventions are kept (Appendix A, Q18); the identities of Section 5 refer to a specified cross-section.

4 Method

Notation: T_c core steps, L the lookback of one step, d the state width, r the relation width. The raw window spans $L + T_c - 1$ days.

4.1 Historical sub-windows and their risk model

Core step $j \in \{0, \dots, T_c - 1\}$ reads the L days ending on window day $j + L - 1$ and is responsible for predicting day $j + L$:

$$z^{(j)} = \Phi\big(X_{[j,\, j+L)}\big), \qquad j = 0, \dots, T_c - 1, \tag{1}$$

so no step reads a row after its own forecast origin. Φ first fits a risk model on the sub-window alone: log returns $\rho_{t,i} = \log(P_{t,i}/P_{t-1,i})$, closes recovered from moving-average columns where the file has them ($P_{d-k} = P_d - k(\mathrm{MA}_k[d] - \mathrm{MA}_k[d-1])$, which lengthens the return history within the sub-window), a volatility v_i, the k leading statistical factors $\psi \in \mathbb{R}^{L_h \times k}$ as the top eigenvectors of the $L_h \times L_h$ Gram matrix of volatility-standardised returns (over the L_h available return observations, so they do not depend on stock order), least-squares loadings $B = \rho^\top \psi$ and residual returns $\varrho = \rho - \psi B^\top$. From these it builds, per stock, the feature groups of Table 7: window-relative values, the last return and residual return, the level columns, volatility and return shares, a reversal-strength pair, characteristics (standardised last moves, market beta, residual variance share) and, when the file has open/high/low/volume, scale-free combinations of them. Every feature is then replaced by its rank among the stocks valid on that day, mapped to a fixed rank scale, optionally through the normal quantile, so the network sees only cross-sectional orderings. The only group that is not a function of returns or ratios is the level group (c), the file's own values on the last day, whose ordering can still carry the future information of Section 7.

4.2 The recurrent cell

The T_c feature vectors are embedded and passed through a cross-stock encoder in parallel. There is no temporal input or position embedding: order is carried by the recurrence alone. For $t = 1, \dots, T_c$ (Figure 2), with $\bar{S} = \mathrm{RMSNorm}(S_{t-1})$ (Zhang & Sennrich, 2019),

$$U_t = E_t + \alpha\, \sigma(W_g [E_t; \bar{S}]) \odot W_s \bar{S}, \tag{2}$$

$$\beta_t = \sigma(W_\beta [U_t; \bar{S}; R_{t-1}]), \qquad R_t = (1 - \beta_t) \odot R_{t-1} + \beta_t \odot \tanh(\mathrm{MLP}[U_t; \bar{S}]), \tag{3}$$

$$\Delta_t = \mathrm{Dec}(U_t; R_t), \qquad S_t = \lambda \odot S_{t-1} + \eta \odot \Delta_t. \tag{4}$$

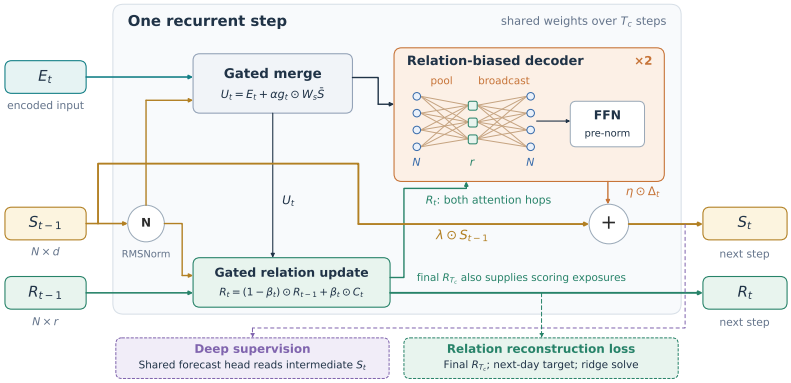

Figure 2: The recurrent cell. The gated merge feeds the relation update, and the relation state biases both attention hops. The decoder update and stock carry form S_t (Eqs. 2–4). Dashed paths show the intermediate forecast and final relation objectives. $C_t = \tanh(\mathrm{MLP}[U_t; \bar{S}])$.

Eq. 2 is how the previous stock state enters the current representation, the gated merge of the Recurrent Looped Transformer (Zhang et al., 2026) applied across days; Eq. 3 evolves the relation state, whose gate β_t is produced from the current step, so the structure adapts within the window while being carried across steps. The decoder is a two-hop attention through the r factors followed by a pre-norm feed-forward block: in hop 1 factor k pools the stocks with weights $\mathcal{A}_{kj} = \mathrm{softmax}_j\big(q_k^\top K_j/\sqrt{d_h} + \lambda_{\mathrm{in}} R_{jk}\big)$, in hop 2 stock i reads the factors with $\mathcal{B}_{ik} = \mathrm{softmax}_k\big(Q_i^\top K'_k/\sqrt{d_h} + \lambda_{\mathrm{out}} R_{ik}\big)$; the relation state biases both hops. Eq. 4 is a per-channel diagonal recurrence with retention $\lambda \in (0,1)^d$ and gain η; λ_i is initialised log-uniformly over $[0.05, 0.95]$ across the d channels, and η_i is initialised to 0.5. Both are learned, with λ constrained by a sigmoid and η unconstrained. Setting $\lambda = 0$ gives the overwriting cell: dependence on S_{t-1} then survives only through the merge and the relation state.

4.3 Heads and the scoring layer

A shared RMSNorm and bias-free linear map read S_{T_c} to produce a forecast f, selection scores s_1, s_2, and an auxiliary output. The outputs are centred over valid stocks, which makes score-proportional weights dollar-neutral but not the long-only top-K portfolio.

Let Z_t contain standardised log volatility, its square, and optionally ranked historical momentum; let B_t contain the selected statistical factor loadings. Write $\mathcal{N}_G(f)$ for the residual from regressing f on $[\mathbf{1}, G]$. The implementation uses a small ridge term for these solves and forms

$$\begin{aligned} e_t^{(B)} &= (1-\gamma)\mathcal{N}_{Z_t}(f_t) + \gamma \mathcal{N}_{[Z_t, B_t]}(f_t), \\ e_t &= (1-\gamma_R) e_t^{(B)} + \gamma_R \mathcal{N}_{[Z_t, B_t, R_{T_c}]}(f_t). \end{aligned} \tag{5}$$

The first selection head optionally undergoes the same full-exposure regression at strength γ_R. We call e_t the *exposure-adjusted forecast*; exact orthogonality holds for full-strength, unregularised projection, not for the implemented partial adjustment.

The scoring layer standardises e_t, caps its upper tail at c, and assigns the top K slots in two tiers. The first $\lceil K/2 \rceil$ stocks are selected by s_1, trained on volatility-scaled returns y/v^p; the remaining slots are selected by s_2, trained on the event $y \geq 0$. Slot s receives $c + \delta(K - s + 1)/K$, ensuring that selected stocks outrank the capped forecast (Lemma 1). With $K = 10$ the first tier determines top-5 returns and both tiers precision@10; IC and ICIR read the composite score, splice included.

4.4 Training

The loss combines negative daily IC of the exposure-adjusted forecast, two selection losses, and deep supervision (Lee et al., 2015): the *same* forecast head reads every state, and the state after step j is trained on the return of the day after step j,

$$\mathcal{L}_{\mathrm{deep}} = -\frac{1}{\sum_j w_j} \sum_{j=0}^{T_c-2} w_j \, \mathrm{IC}\big(e(S_{j+1}),\, y_{j+L}\big), \qquad w_j = \min(1, (j+1)/2). \tag{6}$$

For $j \leq T_c - 2$ the target day $j + L$ lies inside the window, so only the last step's target is the day being predicted. Eq. 6 requires the historical row boundaries in Eq. 1: with whole-window features, step j's input would already contain its own target. The relation state receives an auxiliary reconstruction objective: with $\tilde{y}_t \in \mathbb{R}^N$ the next-day returns divided by the model's volatility estimate, clipped and centred over the valid stocks, and P_R the orthogonal projector onto the column span of R, the *relation loss* is the share of the cross-section the r learned loadings leave unexplained, $\mathcal{L}_{\text{rel}} = \|(I - P_{R_{T_c}})\tilde{y}_t\|^2 / \|\tilde{y}_t\|^2$ (7), with the volatility estimate detached. Gradients reach the relation state and its upstream network parameters. Proposition 2 says what it trains; how the loss and the use of R_{T_c} in the scoring layer were chosen, on validation data among 23 candidates, is recorded in Appendix K. Further terms are differentiable stand-ins for the two top-of-ranking metrics on the selection scores, an optional ListNet term (Cao et al., 2007) on s_1, and an auxiliary output trained on the next-day return net of the forecast's own exposures. Training uses AdamW (Loshchilov & Hutter, 2019) with a one-cycle schedule (Smith & Topin, 2019), 25 epochs, batches of 16 randomly shuffled forecast dates with a random half sub-universe of stocks each, gradient clipping and an exponential moving average of the weights, which is what is evaluated; training-only runs report the epoch with the lowest validation loss, refitted runs the epoch count of their training-only run.

4.5 Hyperparameter selection

One model design serves every dataset; a few hyperparameters (the target p of s_1 and the ListNet term, the risk model's number of statistical factors and volatility statistic, and on S&P500 also the optional input and loss groups, the lookback, the training length and γ_R) are chosen per dataset by a rule that reads validation data only. Each candidate is trained with several seeds and the winner maximises $J = \min_q (\text{validation } q)/(\text{reference } q)$ over the reported metrics q, the reference being the published value the comparison has to beat, ties within 0.01 going to the higher validation IC. Whether to refit on training plus validation days is decided one year earlier, with every split moved one period back; on S&P500 every candidate is scored on two validation years. No candidate test metric enters the rule (Appendix G).

5 Theoretical analysis

Proofs are in Appendix D; where the released implementation departs from the object a statement names, we say so next to it.

The input map. Let $\mathcal{D}$ rescale a stock's price columns by $\theta_i > 0$ and its volume group by $\vartheta_i > 0$.

Proposition 1 (Invariances of the input map)**.** *Let Φ_0 be the feature map without the level group (c), with the validity and close-recovery decisions held fixed and no absolute floors. Then $\Phi_0(\mathcal{D}X) = \Phi_0(X)$ for every such $\mathcal{D}$; and, with or without group (c), the rank scores of a feature are invariant to any strictly increasing transformation applied to it across the stocks of a day. With group (c) included, Φ is not invariant to $\mathcal{D}$.*

Under these assumptions, per-stock normalisers cancel from the remaining features. The released Φ also uses absolute floors and a recovery acceptance window ($10^{-6} < P_{\text{recovered}} < 1.05$), whose decisions can depend on scale. Removing the explicit level group therefore defines a *no levels* ablation, not a certificate of implementation-level non-anticipation; the effect of these numerical decisions has not been quantified.

The recurrence. Stack the per-stock coordinates of S_k and R_k into the joint state $x_k = (\text{vec}(S_k), \text{vec}(R_k))$ and let $G_k = \partial x_k / \partial x_{k-1}$. Write $G_k = A_k + \Xi_k$, where $A_k = \text{diag}\big(I_N \otimes \Lambda,\ \text{diag}(1 - \beta_k)\big)$ is the *carry*, the part of the update that is diagonal in both states, and Ξ_k collects every path through the merge, the decoder and the relation gate.

Theorem 1 (Direct path of the leaky recurrence)**.** *Let $t < T$ and $m = T - t$. (i) $\partial x_T / \partial x_t = \prod_k A_k + \mathcal{X}$, where $\mathcal{X}$ is the sum of all products with at least one Ξ_k factor, and for every stock coordinate ν in channel i, $[\prod_k A_k]_{\nu\nu} = \lambda_i^m$, a term set by the retention alone. (ii) For an overwriting update ($\lambda = 0$) the stock block of $\prod_k A_k$ vanishes, so every term of the stock-to-stock block of the Jacobian contains a Ξ_k factor. (iii) If $\|\Xi_k\|_2 \leq \varepsilon$ for all k and $\bar{a} = \max_k \|A_k\|_2$, then $\|\partial x_T / \partial x_t\|_2 \leq (\bar{a} + \varepsilon)^m$ and $\big|[\partial x_T / \partial x_t]_{\nu\nu}\big| \geq \lambda_i^m - \big[(\bar{a} + \varepsilon)^m - \bar{a}^m\big]$.*

Parts (i) and (ii) are identities: the leaky cell keeps, in every stock channel, a gradient path of size λ_i^{T-t} from the state after step t to the final state that passes through the decoder at no step, whatever the decoder learns, and the overwriting cell keeps none. Part (iii) is the two-sided bound; it needs the learned feedback to be small, which we do not enforce, so it says what (i) guarantees and what is left to training; the measured counterpart is in Section 6.2. Deep supervision adds to every step a loss term whose gradient passes through no temporal Jacobian (Corollary 1). The carry coefficients are geometric: over the $T_c = 8$ steps of a window, channel i averages over an effective $n_i(8) \in [1.1, 7.9]$ steps and gives the last step the share $\omega_i = (1-\lambda_i)/(1-\lambda_i^8)$, on average 0.70 over the initial spread against 0.18 for a shared $\lambda = 0.9$ (Proposition 6): recency and long memory sit in different channels.

