Electronic Research Archive

https://www.aimspress.com/journal/era

ERA, 34(12): 9317–9351.
DOI: 10.3934/era.2026392
Received: 26 July 2026
Revised: 07 September 2026
Accepted: 10 September 2026
Published: 09 November 2026

Research article

Stock return forecasting and portfolio construction using an event-conditional mixture-of-experts framework

Aivaras Bielskis* **and Igoris Belovas**

Institute of Data Science and Digital Technologies, Vilnius University, Akademijos Str. 4, Vilnius, LT-08412, Lithuania

* **Correspondence:** Email: aivaras.bielskis@mif.stud.vu.lt.

Abstract: The problems we investigated are whether two types of economic information can improve the selection of forecasting models, and whether any resulting gains persist in portfolio construction. The first is the timing of scheduled macroeconomic announcements. The second is how closely the current market resembles past (calm, stressed, or recovery) periods. We proposed an event-conditional mixture-of-experts model (Event-MoE). The model combines a long short-term memory (LSTM) network, a gated recurrent unit (GRU), and a transformer encoder. An event-conditional gate sets the mixture from a calendar of Federal Open Market Committee (FOMC), Consumer Price Index (CPI), and nonfarm-payroll (NFP) releases. A regime layer trains one such mixture per historical regime and blends them by how closely the present resembles each. Since per-stock data are limited, the experts are pooled across all stocks. An augmented variant adds a linear expert, a load-balanced gate, and one-step GARCH and HAR-RV volatility forecasts. The models were evaluated both statistically and through long-only portfolio performance, net transaction costs, and an equal-weight benchmark. Statistical significance was assessed using the Diebold–Mariano and Jobson–Korkie/Memmel tests, and robustness was verified through a walk-forward check. The central finding was that forecast accuracy and portfolio value are distinct objectives. So a more-accurate forecast does not automatically translate into a better portfolio.

Keywords: mixture of experts; event-conditional gating; regime similarity; return forecasting; portfolio construction; macroeconomic announcements; deep learning

1. Introduction

Forecasting equity returns remains challenging, not because modern models lack sufficient predictive capacity, but because the target itself is unstable and noisy. A useful signal in past data is typically weak, short-lived, and difficult to distinguish from random noise. The relationship shifts as the market

transitions between different regimes. When the available history is limited but the number of predictors is large, a flexible model is likely to fit noise rather than the true signal [1]. Thus even a simple equal-weight portfolio is hard to beat out of sample [2].

Nevertheless, machine learning has demonstrated its value in this setting. Flexible models capture non-linear interactions among predictors that linear methods cannot represent [1], and large empirical studies report economically meaningful, if modest, gains [3]. Within this progress, standard sequence models still make little use of two structural parts of the problem. The first concerns *which* temporal structure is most informative at a given moment. A single network estimated over the entire sample averages across regimes. No single model fits every market. Calm markets and volatile ones need different models. The second concerns *timing*. A substantial share of market risk falls on scheduled announcement dates: Federal Open Market Committee (FOMC) decisions, Consumer Price Index (CPI) releases, and nonfarm-payroll (NFP) reports [4]. Standard sequence models do not know these dates unless the calendar is supplied as an input. A third problem is how we measure *success*. The model that minimizes forecast error is often not the one that produces the best portfolio [5]. Small forecast errors can have a larger effect when forecasts are converted into portfolio weights [2]. This study investigates whether announcement timing and market-period similarity can help to choose among forecasting experts, and whether any resulting gains persist once the forecasts are used to construct a portfolio.

To address these questions, we propose an event-conditional mixture-of-experts model (Event-MoE). The base model combines three different sequence experts: a long short-term memory (LSTM) network, a gated recurrent unit (GRU), and a transformer encoder. Its gate is event-conditional rather than fixed. It reads the macro-event channels, so the weight on each expert can change as an announcement approaches. A second, regime-level layer trains one mixture per historical regime. At prediction time, it gives more weight to forecasts from past periods that look most similar to the current market window. This soft layer avoids hidden-Markov-model (HMM) state estimation and does not force the current window into a single discrete regime (cf. [6, 7]). A model trained on a single stock has too little data to learn from. Hence, regime experts are fitted using cross-sectionally pooled data across all stocks. An augmented variant adds a linear expert, a load-balanced gate, and two one-step volatility forecasts as inputs. These are a generalized autoregressive conditional heteroskedasticity (GARCH) forecast and a heterogeneous autoregressive realized-volatility (HAR-RV) forecast. The method adopts the hybrid approach of feeding classical volatility-model outputs into a network [8].