The cross-stock attention. For each head, the product $\mathcal{M}_h = \mathcal{B}_h\mathcal{A}_h \in \mathbb{R}^{N\times N}$ of its two softmax weight matrices is row-stochastic, has rank at most r, is non-expansive in ℓ_∞, costs $O(Nrd)$ and depends on R_t through both hops, and the model is equivariant to stock permutations when the prior R_0 is permuted alongside (Proposition 7); the relation state thus controls a rank-r mixing of the cross-section, a statement about the attention weights, not the decoder's Jacobian.

Proposition 2 (What the relation loss trains)**.** *Let $\hat{y}_t = \tilde{y}_t/\|\tilde{y}_t\|$ for nonzero $\tilde{y}_t$, and let $\Sigma_t = \mathbb{E}[\hat{y}_t\hat{y}_t^\top \mid \mathcal{F}_t]$, the conditional second moment of the direction of the next day's standardised cross-section given the window $\mathcal{F}_t$ the cell reads, have eigenvalues $\mu_1 \geq \cdots \geq \mu_N$. For every rank-$r$, $\mathcal{F}_t$-measurable loading matrix R with exact orthogonal projector P_R, $\mathbb{E}[\mathcal{L}_{\mathrm{rel}}(R) \mid \mathcal{F}_t] = 1-\mathrm{tr}(P_R\Sigma_t) \geq 1-\sum_{k\leq r}\mu_k$, with equality exactly when the column span of R is a leading r-dimensional eigenspace of Σ_t (unique when $\mu_r > \mu_{r+1}$). For u orthogonal to R and $\tilde{y}_t = Ra_t + \epsilon_t$, $u^\top\tilde{y}_t = u^\top\epsilon_t$.*

The relation loss thus makes R_{T_c} a *forward-looking* factor model, estimated from the window alone, whereas the factors of Section 4.1 are the backward-looking ones of the window's own returns, and a forecast made orthogonal to R_{T_c} earns none of the learned factors' variance. The implementation solves a ridge-regularised projection (10^{-2}) jointly with the other losses, so we read R as predictive loadings rather than as identified economic factors (the long-only top-K portfolio is not made factor-neutral by it).

The objective. For a specified cross-section, let $\tilde{s}_t = (s_t - \bar{s}_t\mathbf{1})/\|s_t - \bar{s}_t\mathbf{1}\|$ be the unit-norm, dollar-neutral weights proportional to the centred scores and $D_t = \|y_t - \bar{y}_t\mathbf{1}\|$ the day's return dispersion. The return of the portfolio $\tilde{s}_t$ is $\tilde{s}_t^\top y_t = \mathrm{IC}_t D_t$ (Proposition 4): maximising the mean IC maximises the dispersion-scaled return of the score-weighted long–short portfolio, ICIR is that portfolio's daily Sharpe ratio when D_t is constant, and the equal-weight top-K return is not a function of IC_t. Hence the two kinds of head: the IC loss is a return objective for a portfolio holding every stock in proportion to its score, whereas the top-K metrics reward a nonlinear functional of the ranking, for which the selection heads and the tiered splice are built. Exposure adjustment removes the specified exposures from that payoff (Proposition 5); whether it raises ICIR or Sharpe ratio is left to the validation rule that selects γ and γ_R.

6 Experiments

Data and protocol. We use the released files and chronological splits of both benchmark families. The stock-selection benchmark (Wang et al., 2025) has NASDAQ (1026 stocks) and NYSE (1737) from Feng et al. (2019), 2013–2017, with 25 daily columns (open, high, low, close, volume and their 5/10/20/30-day averages); its two other markets are not released. The StockMixer benchmark (Fan & Shen, 2024) has the five close columns for NASDAQ and NYSE and open/high/low/volume/close for S&P500 (474 stocks, 2016–2022, test 2021-01 to 2022-05). The NASDAQ/NYSE files of the two families are the same prices and test year, not independent replications (Appendix F). In both families every price column is divided by the stock's maximum over the whole series, test period included, so every published number on them, including StockMixer's and those of the two SSPT variants and the 22 other methods on the stock-selection benchmark, is obtained with information about the test period (Section 7 measures it). The rows and columns marked *no levels* remove the level inputs, the only inputs carrying this normaliser, and show what the model extracts from the past alone; where the leaked column pays, above all in the correlation metrics, they are expected to fall below the as-given numbers.

Table 1: Stock-selection benchmark (Wang et al., 2025): top-5 IRR and SR on the test year against its Table 2 (24 methods; best other: CI-STHPAN). RRST: mean of 3 seeds (s.e. in Table 10); last column: no level inputs.

		SSPT ind	SSPT comb	Best other	**RRST**	RRST, *no levels*
NASDAQ	IRR	0.74	0.82	0.66	**0.90**	1.02
	SR	2.32	2.25	2.01	**4.48**	4.65
NYSE	IRR	0.41	0.60	0.79	**0.92**	1.24
	SR	2.11	2.35	2.14	**7.02**	5.65

Table 2: StockMixer benchmark (Fan & Shen, 2024), its files and splits: IC, ICIR, prec@10 and top-5 SR on the test period. RRST: mean of 3 seeds (s.e. in Table 10); bold: above the published StockMixer value.

Dataset	Model	IC	ICIR	prec@10	SR
NASDAQ	StockMixer	0.0430	0.501	0.545	1.47
	RRST	**0.0598**	**1.108**	**0.613**	**2.71**
	RRST, *no levels*	0.0213	0.368	**0.610**	**3.04**
NYSE	StockMixer	0.0290	0.351	0.539	1.45
	RRST	**0.0614**	**1.328**	**0.746**	**6.05**
	RRST, *no levels*	**0.0356**	**0.744**	**0.740**	**6.79**
S&P500	StockMixer	0.0410	0.262	0.551	1.59
	RRST	**0.0473**	**0.485**	0.547	1.47
	RRST, *no levels*	0.0065	0.061	0.528	−0.11

Main results average three independently trained seeds, without ensembling. Training-only runs select the epoch with the lowest validation loss; the S&P500 refit uses the epoch count of its training-only run, the decision to refit having been taken one fold earlier. On NASDAQ and NYSE the models are therefore fitted on fewer days than the stock-selection benchmark's own entries, which are fitted on training plus validation days (Wang et al., 2025). Reference results are the published tables, not reruns with our feature map, losses or scoring layer. Standard errors in Table 10 describe variability across seeds; Tables 11 and 12 add block-bootstrap intervals over the test days for the main results and for the ablation differences. Default dimensions are $T_c = 8$, $L = 64$, $d = 64$ and $r = 16$, about 2.2×10^5 parameters and 3–12 minutes per run on an RTX 3090.

6.1 Main results

Table 1 places RRST above every one of the 24 methods reported by Wang et al. (2025) on both NASDAQ and NYSE, on both metrics. The Sharpe ratio is the larger effect: 4.48 against the best published 2.32 on NASDAQ and 7.02 against 2.35 on NYSE, and every seed on both markets is above every published SR (Table 10). The cumulative return rises from 0.82 to 0.90 on NASDAQ and from 0.79 to 0.92 on NYSE; the NASDAQ margin is within one seed standard error (0.09), and we read it as level or above, not more. Table 2 makes the same point on the second family with StockMixer's own five close columns: on NASDAQ and NYSE RRST is above StockMixer on all four metrics, with IC 0.060 and 0.061 against 0.043 and 0.029 and ICIR 1.11 and 1.33 against 0.50 and 0.35. On S&P500, where the validation rule set $\gamma_R = 0$, it is above on IC and ICIR (0.047 and 0.49 against 0.041 and 0.26) and level on prec@10 and SR (0.547 against 0.551; 1.47 ± 0.06 against 1.59; Section 7.1). Appendix H draws both tables and gives block-bootstrap intervals over the test days (Table 11).

The *no levels* columns and rows remove the explicit level group, subject to the preprocessing qualifications of Proposition 1. On the stock-selection benchmark the reported means remain strong: IRR 1.02 and 1.24 and SR 4.65 and 5.65, above every published method on both markets, so the top-of-ranking result does not rest on the leaked column. On the StockMixer files NYSE stays above StockMixer on all four metrics; NASDAQ keeps precision and Sharpe ratio (0.610, 3.04) and gives up the IC the level columns carried (0.021 against 0.060); on S&P500 the IC falls to 0.007.

6.2 Analysis of the recurrence and the relation state

Gradient propagation. Theorem 1 says the state path keeps a decoder-free term and Proposition 6 that the last step should still dominate. On the trained models (Table 13, Appendix J) the share of the forecast's gradient reaching the eight core steps' inputs rises from 0.01–0.10 at the oldest step to 0.49–0.63 at the newest, and the state path retains 5–12% of its gradient norm over seven steps; the overwriting cell, which Theorem 1(ii) leaves without a decoder-free path, retains 0.28%, and the shared retention $\lambda \equiv 0.9$ retains 47% but weighs the initial state $40\times$ the last step's input. The recurrence is recency-weighted and trainable; the probe does not verify the condition of Theorem 1(iii).

Relation state under ranking losses alone. We first probe models trained without the relation loss and with $\gamma_R = 0$ (three seeds per market, validation days). The same-industry share among

a stock's ten nearest neighbours in R-space is at chance (0.040 against 0.037 on NASDAQ, 0.028 against 0.021 on NYSE), and the projection onto $\mathrm{span}(R_{T_c})$ explains 4.0% and 4.6% of the next day's standardised cross-section, against 1.6% and 0.9% for random loadings and 11.5% and 13.6% for sixteen window-PCA loadings. Freezing $R_t \equiv R_0$ or removing the relation bias at inference changes validation IC by less than 0.0015, and the bias strengths never leave their initial values: trained by the ranking losses alone, the relation state is carried but not used.

Relation state with the relation loss. In the final models the learned subspace explains 10.2% and 12.8% of the next day's cross-section, close to the window's own PCA (11.5%, 13.6%) although computed before that day; the same-industry neighbour share rises to 0.118 and 0.103, three to five times chance, and relation similarity ranks stock pairs like their return correlation (Spearman 0.15, 0.29). Freezing $R_t \equiv R_0$ at inference now costs 0.0015 and 0.0045 of validation IC and 3.3 of NYSE top-5 SR; switching the exposure adjustment off costs 0.03 and 0.23 of ICIR and 3.4 of NYSE SR while raising NASDAQ IRR by 0.6; removing the attention bias alone changes little. The relation state reaches the score through the scoring layer, where it is a risk model, not through the attention, which routes the same information by content; the probes identify a predictive subspace, not economically named factors.

6.3 Ablation study

Table 3 removes or replaces one module at a time from the final NASDAQ and NYSE configurations (three seeds on the six arms that decide the claims, one on the others), and Table 12 pairs each arm with the full model day by day over the test year. Two arms are matched baselines with the same features, losses and scoring layer: *no cross-stock attention* is a per-stock network and the *single step* ($T_c = 1$) has no recurrence.

The modules act on the top of the ranking. The IC is flat, within 0.003 of the full model for every arm on both markets, because the features and the exposure adjustment that carry it are shared by all arms (Appendix L); the per-stock network loses the most, 0.003 on NASDAQ with an interval excluding zero. The top-5 metrics, which the stock-selection benchmark scores, are not flat. On NASDAQ, removing the leaky carry, the historical sub-windows, cross-stock attention or the relation exposures costs 0.21 to 0.25 of top-5 IRR each, about a quarter of the full model's 0.90, and 0.8 to 1.4 of Sharpe ratio; these costs are positive in 90 to 99% of the paired bootstrap resamples, and the attention arm's Sharpe cost excludes zero. On NYSE the same removals shift the balance between the two top-5 metrics: without the leaky carry or cross-stock attention the top five earn more (0.22 and 0.23 of IRR, intervals excluding zero) at a Sharpe ratio 1.7 lower, without the relation exposures they earn 0.25 less at a higher Sharpe ratio, and no three-seed arm matches the full model on both.

The relation state. The exposure arm keeps the window's own loadings B_t in the regression, so its cost is what the learned loadings add: about a quarter of top-5 return on both markets (0.22 on NASDAQ, 0.25 on NYSE with an interval excluding zero), the largest return cost that holds on both; the single step also costs return on both markets, but only 0.09 and 0.05. Removing the relation loss instead leaves the metrics within seed variability (0.85, 4.3 on NASDAQ): its contribution is the one the probes measure, an interpretable factor model at no cost in accuracy, and the gain comes from consuming it as exposures; freezing the state at R_0 or removing the attention bias costs little. **Single-seed arms** scatter more widely, and several exceed the full model on one metric, most visibly the shared retention on NASDAQ (1.19 at 4.7) and the model without deep supervision on NYSE (SR 11.7 at lower return); against the three-seed mean on the same days every such advantage has an interval that includes zero, so we report them as open and keep the retention spread and deep supervision for the memory and the direct gradient they give.

7 Look-ahead in the benchmark data

Both benchmark families divide every price column by the stock's maximum over the whole series, test period included. Every reported method, and our as-given numbers, is trained and evaluated on those files, so the comparison above is like for like; what the column carries deserves precision.

Proposition 3 (What the level reveals)**.** *For stock i with positive prices $P_{t,i}$ on all days $1..T$, let $\ell_{t,i} = P_{t,i}/M_i$ with $M_i = \max_{s \le T} P_{s,i}$ be the level column and $\ell^{\mathrm{past}}_{t,i} = P_{t,i}/\max_{s \le t} P_{s,i}$ its past-*

Table 3: Module ablation (test top-5 IRR / SR and IC). Full model and arms marked †: mean over seeds 0–2 (s.e. in Table 10); other arms: seed 0. More arms: Appendix L.

Variant	NASDAQ: IRR / SR / IC	NYSE: IRR / SR / IC
RRST (full)	0.90 / 4.48 / 0.066	0.92 / 7.02 / 0.065
cell: overwriting state ($\lambda = 0$)†	0.68 / 3.52 / 0.066	1.14 / 5.30 / 0.064
cell: shared retention ($\lambda \equiv 0.9$)	1.19 / 4.70 / 0.067	0.92 / 10.23 / 0.062
sub-windows: single step ($T_c = 1$)†	0.81 / 4.17 / 0.064	0.87 / 7.19 / 0.063
sub-windows: whole-window features (non-causal)†	0.66 / 3.65 / 0.066	0.99 / 4.84 / 0.063
relation state frozen ($R_t \equiv R_0$)	0.86 / 3.58 / 0.068	0.66 / 10.41 / 0.064
relation state: no relation loss†	0.85 / 4.34 / 0.066	1.06 / 6.55 / 0.065
relation state: not used by the scoring layer†	0.67 / 3.42 / 0.068	0.67 / 9.74 / 0.063
attention: no relation bias	0.94 / 4.30 / 0.066	0.86 / 5.00 / 0.065
attention: no cross-stock attention†	0.64 / 3.04 / 0.063	1.14 / 5.31 / 0.065
training: no deep supervision	0.89 / 4.45 / 0.067	0.71 / 11.69 / 0.063

only counterpart. Then $\ell_{t,i} \le \ell_{t,i}^{\mathrm{past}}$, with equality if and only if no price after day t exceeds every price up to day t, and when the inequality is strict $\log(\ell_{t,i}^{\mathrm{past}}/\ell_{t,i}) = \log(\max_{s>t} P_{s,i}/\max_{s\le t} P_{s,i}) > 0$: the two levels together determine the height of a future new high relative to the past high.

The full-series level is therefore not measurable from the past, whereas its past-only counterpart is (Corollary 2); a model can read the future through this column, and it pays: on the test periods the single feature $-\ell_{t-1,i}$ has IC 0.028 on NASDAQ, 0.015 on NYSE and 0.034 on S&P500, against 0.005, 0.005 and 0.001 for a causal reference over the preceding 250-day maximum (Appendix I). On S&P500 it is the strongest single signal in every period and models exploit it: the as-given RRST has IC 0.047 there and the model without levels 0.007, whereas on NASDAQ and NYSE the top-of-ranking results survive its removal. The flag `--no_levels` removes every explicit level feature (Proposition 1); a fully causal comparison would in addition require rerunning every published method under the same preprocessing.

7.1 Case study: S&P500

On S&P500 the *no levels* model has near-zero test IC, with an interval that includes zero: every candidate the selection rule scored has IC about 0.03 on both validation years and near zero on 2021-01 to 2022-05. Figure 8 shows why: the one-feature signals the training years contain, 5-day reversal and 12–1 momentum, have IC 0.03 in 2020 and near zero in the test period, where only the 1-day reversal (0.016) and a volume spike (0.013) remain, and a quarterly walk-forward refit does not change the picture. These observations indicate limited transfer for the evaluated models and signals, and a substantial dependence of RRST's IC on the explicit level inputs. They neither establish that the dataset has no predictive signal nor identify how much each published baseline relies on the level column. We retain the validation-selected configuration; the search and quarterly refits did not recover a clear predictive gain without levels.

8 Conclusion and limitations

RRST carries a stock state whose retention keeps a gradient path that no decoder can remove, and a relation state trained to be a factor model of the day it predicts and consumed by the scoring layer as risk exposures. It is above every published method on both benchmark families on NASDAQ and NYSE under a validation-only rule, the ablation says where the gain sits, and the audit proves what the benchmarks' future-normalised level reveals and makes asking what remains without it a one-flag change: most of it on NASDAQ and NYSE, no clear predictive gain on S&P500's test period; we recommend that results on these files be reported both ways. **Limitations.** Table 3 has matched arms for recurrence and attention, not a separately tuned alternative architecture. Most evidence comes from one shared NASDAQ/NYSE test year with three seeds; the intervals of Tables 11 and 12 cover its days, not other regimes. The published reference values enter the selection rule as fixed metric weights, and portfolio metrics are gross of costs, impact and capacity.

Ethics statement

This work uses public historical stock benchmarks and involves no human subjects or private data. Full-series normalisation incorporates future observations, and the no-level ablation does not certify the complete absence of preprocessing leakage. Reported portfolio metrics omit transaction costs, market impact, and capacity and should not be interpreted as achievable live-trading returns or investment advice.

Reproducibility statement

The supplementary `RRST_submit` repository contains the model, data loaders, training and checkpoint evaluation scripts, configuration recipes, benchmark data, the vendored StockMixer evaluator, and the implemented stock-selection metrics. It includes one as-given seed-0 checkpoint and its JSON run record for each of the five dataset variants. The README and `configs.json` describe the main training commands, including the no-level settings. The supplementary archive does not include the complete multi-seed experiment archive, automated selection tools, or the dedicated ablation and probe scripts; the corresponding procedures and aggregate results are documented in the appendix, and the full code repository will be released on GitHub after the review period. The scripts that produce the tables and figures from the run records, including the bootstrap of Table 11, accompany the manuscript sources. Training and reference evaluation use CUDA bfloat16, and numerical differences can alter close score orderings and top-K metrics. Exact agreement across hardware and precision settings is not claimed. Proofs and implementation qualifications appear in Appendix D.

AI use statement

An LLM-based assistant was used under the authors' direction to assist with experimental code and manuscript drafting. It also supported methodological critique, checking and revising mathematical statements and proofs, interpreting existing results, and editing text and figure annotations for clarity. The manuscript revision did not change the model or rerun training. The authors take responsibility for the final text, mathematical claims, code, and reported experimental results.

References

David H. Bailey, Jonathan M. Borwein, Marcos López de Prado, and Qiji Jim Zhu. Pseudo-mathematics and financial charlatanism: The effects of backtest overfitting on out-of-sample performance. *Notices of the American Mathematical Society*, 61(5):458–471, 2014.

Aydar Bulatov, Yuri Kuratov, and Mikhail S. Burtsev. Recurrent memory transformer. *Advances in Neural Information Processing Systems*, 35, 2022.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: From pairwise approach to listwise approach. In *International Conference on Machine Learning (ICML)*, 2007.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

Gregory Connor and Robert A. Korajczyk. Performance measurement with the arbitrage pricing theory: A new framework for analysis. *Journal of Financial Economics*, 15(3):373–394, 1986.

Eugene F. Fama and Kenneth R. French. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1):3–56, 1993.

Angela Fan, Thibaut Lavril, Edouard Grave, Armand Joulin, and Sainbayar Sukhbaatar. Addressing some limitations of transformers with feedback memory. *arXiv preprint arXiv:2002.09402*, 2020.

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*, 2024.

Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems*, 37(2):1–30, 2019.

Richard C. Grinold and Ronald N. Kahn. *Active Portfolio Management: A Quantitative Approach for Producing Superior Returns and Controlling Risk*. McGraw-Hill, 2nd edition, 2000.

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

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations (ICLR)*, 2022.

Shihao Gu, Bryan Kelly, and Dacheng Xiu. Autoencoder asset pricing models. *Journal of Econometrics*, 222(1):429–450, 2021. doi: 10.1016/j.jeconom.2020.07.009.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997.

Narasimhan Jegadeesh. Evidence of predictable behavior of security returns. *The Journal of Finance*, 45(3):881–898, 1990.

Narasimhan Jegadeesh and Sheridan Titman. Returns to buying winners and selling losers: Implications for stock market efficiency. *The Journal of Finance*, 48(1):65–91, 1993.

Shachar Kaufman, Saharon Rosset, Claudia Perlich, and Ori Stitelman. Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data*, 6 (4):1–21, 2012.

Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. Deeply-supervised nets. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning (ICML)*, 2019.

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*, 2024a.

Weixian Waylon Li, Yftah Ziser, Yifei Xie, Shay B. Cohen, and Tiejun Ma. TSPRank: Bridging pairwise and listwise methods with a bilinear travelling salesman model. *arXiv preprint arXiv:2411.12064*, 2024b.

Marcos López de Prado. *Advances in Financial Machine Learning*. Wiley, 2018.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Antonio Orvieto, Samuel L. Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. In *International Conference on Machine Learning (ICML)*, 2023.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning (ICML)*, 2013.

Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, 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*, 2021.

Leslie N. Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, 11006, 2019.

Heyuan Wang, Tengjiao Wang, Shun Li, Jiayi Zheng, Shijie Guan, and Wei Chen. Adaptive long-short pattern transformer for stock investment selection. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2022.

Mengyu Wang, Tiejun Ma, and Shay B. Cohen. Pre-training time series models with stock data customization. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations (ICLR)*, 2024.

Hongjie Xia, Huijie Ao, Long Li, Yu Liu, Sen Liu, Guangnan Ye, and Hongfeng Chai. CI-STHPAN: Pre-trained attention network for stock selection with channel-independent spatio-temporal hypergraph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

Jaemin Yoo, Yejun Soun, Yong-chan Park, and U Kang. Accurate multivariate stock movement prediction via data-axis transformer with multi-level contexts. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2021.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Yifan Zhang, Jichen Feng, and Shihan Qin. Recurrent looped transformer. Technical report, updated September 20, 2026.
`https://github.com/yifanzhang-pro/recurrent-looped-tranformer`, September 2026.

A ANTICIPATED QUESTIONS AND ANSWERS

Table 4 lists the paper's claims with the evidence behind each and what that evidence does not establish; the questions that follow expand on them.

Table 4: Claims, evidence, and what the evidence does not establish.

Claim	Evidence	Not established
Above all 24 published methods on NASDAQ and NYSE in top-5 IRR and SR (Table 1)	means over 3 seeds; every seed above every published SR; block-bootstrap SR intervals above the published values (Table 11)	the NASDAQ IRR margin (0.08, within one s.e.); paired tests against the published methods
Above StockMixer on all four metrics on its NASDAQ and NYSE files (Table 2)	intervals above StockMixer for IC, ICIR and prec@10 on both markets and for SR on NYSE	the NASDAQ SR interval includes 1.47; S&P500 prec@10 and SR are level
The benchmarks' level column encodes the future and pays (Proposition 3, Section 7)	proof; single-feature test IC 0.028/0.015/0.034 against 0.005/0.005/0.001 for a causal reference	how much each published method exploits it
Without level inputs the top-of-ranking results stand on NASDAQ and NYSE	last column of Table 1, rows of Table 2, intervals of Table 11	IC on NASDAQ (0.021) and every S&P500 metric; baselines were not retrained without levels
The relation state is a forward-looking factor model that pays as scoring exposures	probes: 10.2%/12.8% of the next day's cross-section explained, same-industry share 3–5× chance; the exposure arm, with the window's PCA loadings kept, costs 0.23/0.25 of IRR on both markets, the NYSE interval excluding zero	a gain through the attention path; economic identification of the factors; the mechanism on S&P500, where the rule set $\gamma_R = 0$
The leaky cell keeps a decoder-free gradient path (Theorem 1)	exact identity; probe ratio 5–12% against 0.28% for the overwriting cell	that the path improves accuracy: the paired return and Sharpe differences against the overwriting cell include zero on NASDAQ, and the return difference reverses on NYSE (Table 12)
Hyperparameters chosen on validation data only	the rule, its record (Appendix G) and the earlier-fold refit decision	reference test values set the metric weights; relation mechanisms chosen after the diagnosis

A.1 BENCHMARK VALIDITY AND INTERPRETATION

Q1. The benchmark files contain future-dependent normalisation. What do the reported comparisons establish?