Each model variant is compared with simple benchmark forecasters: a zero-return random-walk baseline, an autoregressive (ARIMA) model, and a single-LSTM with no mixture and no gate. This isolates the contribution of the event gate, the regime blend, and cross-sectional pooling. The forecasts are then converted into long-only portfolios under six allocation rules, spanning prediction-driven and prediction-free (risk only) schemes, so that statistical accuracy and portfolio value can be judged separately.

The seven forecasting models under comparison are as follows:

- **NAIVE**: assumes the expected future return is zero (it is the random-walk benchmark);
- **ARIMA**: a linear autoregression on daily log returns (the one-day forecasts are summed over the horizon);
- **Single-LSTM**: one LSTM expert, with no mixture and no gate;
- **Event-MoE**: LSTM, GRU, and transformer experts combined by an event-conditional gate;

- **Event-MoE-Regime**: adds the regime-similarity blend over per-period experts;
- **Event-MoE-Regime-Pooled**: pools the regime experts across the cross-section;
- **Event-MoE-Regime-Pooled (augmented)**: adds a linear expert, a load-balanced gate, and GARCH/HAR-RV inputs.

The long-only allocation rules are:

- **TopK**: equal weight on the top-K names by score;
- **ScoreWeighted**: weight each name in proportion to its positive score;
- **TopKInvVol**: take the top-K names by score, and give more weight to the less-volatile ones;
- **MaxSharpe**: the mean-variance portfolio that aims for the highest Sharpe ratio, built from the trailing mean return and covariance;
- **MinVariance**: the portfolio with the lowest variance, built from the trailing covariance;
- **EqualWeight**: equal weight on every name (the $1/N$ benchmark).

This study makes three contributions. First, it introduces event-conditional routing, which ties the choice among experts to the calendar of scheduled announcements. Second, it adds a regime-similarity layer that blends per-period experts by how closely the present resembles each past period, rather than assigning each date to a single regime. Third, it employs cross-sectional pooling, training each expert on all stocks at once instead of a separate, data-limited model per asset.

The analysis also covers two additional design choices. First, the augmented variant adds a linear branch, a load-balanced gate, and causal one-step GARCH and HAR-RV forecasts. These additions give the model a simpler forecasting option and extra short-horizon volatility information. Second, the two-level mixture remains interpretable. The gate weights show which expert receives more weight at each step. The period weights show which past market periods look most similar to the present. Both sets of weights are used to interpret the results.

The remainder of the paper is organized as follows. Section 2 surveys the related literature. Section 3 describes the data, models, and the portfolio-construction and evaluation protocol. Section 4 presents the empirical evaluation, Section 5 discusses the findings, and Section 6 concludes the paper.

We formalize the study's objectives as three testable hypotheses, which are evaluated in Section 4:

H1. Conditioning the mixture gate on the calendar of scheduled macroeconomic announcements (FOMC, CPI, NFP) improves return-forecast accuracy relative to a single-network baseline.

H2. Blending per-period regime experts by market-window similarity, with the experts pooled across the cross-section of stocks, further improves forecast accuracy and stability.

H3. Gains in forecast accuracy do not translate directly into portfolio performance: forecast accuracy and risk-adjusted portfolio value are distinct objectives, so a more-accurate forecast need not yield a better portfolio.

2. Related literature

The following survey reviews eight main lines of research supporting the study.

2.1. Deep learning for financial time-series forecasting

Currently, deep networks are a common tool for financial prediction. Sezer et al. [3] covered more than one hundred deep-learning studies from 2005 to 2019, organized by forecasting domain and by architecture (convolutional, recurrent/LSTM, deep-belief, reinforcement-learning, and autoencoder models). They find deep models dominant, with the LSTM commonly adopting a single architecture (particularly for trend prediction) and convolutional networks often serving as feature extractors in hybrids. Notably, no model type is superior across the literature. Gu et al. [1] compared ordinary least squares, elastic net, principal-component, and partial-least-squares regression, regression trees, random forests, and neural networks on roughly sixty years of U.S. equity data. A linear benchmark explains very little out of the sample. Trees and neural networks raise the monthly out-of-sample R^2 to several tenths of a percent at the stock level, and above one percent at the portfolio level. The gains trace to a few dominant predictors, chiefly variants of momentum, liquidity, and volatility, and translate into economically large market-timing improvements. Krauss et al. [9] tested deep neural networks, gradient-boosted trees, and random forests on daily S&P 500 data. Each model predicted whether a stock will outperform the cross-sectional median return. The extreme deciles were then traded long and short. The random forest was the strongest single learner, and an equal-weighted ensemble was best overall, with mean daily returns around 0.45% before costs and directional accuracy of roughly 54%–56%. The authors noted that these returns faded later in the sample, as the inefficiency was arbitraged away.