That RRST is above every published entry under the benchmark's own files and protocol: every reported method reads the same columns, so the as-given comparison is like for like. What the files do not allow is a statement about a verified causal pipeline, and a common input does not ensure that methods exploit the future equally; Section 7 therefore measures the column and reports the model without it. Removing the level inputs keeps IRR 1.02 and 1.24 and SR 4.65 and 5.65 on the stock-selection NASDAQ and NYSE files (Table 1), above every published number, so the top-of-ranking result does not depend on the leaked column; isolating the leak's share of each baseline's number would require retraining the baselines under the same causal preprocessing.

Q2. Does removing the level inputs make the implementation fully look-ahead-free?

No. Proposition 1 establishes cancellation of per-stock normalisers only with fixed validity and recovery decisions and no absolute floors. The implementation's floors and recovery acceptance window are scale-dependent; whether they transmit material future information on these files has not been quantified. The no-level results measure robustness to removing explicit level features,

not performance under verified causal preprocessing. Certification requires auditing these decisions and constructing inputs solely from information available at each forecast origin; historical row boundaries alone are insufficient.

Q3. Are the gains statistically significant with three seeds and one NASDAQ/NYSE test year?

The large numerical gains do not establish statistical significance against the published methods. Table 11 resamples dates jointly across three fitted seeds in ten-day blocks. The stock-selection SR intervals exceed the published references on both markets and in both settings, but omit baseline uncertainty and covariance between methods; most IRR intervals include the references. Table 12 provides paired intervals for full-model versus ablation differences. These are conditional on the fitted models, uncorrected for multiple comparisons, and exclude model-selection uncertainty. Both NASDAQ/NYSE families share one test year; S&P500 and its walk-forward analysis share a second period. Stronger claims require matched baseline daily outputs and additional test periods.

Q4. How should a reader interpret the very high reported Sharpe ratios?

They are gross benchmark metrics. For NYSE seed 0, SR 9.7 corresponds to a mean daily return of 22 bp and standard deviation of 36 bp, positive returns on 78% of days, and a worst day of -0.9%. The portfolio replaces 83% of its names daily. With that turnover, a one-way cost of 5 to 10 bp per trade would consume about 8 to 17 bp of the 22 bp mean daily return under a simple turnover calculation. This is a sensitivity illustration, not a cost-adjusted backtest; execution, impact, and capacity are not modelled.

Q5. Does the weak S&P500 result show that the dataset has no predictive signal?

No. The selected no-level model has IC 0.0065, with an interval spanning zero (Table 11); the search and quarterly refits did not recover a clear gain on 2021-01 to 2022-05. This establishes limited transfer for the evaluated pipeline, not failure of all predictors. Indeed, 1-day reversal and a volume spike have positive test IC point estimates (0.016, 0.013). The drop from RRST's as-given IC 0.0473 indicates sensitivity to explicit levels. Attributing any published baseline's performance to those inputs requires matched preprocessing ablations of that baseline.

A.2 Architecture, objectives, and theoretical scope

Q6. What is the contribution beyond existing recurrent, attention, and factor models?

The ingredients have precedents, which the paper cites: the gated encoder–decoder merge follows the Recurrent Looped Transformer, the stock update is a diagonal recurrence, the attention uses inducing points, and the risk model is classical. What is new is their integration for a cross-section of stocks and three elements without a precedent in this literature: a relation state that is a recurrent, forward-looking factor model of the day being predicted (Proposition 2) and is consumed by the scoring layer as exposures; the causal sub-window construction with a risk model refitted per step and deep supervision on past days; and the audit of the benchmarks' level column. Table 5 sets RRST against the closest design element by element. We do not claim a new general factor-model principle.

Q7. Could the results come from feature engineering or scoring instead of the recurrent architecture?

Yes. Table 3 retains the feature and scoring machinery while removing attention or using one step; the latter also removes intermediate-step supervision. The single-step NASDAQ IRR/SR deficits have paired intervals spanning zero, although its small IC deficit excludes zero. On NYSE, single-step gives *lower return and higher Sharpe* ($0.87/7.19$ versus $0.92/7.02$), with neither difference excluding zero. Removing attention lowers NASDAQ SR/IC but increases NYSE return, with these intervals excluding zero. Additional single-seed arms show substantial feature and scoring effects (Appendix L). Thus the published-method comparisons support the complete pipeline, while architectural effects are market-dependent. Separately tuned simpler pipelines remain necessary to justify the complexity; the gradient theorem cannot establish a predictive advantage.

Q8. What evidence supports the relation state if its attention contribution is small?

Its use as scoring exposures: removing it is the one ablation arm that costs cumulative return on both markets (0.23 and 0.25 of IRR, Table 3), and the probes show why it can serve as a risk model. With the relation loss the learned subspace explains 10.2% and 12.8% of the next day's cross-section, more than twice the value without the loss, and its neighbourhoods are same-industry at three to five times chance. On the 25-column NASDAQ files without levels, IC is 0.032 with the relation loss (Table 10c) against 0.021 for the diagnosis models trained without it (not tabulated). Removing the attention bias or the relation loss alone has no consistent effect on ranking, so the paper attributes no gain to the attention path; the probes identify a predictive subspace, not named economic factors, and do not show that it exceeds window PCA.

Q9. What does Theorem 1 actually guarantee?

Parts (i) and (ii) are identities that hold for every trained network: the temporal Jacobian of the joint stock/relation state splits into the product of the carries and terms that pass through the learned maps, the carry product contributes exactly λ_i^{T-t} to each stock channel, and an overwriting update has no such term. Part (iii) turns this into a two-sided bound under a bound ε on the learned part; we neither enforce nor verify that bound, so (iii) states what the identity guarantees and what it leaves to training. Table 13 is the measurement: the leaky cells retain 5–12% of the state gradient over seven steps, the overwriting cell 0.28%. The finite-window effective sample size of Proposition 6 describes the geometric carry coefficients, not the full nonlinear memory of the model.

Q10. Does the relation loss recover the factor model characterised in the proposition?

Proposition 2 characterises the population optimum of the exact-projector objective, which is what the loss rewards. The released model optimises a ridge-regularised version of it jointly with the other losses, so the proposition does not guarantee that training reaches that optimum, a unique basis, or identifiable economic factors. Likewise, the attention rank statement applies to the product of per-head softmax weight matrices, not to the full nonlinear decoder.

Q11. Does exposure adjustment guarantee a factor-neutral portfolio or a higher Sharpe ratio?

No. Exact full-strength projection removes specified exposures from a score-proportional portfolio, as described in Appendix C. Ridge regularisation and partial adjustment do not preserve exact orthogonality. Selecting a long-only top-K portfolio from residual scores does not enforce zero portfolio exposure. The IC identity also concerns a particular normalised payoff; maximising mean IC is not equivalent to maximising Sharpe.

Q12. Why retain deep supervision and tiered selection when some ablations favour removing them?

Deep supervision provides a direct per-step gradient (Corollary 1); its measured predictive effect is nil on NASDAQ, and on NYSE a single seed without it reaches a higher Sharpe ratio at lower return with a paired interval that includes zero (Table 12), so it could be omitted in a simpler variant. The scoring tiers improve NYSE SR (7.0 versus 2.3 for the raw forecast) but reduce NASDAQ IRR, where the raw forecast earns more at similar SR. These are component trade-offs, not evidence of a uniformly optimal design; Appendix L reports the additional scoring comparisons.

Q13. Is the model tied to a fixed stock universe?

Yes. R_0 is a learned per-stock prior, and both states reset for each prediction window. Equivariance requires permuting that prior with the stock data. A constant prior could remove this particular dependence, but the paper does not evaluate transfer to an unseen universe. There is no recurrent state carried indefinitely between successive forecast windows.

Q14. Is the relation state a learned PCA, and why not use the window's own loadings as exposures?

The window's leading statistical loadings B_t are exposures in every configuration (Eq. 5), and the exposure arm of Table 3 removes only R_{T_c} while keeping them, so its return deficit (0.22 and 0.25, Table 12) is what the learned loadings add beyond the window's PCA. The two are computed from the same window but fitted to different things: B_t to the window's own returns, R_{T_c} to the day after it, and R_{T_c} explains 10.2% and 12.8% of that day against 11.5% and 13.6% for the PCA loadings (Section 6.2). The screens of Appendix K also tried replacing the statistical factors by the relation state; those candidates scored lower on validation and were not adopted.

A.3 Selection, evaluation, and reproducibility

Q15. Were candidate test metrics used for model selection?

The rule reads candidate validation metrics only, an earlier fold for the refit decision, and two validation years on S&P500. Training prints test metrics each epoch next to the validation ones, as the benchmarks' own code does; the rule does not read them. The published reference test values supply the fixed metric weights of J, so that a candidate is not chosen for a large gain in one metric at the cost of another; any fixed positive weights would serve. The complete search archive is not part of the release, so the development history can be audited from Appendix G but not re-executed.

Q16. Were the relation mechanisms added after the initial diagnosis, and how were they selected?

Yes. The initial probes showed the relation state carried but unused, which motivated the relation loss and the use of R_{T_c} as scoring exposures. The mechanisms were screened among 23 candidates on validation data with one seed each (Table 14), followed by three-seed final runs made once; no test number entered. On S&P500 the same rule retained the relation loss but set $\gamma_R = 0$, which we report rather than override. We report the diagnosis and the selected mechanism together so that the change in representation can be assessed separately from its predictive contribution. The selection and probe scripts are not in the release; Appendix K records the procedure and every candidate's score.

Q17. Does training-plus-validation refitting make the S&P500 comparison strictly matched?

It changes the training protocol. The earlier-fold decision uses no future targets, but refitting adds historical training targets relative to a training-only baseline. The same rule rejected refitting on NASDAQ and NYSE, where our models are consequently fitted on fewer days than the stock-selection benchmark's own entries, which are fitted on training plus validation days (Wang et al., 2025). References were not rerun with matched refitting, features, losses, and scoring, so the comparison remains at the pipeline level.

Q18. Why are TOPIX-100 and FTSE-100 absent, and which evaluator conventions are used?

The benchmark release used here does not supply the ticker lists and price files needed for direct replication on those markets; reconstructed universes would not be directly comparable. The reported metrics follow the benchmark conventions: Pearson IC, zero returns counted as hits, and the 15.87 annualisation factor. The Pearson evaluator zero-masks invalid entries rather than dropping them; the analytical identities specify their cross-section separately. The supplied run records also store Spearman rank IC.

Q19. What can a reviewer reproduce now, and what will be released?

The supplementary `RRST_submit` archive contains the model, loaders, training and checkpoint-evaluation scripts, configurations, benchmark files, the vendored StockMixer evaluator and the stock-selection metrics, with one as-given seed-0 checkpoint and JSON record per dataset variant; `evaluate.py` reproduces each checkpoint's stored metrics. The multi-seed run records and the search, ablation and probe scripts are not in the archive; the appendix records their procedures and results, and the full code repository will be released on GitHub after the review period.

B ARCHITECTURAL COMPARISON

Table 5: RRST against the Recurrent Looped Transformer (RLT-1, Zhang et al., 2026), the closest design.

	RLT-1	RRST
Task; unit of recurrence	next-token prediction on one sequence; one token	cross-sectional ranking of N stocks; one trading day of the last T_c days of a window, re-run per prediction day
Input of a step	one token embedding	a cross-section: N stocks $\times$ feature groups of a risk model refitted on that step's historical sub-window, rank-scored across stocks
Encoder	causal transformer over tokens; its key–value memory of the whole prefix is kept and read by the decoder	one cross-stock attention layer over the N stocks of the step; nothing is kept across steps
Recurrent state	one vector $s_t \in \mathbb{R}^d$ plus sliding-window KV caches C_t^D	a set of N per-stock states $S_t \in \mathbb{R}^{N\times d}$ and a relation state $R_t \in \mathbb{R}^{N\times r}$ (loadings on latent factors)
Feedback into the step	gated merge $u_t = e_t + \alpha\, g_t \odot W_s \mathrm{RMSNorm}(s_{t-1})$, $\alpha = 0.1$	the same gated merge per stock (Eq. 2, $\alpha = 1$), plus R_{t-1} entering its own gate
Decoder	L_D blocks of causal sliding-window self-attention over recent tokens, cross-attention to the encoder memory, FFN	two blocks of two-hop inducing-point attention over stocks, both hops biased by R_t, FFN; no attention over time
State update	overwrite: $s_t =$ decoder output	leaky per-channel: $S_t = \lambda \odot S_{t-1} + \eta \odot \Delta_t$, λ log-uniform over $[0.05, 0.95]$; $R_t = (1-\beta_t)R_{t-1} + \beta_t C_t$
Initial state	learned $s^\star$; empty caches	learned S_0 shared across stocks; learned per-stock prior R_0
Temporal channels	three: state, decoder KV window, encoder memory	one: the two states (no memory or cache)
Output	softmax over the vocabulary from s_t	forecast, two selection heads and an auxiliary output from S_{T_c}; a scoring layer neutralises the forecast to the risk model and to R_{T_c} and fills the top-K slots
Losses	teacher-forced cross-entropy at every token; full BPTT	daily IC of the exposure-adjusted forecast, selection losses, deep supervision on past target days, relation loss on R_{T_c}, auxiliary residual loss
Analysis	work/storage cost; invariance to the prompt–response split	exact decomposition and conditional bounds of the joint-state temporal Jacobian, finite-window update weights, attention-weight structure, the optimum of the relation objective, IC as a score-portfolio return

Experiment inventory. The study comprises 30 final runs (the six S&P500 runs trained again after selecting γ_R), 54 module ablations, six additional appendix runs, 28 relation-design and 20 S&P500 strength screens, approximately 300 earlier selection runs, and 18 walk-forward retrainings. Tables 11 and 12 re-score stored checkpoints and involve no training. This is the study inventory; the release contains five seed-0 checkpoints.

C THE SCORE PORTFOLIO AND EXACT FORECAST PROJECTION

With $\tilde{s}_t$ and D_t as in Section 5, for a specified cross-section with nonzero denominators:

Proposition 4 (The IC is a portfolio return)**.** *The return of the portfolio $\tilde{s}_t$ is $\rho_t^s = \tilde{s}_t^\top y_t = \mathrm{IC}_t D_t$. Hence maximising the mean IC maximises the dispersion-scaled return of the score-weighted long–short portfolio, and if D_t is constant across days ICIR is that portfolio's daily Sharpe ratio. The equal-weight top-K return is not a function of* IC_t.

The following statement concerns an exact, full-strength projection on a specified valid cross-section.

Proposition 5 (What projection removes)**.** *Let $P^\perp = I - P_{[\mathbf{1}, Z_t]}$, $e = P^\perp f$, and $y^\perp = P^\perp y$, with nonzero denominators below. Then (i) e has zero sample covariance with every column of Z_t; (ii) $\mathrm{IC}(e, y) = \mathrm{IC}(e, y^\perp)\|y^\perp\|/\|y - \bar{y}\mathbf{1}\|$; and (iii) the score-proportional payoff satisfies $\tilde{e}^\top y = \tilde{e}^\top y^\perp$.*

The exposure-spanned component contributes zero to this particular payoff. This identity does not guarantee lower variance than an unprojected strategy or higher ICIR or Sharpe: the weights change, and covariances also matter. The statement applies to an exact projection with relation columns included as additional exposures. Ridge regularisation, partial adjustment, capping, and tiered top-K selection do not preserve its exact orthogonality guarantees. In particular, selecting the largest residual scores does not make a long-only top-K portfolio factor-neutral.

D PROOFS

D.1 THE TEMPORAL JACOBIAN AND DIRECT SUPERVISION

Use x_k, $G_k = A_k + \Xi_k$ and $\bar{a} = \max_k \|A_k\|_2$ as defined in Section 5; since $\lambda_i \in (0, 1)$ and $\beta_k \in (0, 1)$ entrywise, $\bar{a} < 1$.

Proof of Theorem 1. The chain rule for the joint state gives $\partial x_T / \partial x_t = (A_T + \Xi_T) \cdots (A_{t+1} + \Xi_{t+1})$, a product of m factors in the order of the recurrence. Expanding it gives 2^m terms, each a product of m factors chosen from $\{A_k, \Xi_k\}$; the one term with no Ξ factor is $\prod_k A_k$, and $\mathcal{X}$ is the sum of the others. Each A_k is diagonal, so $\prod_k A_k$ is diagonal with entry $\prod_k \lambda_i = \lambda_i^m$ at every stock coordinate of channel i and $\prod_k (1 - \beta_{k,j})$ at every relation coordinate; this proves (i). For (ii), $\lambda = 0$ makes the stock block of every A_k zero, so the stock block of $\prod_k A_k$ is zero and the stock-to-stock block of $\partial x_T / \partial x_t$ equals that of $\mathcal{X}$, every term of which contains a Ξ factor. For (iii), $\|A_k\|_2 \le \bar{a}$ and $\|\Xi_k\|_2 \le \varepsilon$ give $\|G_k\|_2 \le \bar{a} + \varepsilon$ and, by submultiplicativity, the upper bound. A term of $\mathcal{X}$ with $b \ge 1$ factors Ξ has norm at most $\bar{a}^{\,m-b} \varepsilon^{\,b}$ whatever the positions of its factors, and summing over the $\binom{m}{b}$ positions and over b bounds $\|\mathcal{X}\|_2$ by $\sum_{b=1}^m \binom{m}{b} \bar{a}^{\,m-b} \varepsilon^{\,b} = (\bar{a} + \varepsilon)^m - \bar{a}^{\,m}$. Since $|[\mathcal{X}]_{\nu\nu}| \le \|\mathcal{X}\|_2$, the diagonal entry of the Jacobian is at least $\lambda_i^m - \|\mathcal{X}\|_2$, which is the lower bound. The proof carries the relation state along in x_k throughout, so it makes no assumption that the stock state alone is Markov. □

The lower bound in (iii) is informative when it is positive, and non-expansion needs $\bar{a} + \varepsilon \le 1$; neither is enforced in the released model, whose gain η is initialised at 0.5 and learned freely, so (iii) says what would follow from small learned feedback rather than certifying the trained network. The measured gradient ratios of Table 13 are the empirical counterpart of (i) and (ii).

Corollary 1 (Direct gradients from deep supervision)**.** *For $\mathcal{L} = \sum_j w_j \ell_j(x_j)$ and $G_{m \leftarrow j} = G_m \cdots G_{j+1}$, the gradient at x_j is $w_j \nabla_{x_j} \ell_j + \sum_{m > j} w_m G_{m \leftarrow j}^\top \nabla_{x_m} \ell_m$: the first term passes through no temporal Jacobian, whatever the cell and T. The later terms can partially cancel it, so it bounds the direct gradient of step j, not the total.*

Proof. Differentiate the sum: ℓ_j depends on x_j directly and each later ℓ_m through $x_m(x_j)$, whose Jacobian is $G_{m \leftarrow j}$. □

D.2 UPDATE WEIGHTS AND ATTENTION STRUCTURE

Proposition 6 (Finite-window weights of the leaky update)**.** *For each stock, write its state as $s_k \in \mathbb{R}^d$ and its decoder update as δ_k. Then $s_T = \Lambda^T s_0 + \sum_{k=1}^T \Lambda^{T-k} H \delta_k$, with $H = \mathrm{diag}(\eta)$.*

For channel i, the last update's share of the geometric coefficients is $\omega_i = (1-\lambda_i)/(1-\lambda_i^T)$, and their finite-window Kish effective sample size is $n_i(T) = \frac{1+\lambda_i}{1-\lambda_i}\frac{1-\lambda_i^T}{1+\lambda_i^T} \le T$.

For $T = 8$ and the specified initial spread, $n_i(T)$ ranges from approximately 1.1 to 7.89 and the mean last-update share is 0.70; shared $\lambda = 0.9$ gives 0.18. The infinite-horizon limit ranges from 1.1 to 39. These are coefficient summaries, not a complete memory spectrum: Δ_k itself depends on earlier states.

Proposition 7 (Attention weights and stock permutations)**.** *For each attention head, let $\mathcal{A}_h \in \mathbb{R}^{r\times N}$ and $\mathcal{B}_h \in \mathbb{R}^{N\times r}$ be the two softmax weight matrices. Their product $\mathcal{M}_h = \mathcal{B}_h\mathcal{A}_h$ is row-stochastic, has rank at most r, and is non-expansive in ℓ_∞ when applied to a fixed value vector. The two attention hops cost $O(Nrd)$. The model is permutation-equivariant when its per-stock prior R_0 is permuted with the inputs, subject to consistent numerical and tie-breaking conventions.*

$\mathcal{M}_h$ describes the composition of attention weights, not the full decoder operator. Inter-hop normalisation, value projections, multiple heads, residual connections, and input-dependent weights prevent a direct extension of this rank or non-expansion claim to the decoder or its Jacobian.

D.3 Proofs of the analytical statements

Proof of Proposition 1. Under the stated idealisation, a price ratio satisfies $\log(\theta_i P/(\theta_i P')) = \log(P/P')$, and a volume value relative to its window mean satisfies $\log(\vartheta_i V/(\vartheta_i \bar{V})) = \log(V/\bar{V})$. Recovered closes are linear in a stock's close and moving-average columns, so their ratios are unchanged when the recovery decisions are held fixed. Returns, volatility, return shares, factor loadings, residuals, and the derived characteristics are consequently unchanged. For separately normalised price columns, a log cross-column ratio gains a stock-specific constant; subtracting its within-window mean removes that constant. Volume and its moving averages share the same scale. All arguments are conditional on fixed masks and exclude absolute price floors or acceptance cutoffs. The level features themselves can change order under stock-specific rescaling. Finally, strictly increasing transformations of any constructed feature preserve its ranks and ties, including those of the level features. □

Proof of Proposition 6. For one stock, write its state and update as vectors $s_k, \delta_k \in \mathbb{R}^d$. Repeated substitution into $s_k = \Lambda s_{k-1} + H\delta_k$ gives the expansion in the proposition. Channel i has coefficients $1, \lambda_i, \ldots, \lambda_i^{T-1}$. Their sum is $(1-\lambda_i^T)/(1-\lambda_i)$, giving the last-update share ω_i. The finite-window Kish effective sample size is

$$n_i(T) = \frac{\left(\sum_{m=0}^{T-1}\lambda_i^m\right)^2}{\sum_{m=0}^{T-1}\lambda_i^{2m}} = \frac{1+\lambda_i}{1-\lambda_i}\frac{1-\lambda_i^T}{1+\lambda_i^T} \le T,$$

where the inequality follows from Cauchy–Schwarz. With $\lambda_i = 0.05(0.95/0.05)^{(i-1)/63}$ for $i = 1, \ldots, 64$ and $T = 8$, the range is approximately $[1.11, 7.89]$ and $\bar{\omega} = 0.698$. For shared $\lambda = 0.9$, $\omega = 0.1/(1-0.9^8) = 0.176$. Only in the limit $T \to \infty$ does the effective sample size become $(1+\lambda_i)/(1-\lambda_i)$, whose range over the initial retentions is $[1.11, 39.0]$. The expansion is an identity in the realised updates, not a linear time-invariant response to the inputs, because each δ_k depends on previous stock and relation states. □

Proof of Proposition 7. Each row of $\mathcal{A}_h$ is a softmax over eligible stocks, and each row of $\mathcal{B}_h$ is a softmax over factors. Thus both matrices are nonnegative and row-stochastic, assuming at least one eligible stock. Their product obeys $\mathcal{M}_h\mathbf{1} = \mathcal{B}_h\mathcal{A}_h\mathbf{1} = \mathbf{1}$ and $\operatorname{rank}(\mathcal{M}_h) \le r$. For fixed $\mathcal{M}_h$ and a value vector v, $|(\mathcal{M}_h v)_i| \le \sum_j (\mathcal{M}_h)_{ij}|v_j| \le \|v\|_\infty$. The attention contractions require $O(Nrd)$ operations across heads; feature projections add their usual width-dependent cost. The implementation's inter-hop RMSNorm and value projections mean that the full value map is not simply $\mathcal{M}_h v$, and this calculation does not bound its Jacobian.

For a stock permutation Π, the temporal Gram matrix is unchanged because it sums over stocks. The resulting per-stock loadings, residuals, features, and rank scores permute consistently, provided the same eigenspace convention is used. Stockwise layers and the two attention hops commute

with the permutation when R_0 is also permuted, as do the exposure regressions. The scoring selection requires consistent tie-breaking; with distinct selection scores, the top-K sets and final scores permute accordingly. Floating-point arithmetic and index-based resolution of exact ties can break exact implementation-level equality. □

Proof of Proposition 2. For the exact orthogonal projector P_R, symmetry and idempotence give $\mathcal{L}_{\text{rel}} = \|(I - P_R)\hat{y}_t\|^2 = 1 - \hat{y}_t^\top P_R \hat{y}_t$. Conditioning on $\mathcal{F}_t$ makes P_R fixed, so the expected quadratic form is $\text{tr}(P_R \Sigma_t)$. Ky Fan's variational principle bounds this trace by $\sum_{k \le r} \mu_k$, attained by a leading eigenspace. If eigenvalues tie at the cutoff, multiple subspaces attain the same value. For the final statement, $R^\top u = 0$ gives $u^\top (Ra_t + \epsilon_t) = u^\top \epsilon_t$. The implemented ridge map $R(R^\top R + 10^{-2} I)^{-1} R^\top$ is generally not idempotent; its squared-residual loss is therefore not exactly the projector objective used in this proof. Nor does the proof imply that joint optimisation with the other losses reaches the conditional population optimum. □

Proof of Proposition 4. Since $\tilde{s}_t^\top \mathbf{1} = 0$,

$$\tilde{s}_t^\top y_t = \frac{(s_t - \bar{s}_t \mathbf{1})^\top (y_t - \bar{y}_t \mathbf{1})}{\|s_t - \bar{s}_t \mathbf{1}\|} = \text{IC}_t D_t.$$

If D_t is a positive constant, it cancels from the payoff's mean-to-standard-deviation ratio. For a counterexample to determining top-K return from IC, take $y = (1, 0, -1)$, $s^{(1)} = (3, -4, 1)$, and $s^{(2)} = (-1, 4, -3)$. Both score vectors have Pearson correlation $1/\sqrt{13}$ with y, but their top-1 returns are 1 and 0. Repeating each coordinate K times gives the same counterexample for top-K selection. □

Proof of Proposition 5. Let $G = [\mathbf{1}, Z_t]$ and let P_G be the orthogonal projector onto its column span. Then $G^\top (I - P_G) = 0$, so $\mathbf{1}^\top e = 0$ and $Z_t^\top e = 0$, proving (i). Since $y - y^\perp$ lies in that span, $e^\top y = e^\top y^\perp$; both e and $y^\perp$ have zero mean. Dividing by the relevant norms gives $\text{IC}(e, y) = \text{IC}(e, y^\perp)\|y^\perp\|/\|y - \bar{y}\mathbf{1}\|$, proving (ii). Normalising e in the same identity proves (iii). This is a statement about a particular projected score payoff, not a comparison of the variances of two different strategies. □

Lemma 1 (Order of the composite score)**.** *Assume at least K eligible stocks and $\delta > 0$. The K selected stocks occupy the top K positions, in slot order. Unselected stocks retain their capped-forecast values; capping can introduce ties.*

Proof. Each assigned value $c + \delta(K - s + 1)/K$ exceeds c and decreases strictly with slot index s. All unselected values are at most c. □

Corollary 2 (Normalisation and non-anticipation)**.** *Full-series normalisation need not be measurable from observations through day t, whereas $\ell_{t,i}^{\text{past}}$ is measurable from those observations. An idealised ratio-based map satisfying Proposition 1 is invariant to fixed per-stock normalisers; this does not certify the released implementation with its absolute cutoffs.*

Proof of Proposition 3 and Corollary 2. For positive prices, the full-series maximum satisfies

$$M_i = \max\left(\max_{s \le t} P_{s,i}, \max_{s > t} P_{s,i}\right) \ge \max_{s \le t} P_{s,i}.$$

Dividing $P_{t,i}$ by these quantities yields the stated inequality, with equality exactly when no future price exceeds the historical maximum. In the strict case, cancelling $P_{t,i}$ gives the log-ratio identity. Two series can agree through day t and have different future maxima, producing different full-series levels at t but identical past-only levels. This establishes the possible failure of non-anticipation. The final idealised invariance follows by setting $\theta_i = 1/M_i$ in Proposition 1, under its assumptions. A particular score may cancel or ignore these normalisers, so use of the benchmark inputs alone does not prove that every predictor exploits their future information. □

E NOTATION, FEATURE GROUPS AND THE SCORING LAYER

Table 6: Notation used in Sections 3–5.

Symbol	Meaning
N, F, W	number of stocks, columns per stock and day, days in the input window
$X_{t,i}$, $P_{t,i}$, $y_{t,i}$	observations, close and return $P_{t,i}/P_{t-1,i} - 1$ of stock i on day t
T_c, L, d, r	core steps per window, lookback of one step, state width, relation width; the window spans $L + T_c - 1$ days
Φ, $z^{(j)}$, E_t	feature map of one sub-window, its rank-scored features for step j, their encoded form entering cell step t
ρ, v_i, ψ, B, ϱ	log returns, volatility, statistical factors, loadings and residual returns of the sub-window risk model
S_t, R_t, S_0, R_0	stock state ($N \times d$), relation state ($N \times r$), their learned initial values
$\bar{S}$, U_t, β_t, C_t	normalised previous stock state, merged representation, relation gate, relation candidate $\tanh(\mathrm{MLP}[U_t; \bar{S}])$
Δ_t, λ, η	decoder update, per-channel retention in $(0, 1)^d$, per-channel gain
$\mathcal{A}_{kj}$, $\mathcal{B}_{ik}$, $\mathcal{M}_h$, λ_{in}, λ_{out}	attention weights of hop 1 (stocks to factors) and hop 2 (factors to stocks), their per-head product, relation-bias strengths
f, s_1, s_2	forecast head and the two selection heads read from S_{T_c}
Z_t, B_t, γ, γ_R	volatility and momentum exposures, statistical loadings, adjustment strengths for the loadings and for R_{T_c}
$\mathcal{N}_G(f)$, e_t	residual of f regressed on $[\mathbf{1}, G]$, exposure-adjusted forecast (Eq. 5)
K, c, δ, p	top-K slots, cap of the standardised forecast, slot increment, volatility power of the first selection target
$\tilde{y}_t$, $\hat{y}_t$, P_R, $\mathcal{L}_{\text{rel}}$	standardised next-day cross-section, its direction, projector onto $\mathrm{span}(R)$, relation loss
IC_t, D_t, $\tilde{s}_t$	daily Pearson correlation, return dispersion $\|y_t - \bar{y}_t\mathbf{1}\|$, unit-norm centred score weights
x_k, G_k, A_k, Ξ_k	joint state, its step Jacobian, the carry part and the learned part (Theorem 1)
$\mathcal{D}$, θ_i, ϑ_i	rescaling of a stock's price columns and volume group (Proposition 1)
ω_i, $n_i(T)$	last-step share and finite-window effective sample size of the carry coefficients (Proposition 6)
J	selection criterion $\min_q(\text{validation } q)/(\text{reference } q)$
$\ell_{t,i}$, M_i	benchmark level $P_{t,i}/M_i$ and the full-series price maximum (Proposition 3)

Table 7: The feature map Φ of one core step, per stock. Groups (f)–(h) are optional; `--no_levels` removes (c). The other groups are ratio-based, subject to the numerical qualifications in Proposition 1.

Group	Features	Count
(a) window-relative values	price columns relative to the sub-window's last close; other columns relative to their own mean over it; logs, clipped	F
(b) last moves	the last return and the last residual return	2
(c) levels	the file's own values on that day	F
(d) volatility, return shares	v_i; share of history days with return ≥ 0 and with return $= 0$	3
(f) reversal strength	lag-1 autocorrelation κ_i of the residual returns; the AR(1) forecast $\kappa_i \varrho_{L,i}$	2
(g) characteristics	$\rho_{L,i}/v_i$, $\lvert\rho_{L,i}/v_i\rvert$ (v_i the mean absolute return), the same for the residual, market beta, residual variance share	6
(h) open/high/low/volume	deviations from the window mean of $\log H/L$, $\log C/H$, $\log C/L$, $\log C/O$, the overnight gap; $\log V/\mathrm{MA}_5$, $\log V/\mathrm{MA}_{20}$	7

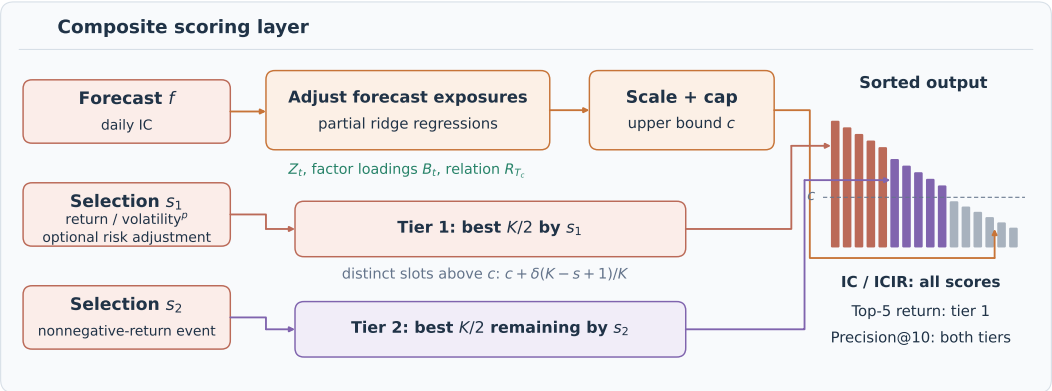

Figure 3: The scoring layer (Section 4.3): the forecast is adjusted for exposures, standardised and capped; the top-K slots are filled tier by tier from the two selection heads; IC and ICIR use the entire composite score, while top-of-ranking metrics depend on the selected tiers.

F Data and splits

Table 8: Splits, the benchmarks' own. The first $L + T_c - 1 = 71$ days of a file are inputs only.

Dataset (stocks)	Training targets	Validation	Test
NASDAQ (1026), both benchmarks	2013-04-16 to 2015-12-31	2016 (252 days)	2017-01-03 to 2017-12-08 (237 days)
NYSE (1737), both benchmarks	2013-04-16 to 2015-12-31	2016 (252 days)	2017 (236–237 days)
S&P500 (474), StockMixer	about 935 days from 2016	253 days (2020)	352 days (2021-01 to 2022-05)

NASDAQ and NYSE of the two benchmarks are the same prices (they agree to 5×10^{-7} after normalisation) with identical masks; the stock-selection benchmark adds open, high, low, volume and moving-average columns, and its loader drops the last NYSE test day. We use each benchmark's own files for its comparison, so the StockMixer rows read only the five close columns.

G Hyperparameter selection procedure

Scope of the record. This section describes the staged selection procedure and summarises its reported candidate scores. The candidates are the target of s_1 ($p \in \{1, 0.5\}$) and the ListNet term, then the risk model of the window (1, 5 or 10 statistical factors; root-mean-square or mean-absolute volatility), and on S&P500 also the optional input/loss groups, the lookback, the training length and γ_R. It is not a complete executable search log; the submitted release contains the selected recipes and five checkpoint records, as described in the reproducibility statement.

Configurations. $T_c = 8$, $L = 64$, $d = 64$, $r = 16$, 4 heads, one encoder and two decoder layers, about 2.2×10^5 parameters; a run takes 3–12 minutes and 3–9 GB on one RTX 3090. The selected hyperparameters (Section 4.5; command lines below) are: NASDAQ and NYSE, all 25 columns, $p = 0.5$, no ListNet, 10 or 5 statistical factors at $\gamma = 0.5$, no refit, the relation loss at weight 1 with $\gamma_R = 0.5$ and the neutral selection head; S&P500, $p = 0.5$, one fully removed factor, refit on training plus validation, the relation loss at weight 1 with $\gamma_R = 0$. The *no levels* rows use the same configuration without the level group (on S&P500 the rule selected ListNet there). The head, ListNet and risk-model hyperparameters below were selected on the configuration without the relation-state mechanisms (the diagnosis model of Section 6.2) and carried over unchanged; the relation-state mechanisms and γ_R were selected afterwards by the same rule (Appendix K).

Base recipe for every dataset: $T_c = 8$, $L = 64$, 5 factors removed from the input returns, cap $c = 2$, deep-supervision weight 0.3, recovered closes, groups (f) and (g), momentum neutralisation, normal scores, soft top-K terms (0.1, 0.2), auxiliary output (1.0). The validation record follows in Section G.1.

- **NASDAQ** (stock-selection files, J on IRR/SR, 3 seeds): $p = 0.5$ without ListNet, then 10 factors ($\gamma = 0.5$), RMS volatility; refit rejected on the fold-1 check.
- **NYSE**: $p = 0.5$ without ListNet, then 5 factors ($\gamma = 0.5$) with mean-absolute volatility; refit rejected.

Command-line additions to the base recipe. The following switches supplement the base settings above. For StockMixer-file NASDAQ and NYSE, omit `--features all`. The repository README gives the complete training entry points.

NASDAQ

```
--features all --select_vol_power 0.5 --factor_k 10
--factor_gamma 0.5 --rel_loss_weight 1.0 --rel_gamma 0.5 --rel_select
```

NYSE

```
--features all --select_vol_power 0.5 --factor_k 5
--factor_gamma 0.5 --vol_stat mad --rel_loss_weight 1.0
--rel_gamma 0.5 --rel_select
```

S&P500, as given

```
--select_vol_power 0.5 --factor_k 1 --factor_gamma 1
--vol_stat mad --refit_epoch 12 --rel_loss_weight 1.0 --rel_gamma 0
```

S&P500, no levels

```
--select_vol_power 0.5 --listnet_weight 0.5 --factor_k 1
--factor_gamma 1 --refit_epoch 18 --rel_loss_weight 1.0
--rel_gamma 0 --no_levels
```

G.1 VALIDATION SEARCH RECORD

The S&P500 search uses two validation years and two seeds per candidate: mean J for the as-given setting, and the worse of the two years for no levels. Codes P/F denote plain/full feature groups, s/r denote $p = 1/0.5$, and n/l denote ListNet off/on. Bold marks the retained choice. On the no-level runs, stages 3–6 kept their defaults. These are the reported candidate summaries, not a complete executable search archive.

Table 9: S&P500 validation candidates and scores J, grouped by selection stage.

Stage	As given	No levels
1	Psn 0.956, Psl 0.929, Prn 0.956, Prl 0.930, Fsn 0.912, Fsl 0.899, **Frn 0.979**, Frl 0.931	Psn 0.307, Psl 0.292, Prn 0.292, Prl 0.310, Fsn 0.315, Fsl 0.317, Frn 0.288, **Frl 0.382**
2	k5 0.930, k1 0.942, k10 0.946, k5mad 0.928, **k1mad 0.971**	k5 0.322, **k1 0.350**, k10 0.322, k5mad 0.242, k1mad 0.280
3	**w64 0.971**, w32 0.936, w128 0.910, e12 0.936	**w64 0.350**, w32 0.306, w128 0.155, e12 0.222
4	**t0 0.971**, e40 0.952, lr3 0.926, wd1 0.947, keep1 0.940	**t0 0.350**, e40 0.325, lr3 0.224, wd1 0.228, keep1 0.277
5	**i0 0.971**, r0 0.853, r10 0.946, d32 0.942, g1 0.971	**i0 0.350**, r0 0.248, r10 0.285, d32 0.242, g1 0.350
6	**all 0.971**, last2 0.903, last2w32 0.797	**all 0.350**, last2 0.228, last2w32 0.052

Earlier-fold refit check. For the as-given setting, train-only scores 0.983 and refit scores 1.009; refitting is adopted at epoch 12. For no levels, the reported comparison is refit 0.640 versus train-only 0.598; refitting is adopted at epoch 18. The refit procedure is specified in Section 4.5.

H RESULTS OVERVIEW

Table 10: Seed variability. Panel (a): standard errors across the three seeds of Tables 1 and 2; panel (b): the same for the repeated arms of Table 3; panel (c): the per-seed test values behind the main means. These quantify seed variability, not uncertainty across market periods or significance relative to the published baselines. Single-seed arms have no reported standard error.

(a) Main comparisons

Dataset	Metric order	As given	No levels
NASDAQ	IRR / SR	0.09 / 0.34	0.09 / 0.22
NYSE	IRR / SR	0.12 / 1.35	0.14 / 0.38
NASDAQ	IC / ICIR / prec@10 / SR	0.0004 / 0.016 / 0.001 / 0.29	0.0006 / 0.014 / 0.003 / 0.55
NYSE	IC / ICIR / prec@10 / SR	0.0005 / 0.025 / 0.005 / 1.02	0.0006 / 0.010 / 0.005 / 1.43
S&P500	IC / ICIR / prec@10 / SR	0.0007 / 0.004 / 0.001 / 0.06	0.0009 / 0.006 / 0.003 / 0.04

(b) Module ablations; metric order: IRR / SR / IC

Variant	NASDAQ	NYSE
RRST (full)	0.09 / 0.34 / 0.000	0.12 / 1.35 / 0.001
overwriting state ($\lambda = 0$)	0.08 / 0.41 / 0.001	0.13 / 0.44 / 0.001
single step ($T_c = 1$)	0.12 / 0.49 / 0.000	0.13 / 1.73 / 0.000
whole-window features (non-causal)	0.05 / 0.38 / 0.001	0.14 / 0.40 / 0.000
no relation loss	0.12 / 0.34 / 0.001	0.23 / 1.88 / 0.000
not used by the scoring layer	0.09 / 0.43 / 0.000	0.06 / 1.22 / 0.001
no cross-stock attention	0.04 / 0.26 / 0.001	0.14 / 0.29 / 0.000

(c) Per-seed test values (seeds 0 / 1 / 2) behind the means of Tables 1 and 2

		stock-selection files			StockMixer files	
Dataset	Setting	IRR	SR	IC	IC	SR
NASDAQ	as given	0.82 / 0.80 / 1.07	4.19 / 4.08 / 5.16	0.0654 / 0.0659 / 0.0664	0.0599 / 0.0592 / 0.0605	2.18 / 3.20 / 2.74
NASDAQ	no levels	1.02 / 0.87 / 1.17	4.61 / 4.29 / 5.05	0.0329 / 0.0320 / 0.0311	0.0214 / 0.0203 / 0.0222	3.43 / 1.95 / 3.74
NYSE	as given	0.68 / 1.04 / 1.02	9.71 / 5.77 / 5.58	0.0637 / 0.0651 / 0.0653	0.0606 / 0.0621 / 0.0613	5.01 / 8.09 / 5.03
NYSE	no levels	1.42 / 1.32 / 0.97	5.15 / 6.39 / 5.40	0.0391 / 0.0388 / 0.0373	0.0351 / 0.0350 / 0.0367	8.42 / 3.94 / 8.01
S&P500	as given	–	–	–	0.0462 / 0.0472 / 0.0485	1.36 / 1.55 / 1.49
S&P500	no levels	–	–	–	0.0048 / 0.0079 / 0.0067	-0.18 / -0.06 / -0.09

Table 11: Block-bootstrap intervals over the test days, computed from the final checkpoints (seed 0 is released; seeds 1–2 are the other two runs of Tables 1 and 2). The daily IC, top-10 hit and top-5 return series of the three seeds are resampled jointly in circular blocks of 10 trading days (2000 resamples) and the seed-mean statistic is recomputed; cells give the point estimate and the 2.5–97.5% percentile interval. Bold: the whole interval lies above the published value. The intervals describe the sampling variability of one test period, not other market regimes.

(a) Stock-selection files: top-5 IRR and SR (published best 0.82 / 2.32 on NASDAQ, 0.79 / 2.35 on NYSE)

Dataset	Setting	IRR	SR
NASDAQ	as given	0.90 [0.50, 1.44]	**4.48** [2.82, 6.24]
NASDAQ	no levels	1.02 [0.61, 1.53]	**4.65** [3.15, 6.12]
NYSE	as given	0.92 [0.67, 1.34]	**7.02** [5.76, 12.70]
NYSE	no levels	**1.24** [0.85, 2.05]	**5.65** [4.40, 12.71]

(b) StockMixer files: IC, ICIR, prec@10 and top-5 SR (published values in Table 2)

Dataset	Setting	IC	ICIR	prec@10	SR
NASDAQ	as given	**0.060** [0.052, 0.068]	**1.11** [0.96, 1.28]	**0.613** [0.589, 0.636]	2.71 [1.00, 5.18]
NASDAQ	no levels	0.021 [0.015, 0.027]	0.37 [0.27, 0.48]	**0.610** [0.587, 0.633]	**3.04** [1.64, 4.75]
NYSE	as given	**0.061** [0.057, 0.067]	**1.33** [1.22, 1.48]	**0.746** [0.731, 0.762]	**6.05** [4.80, 10.69]
NYSE	no levels	**0.036** [0.030, 0.042]	**0.74** [0.61, 0.90]	**0.740** [0.722, 0.759]	**6.79** [5.40, 9.41]
S&P500	as given	0.047 [0.038, 0.058]	**0.49** [0.38, 0.61]	0.547 [0.519, 0.576]	1.47 [−0.03, 3.13]
S&P500	no levels	0.007 [−0.004, 0.016]	0.06 [−0.03, 0.16]	0.528 [0.503, 0.553]	−0.11 [−1.53, 1.40]

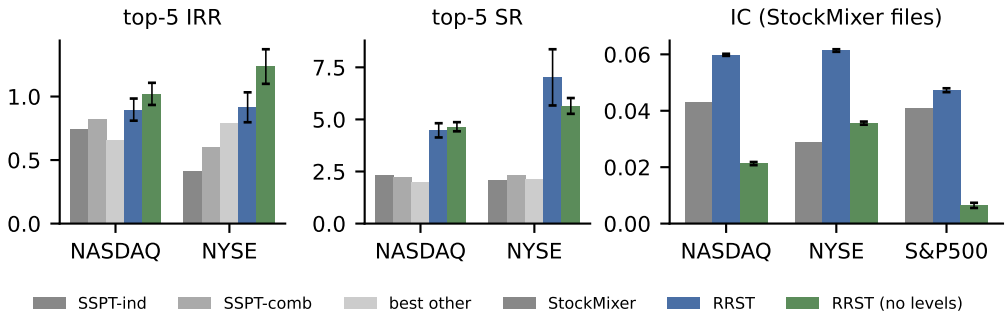

Figure 4: Both benchmarks at a glance. Left, centre: top-5 IRR and SR on the stock-selection benchmark against the three strongest reported values; right: IC on the StockMixer files. Error bars: s.e. over 3 seeds.

Table 12: Paired block-bootstrap differences over the test days, full model minus ablation arm (positive: the arm costs), for top-5 IRR, SR and IC, computed from the checkpoints of Table 3 on the same resampled days for both models (circular 10-day blocks, 2000 resamples). Arms marked † are paired seed by seed over three seeds; single-seed arms are compared with the three-seed mean of the full model on the same days. Cells give the point difference and the 2.5–97.5% percentile interval; bold: the interval excludes zero.

(a) NASDAQ, stock-selection files

Variant	IRR	SR	IC
cell: overwriting state ($\lambda = 0$)†	$+0.21\,[-0.07, +0.52]$	$+0.95\,[-0.31, +1.97]$	$0.000\,[-0.001, +0.001]$
cell: shared retention ($\lambda \equiv 0.9$)	$-0.29\,[-0.85, +0.15]$	$-0.23\,[-1.75, +1.29]$	$-0.001\,[-0.002, +0.001]$
sub-windows: single step ($T_c = 1$)†	$+0.09\,[-0.23, +0.43]$	$+0.30\,[-1.06, +1.45]$	$\mathbf{+0.002}\,[+0.001, +0.003]$
sub-windows: whole-window features†	$+0.23\,[-0.02, +0.52]$	$+0.83\,[-0.23, +1.75]$	$0.000\,[-0.001, +0.001]$
relation state frozen ($R_t \equiv R_0$)	$+0.03\,[-0.45, +0.48]$	$+0.90\,[-0.60, +2.08]$	$\mathbf{-0.002}\,[-0.003, 0.000]$
relation state: no relation loss†	$+0.05\,[-0.27, +0.33]$	$+0.14\,[-0.90, +0.98]$	$0.000\,[-0.001, +0.001]$
relation state: not used by the scoring layer†	$+0.22\,[-0.08, +0.56]$	$+1.05\,[-0.59, +2.27]$	$\mathbf{-0.002}\,[-0.003, 0.000]$
attention: no relation bias	$-0.04\,[-0.49, +0.33]$	$+0.18\,[-1.44, +1.54]$	$0.000\,[-0.001, +0.002]$
attention: no cross-stock attention†	$+0.25\,[-0.13, +0.61]$	$\mathbf{+1.44}\,[+0.14, +2.68]$	$\mathbf{+0.003}\,[+0.001, +0.005]$
training: no deep supervision	$+0.01\,[-0.38, +0.36]$	$+0.02\,[-1.62, +1.47]$	$-0.001\,[-0.002, +0.001]$

(b) NYSE, stock-selection files

Variant	IRR	SR	IC
cell: overwriting state ($\lambda = 0$)†	$\mathbf{-0.22}\,[-0.70, -0.01]$	$+1.71\,[-0.77, +2.89]$	$+0.001\,[0.000, +0.002]$
cell: shared retention ($\lambda \equiv 0.9$)	$-0.01\,[-0.24, +0.36]$	$-3.21\,[-4.83, +1.85]$	$\mathbf{+0.003}\,[0.000, +0.005]$
sub-windows: single step ($T_c = 1$)†	$+0.05\,[-0.07, +0.17]$	$-0.17\,[-0.90, +1.04]$	$+0.001\,[0.000, +0.003]$
sub-windows: whole-window features†	$-0.08\,[-0.51, +0.14]$	$+2.18\,[-0.05, +3.24]$	$\mathbf{+0.001}\,[0.000, +0.003]$
relation state frozen ($R_t \equiv R_0$)	$+0.25\,[-0.01, +0.71]$	$-3.39\,[-5.12, +2.04]$	$+0.001\,[-0.001, +0.003]$
relation state: no relation loss†	$-0.14\,[-0.78, +0.11]$	$+0.47\,[-0.83, +4.76]$	$0.000\,[-0.002, +0.001]$
relation state: not used by the scoring layer†	$\mathbf{+0.25}\,[+0.01, +0.69]$	$-2.72\,[-4.12, +2.56]$	$+0.001\,[0.000, +0.003]$
attention: no relation bias	$+0.05\,[-0.19, +0.22]$	$+2.02\,[-0.61, +2.83]$	$0.000\,[-0.001, +0.001]$
attention: no cross-stock attention†	$\mathbf{-0.23}\,[-0.73, 0.00]$	$+1.70\,[-0.71, +2.56]$	$0.000\,[-0.001, +0.001]$
training: no deep supervision	$+0.20\,[-0.05, +0.65]$	$-4.67\,[-6.56, +0.41]$	$\mathbf{+0.002}\,[+0.001, +0.003]$

I MEASURING THE LEVEL LEAK

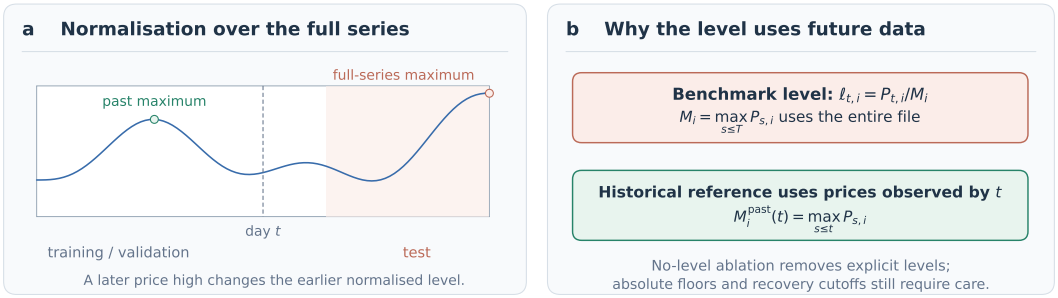

Figure 5: Full-series and historical normalisation. Fixed normalisers cancel in idealised ratios; the implemented no-level ablation retains the numerical qualifications of Section 7.

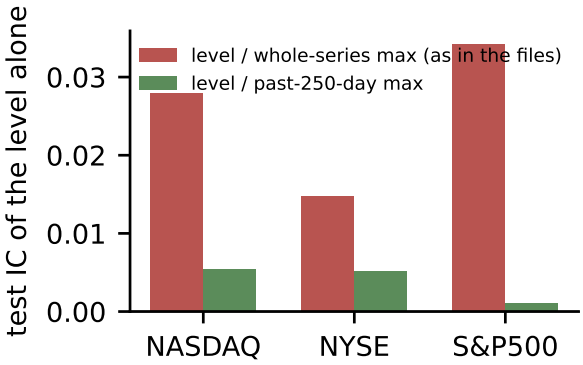

Figure 6: Test IC of the single feature $-\ell_{t-1,i}$ (level over the whole-series maximum, as in the files) and of a causal rolling reference (over the maximum of the previous 250 days). The rolling reference differs from the expanding historical maximum in Proposition 3.

On S&P500 by period (training / validation / test) the leak feature has IC 0.041 / 0.034 / 0.034; among past-only one-feature signals the largest test-period IC is the 1-day reversal at 0.016.

J GRADIENT PROBE OF THE RECURRENCE

Table 13: Gradient probe on the selected models (seed 0, validation days): share of the forecast's input gradient by core step (oldest to newest). The last column is $\|\partial f/\partial S_1\|/\|\partial f/\partial S_8\|$.

Dataset	input gradient share by step								ratio
NASDAQ	.014	.016	.022	.031	.050	.084	.166	.618	0.062
NYSE	.024	.025	.029	.037	.052	.065	.135	.634	0.055
S&P500	.099	.044	.047	.055	.066	.085	.119	.485	0.117
Ablated cells, NASDAQ (state-path norms $\|\partial f/\partial S_t\|$ by step, then the ratio)									
overwriting cell	6.8e-6	8.7e-6	1.7e-5	3.4e-5	7.7e-5	2.1e-4	6.8e-4	2.5e-3	0.0028
shared $\lambda \equiv 0.9$	6.3e-4	6.5e-4	7.0e-4	7.7e-4	8.4e-4	9.1e-4	1.0e-3	1.4e-3	0.47

K RELATION-STATE PROBES

The probes of Section 6.2 are computed on 40 validation days spread over the validation year, with the model in inference mode and a trace of R_t and β_t at every core step.

Industry alignment. For every stock with an industry in the benchmark's own lists (113 groups on NASDAQ, 130 on NYSE; unassigned stocks excluded), the share of its ten nearest neighbours by cosine similarity in R-space that are in the same industry, averaged over stocks; the chance level is the share of same-industry stocks among all others, and the same statistic is computed for the prior R_0, for a Gaussian random R and for the return correlation over the last sub-window.

Explained variance. With $\tilde{y}$ the next day's centred, volatility-standardised return vector over the valid stocks, the statistic $1 - \|(I - P_R)\tilde{y}\|^2/\|\tilde{y}\|^2$ for $R = R_{T_c}$, R_0, a random R, and the r leading eigenvectors of the sub-window's return correlation matrix (a backward-looking factor model of the same rank).

Dynamics. The mean gate β_t and the relative change $\|R_t - R_{t-1}\|/\|R_t\|$ by step, and the mean cosine between a stock's loadings on consecutive days.

Interventions. The validation metrics of the trained model, of the same model with $R_t \equiv R_0$, with $\lambda_{\text{in}} = \lambda_{\text{out}} = 0$, and, where the scoring layer neutralises against R_{T_c}, with that neutralisation switched off; no parameter is retrained during these interventions. The study used a dedicated probe script and diagnosis models trained with $w_{\text{rel}} = 0$ and $\gamma_R = 0$ for three seeds. These scripts, diagnosis checkpoints, and individual probe records are not included in `RRST_submit`; the procedures and aggregate measurements are reported here.

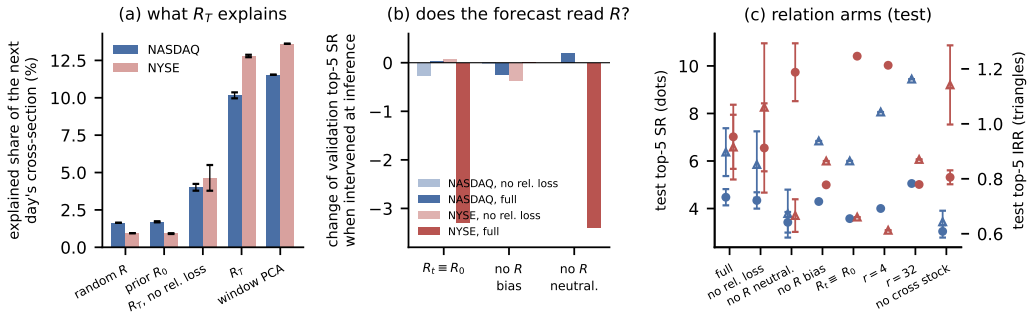

Figure 7: Relation-state probes on validation days, final models (3 seeds) against the diagnosis models trained without the relation loss. (a) Share of the next day's standardised cross-section explained by the projection onto $\text{span}(R_{T_c})$, against a random R, the prior R_0 and the window's own PCA loadings. (b) Change of the validation top-5 Sharpe ratio when, at inference, R_t is frozen at R_0, the attention bias is removed, or the neutralisation against R_{T_c} is switched off. (c) The relation arms of the ablation (test, one seed; the full model three seeds).

K.1 SELECTION OF THE RELATION-STATE MECHANISMS

The initial probes in Section 6.2 used the model trained by ranking losses alone. We then screened, with one seed per candidate and validation data only, the following relation mechanisms: the weight w_{rel} of the relation loss, the strength γ_R at which the forecast is made neutral to R_{T_c}, the same neutralisation of the first selection head (S), the relation loss applied to every step's R_j on its past target day (D), an attention whose two hops read the relation state alone with no content logits (R), a larger initial bias strength, the relation state replacing the window's statistical factors, and a relation module that reads the stock features without training them (detached).

Validation rule. Table 14 scores every candidate by the rule of Section 4.5 with the seed-0 diagnosis model (the same configuration without these mechanisms) as the reference ($J = \min_q$ validation q / reference q; the reference's own seeds score 0.90–1.00 on NASDAQ and 0.89–1.00 on NYSE). The candidates within noise of the reference on NASDAQ were re-run on NYSE, and the one with the highest mean J was adopted; no candidate test metric entered the reported selection rule.

S&P500. With hyperparameters chosen using two validation years, the strength of the mechanism was then screened the same way (train-only runs, two seeds, two validation years, StockMixer references): $\gamma_R = 0.5$ with the neutral selection head scores $J = 0.924$, $\gamma_R = 0.3$ 0.881, $\gamma_R = 1$ 0.877, $\gamma_R = 0.5$ without the selection head 0.866, and $\gamma_R = 0$ (relation loss kept, R_{T_c} not used by the scoring layer) 0.933; the $\gamma_R = 0$ and $\gamma_R = 0.5$ configurations with neutral selection tie within 0.01, and the tie goes to the higher validation IC, so S&P500 runs with $\gamma_R = 0$.

Table 14: Relation-state mechanisms screened on validation data (seed 0). w_{rel}: relation-loss weight; γ_R: neutralisation strength; S: first selection head neutralised too; D: relation loss on every step; R: attention through the relation state alone. Bold: adopted.

w_{rel}	γ_R	S	D	R	other	J NASDAQ	J NYSE
1	**0.5**	✓				**0.981**	**0.908**
1	0.5	✓	✓		detached	0.967	–
1	0.5				detached	0.964	0.807
1	0.5	✓	✓	✓	λ init 3	0.962	0.860
1	0.5	✓			detached	0.957	0.892
1	0			✓	λ init 3	0.955	0.782
1	0.3				λ init 3	0.944	–
1	0.5	✓	✓			0.932	–
1	0.5		✓			0.931	–
0.3	0					0.909	–
1	0.5	✓			no statistical factors	0.907	–
0.3	0.3				λ init 3	0.900	–
1	0.5	✓		✓	λ init 3	0.896	–
1	0					0.895	–
0	0.3					0.889	–
3	0.5					0.878	–
0.3	0.3					0.872	–
1	0.5					0.859	–
1	0.5			✓	λ init 3	0.854	–
1	0.5				no statistical factors	0.824	–
1	1		✓		no statistical factors	0.809	–
1	1				no statistical factors	0.801	–
1	0.3				λ init 3; detached	0.742	–

L ADDITIONAL ABLATIONS: INPUT COLUMNS AND THE SCORING LAYER

Table 15: Arms outside the model's modules (seed 0; the full model seeds 0–2): the scoring layer without the risk neutralisation, without the tiered top-K selection (the capped exposure-adjusted forecast alone), and the five close columns as the only input; and the relation width r.

Variant	NASDAQ: IRR / SR / IC	NYSE: IRR / SR / IC
no risk neutralisation	0.79 / 5.00 / 0.067	0.94 / 4.73 / 0.058
no tiered selection (raw forecast)	1.66 / 4.46 / 0.065	0.63 / 2.27 / 0.064
close columns only	0.45 / 3.16 / 0.060	0.75 / 8.21 / 0.060
relation width $r = 4$	1.04 / 4.00 / 0.066	0.61 / 10.03 / 0.063
relation width $r = 32$	1.16 / 5.05 / 0.065	0.87 / 5.01 / 0.065

Removing risk adjustment leaves NASDAQ IC similar and reduces NYSE IC by 0.007 in this single-seed comparison. The tiers improve NYSE SR (7.0 versus 2.3 for the raw forecast) but reduce NASDAQ IRR, where the raw forecast earns more at similar SR. Restricting inputs to five close-related columns reduces NASDAQ IC by about 0.006 and substantially reduces its top-5 return. These comparisons describe the reported arms; they do not establish universal benefits for the scoring layer or feature groups.

M S&P500: WALK-FORWARD REFIT AND SINGLE-FEATURE SIGNALS

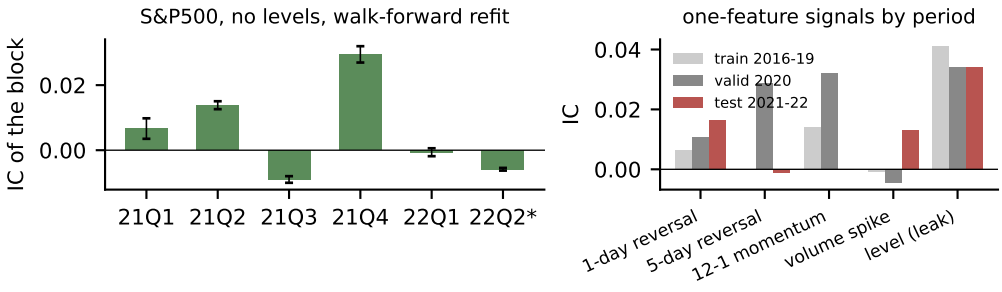

Figure 8: S&P500 without the level inputs. Left: IC per 63-day block of the walk-forward refit (3 seeds), each block scored by a model retrained on every available eligible target day before it. Right: IC of one-feature past-only signals by period; the level column (Section 7) is the only one that persists.

The test period is cut into blocks of 63 trading days. For each block the model is retrained from scratch on every day before it (training, validation and the earlier test blocks) for the validated number of epochs (18 on S&P500, *no levels*), and scores the block; the blocks' scores are joined and evaluated exactly as a fixed-split run. Over 3 seeds this gives IC 0.0066 ± 0.0006, ICIR 0.064 ± 0.007, prec@10 0.524 ± 0.002 and SR -0.14 ± 0.18; the per-block ICs (mean over seeds) are 0.007, 0.014, -0.009, 0.029, -0.001, -0.006. This is a training protocol, not a model change, and the configuration is fixed before the test period; because published numbers use the fixed split, it is reported apart from them.