The recurrent experts used in the study were based on two architectures developed outside finance (LSTM [10] and GRU [11]). LSTM internal memory and gates let information flow across very long gaps that earlier recurrent networks could not handle. GRU has two gates (update and reset), one fewer than the LSTM, and fewer parameters. It improved translation quality in their experiments. Greff et al. [12] compared eight LSTM variants over thousands of runs on speech, handwriting, and music tasks. None significantly improved on the standard LSTM once hyperparameters were tuned, and the forget gate and output activation were the critical components. This motivates keeping our recurrent experts deliberately simple. Vaswani et al. [13] introduced the transformer, built entirely on self-attention without recurrence or convolution, which reaches state-of-the-art translation quality at lower training cost. The architecture has since reached finance. Wang et al. [14] applied a transformer to the CSI 300, S&P 500, Hang Seng, and Nikkei 225 from daily closing prices. They reported lower error than LSTM, recurrent neural network (RNN), and convolutional neural network (CNN) baselines (significant under the Mann-Whitney tests). Wen et al. [15] surveyed transformers for time series and found that performance deteriorates for very long inputs, with three to six layers working best.

2.2. Mixture-of-experts and gating

The mixture-of-experts idea was originally introduced by Jacobs et al. [16], who trained local experts competitively under a gating network that partitions the input space, so each specializes on a region of the data. Jordan and Jacobs [17] extended it into a hierarchical, tree-structured gate trained by expectation-maximization, establishing the statistical foundation on which later mixtures were built. For time series, Zeevi et al. [18] proved that gated mixtures of autoregressive experts are universal approximators of stationary prediction functions, which justifies mixing experts on sequences. Ebrahimpour et al. [19] applied multilayer perceptron (MLP)-expert mixtures to trend forecasting on the Tehran

Stock Exchange. They reached an accuracy in the mid-eighties (percent), above stacked generalization, a modified mixture, and a neuro-fuzzy system, with a trading filter that beats buy-and-hold.

In a recent study, Vallarino [20] combined a recurrent network and a linear regression through a volatility-aware gate. The linear expert dominates in calm periods and the recurrent one in volatile ones, cutting error by roughly one-third on volatile firms. The gate was a fixed function of volatility, however, rather than learned. Du et al. [21] proposed a dual-stream network whose event-aware router fuses prices with encoded news, announcements and macroeconomic data, and a clustering of related assets, beating eight baselines, most at long horizons. Yu et al. [22] developed deep statistical jump models. They placed neural experts in a mixture and inferred the active regime via a smoothed relaxation of the Viterbi algorithm. This unified regime switching and expert mixing. Shi et al. [23] introduced TIME-MoE, a sparse mixture-of-experts foundation model pre-trained on over three hundred billion time points. Its additional load-balancing loss helps keep the gate from relying on only one expert. The gate used was driven by a verified calendar of scheduled macroeconomic announcements rather than by volatility, news text, or an inferred latent market state.

2.3. Regime switching and market states

It is well-established that the return-generating process changes with the market state, and the phenomenon has been widely modeled. Franke [6] reviewed Markov-switching and hidden-Markov time series models. The review covers EM estimation of the latent state, Viterbi decoding of the most likely path, and extensions to autoregressive, ARCH, and GARCH dynamics. Segnon et al. [24] developed an autoregressive Markov-switching GARCH-MIDAS model and investigated whether geopolitical-risk indices improve volatility forecasts. They improved the single-state specification, but a regime-switching component captured much of the same information on its own. Chen and Chien [25] proposed a threshold-based regime-switching volatility model (a realized double-hysteretic GARCH with skew-t innovations). The model switches regime only when the recent price path crosses a threshold, and its parameters are estimated using Markov chain Monte Carlo simulation. It improved 1% and 2.5% value-at-risk and expected-shortfall forecasts over a four-year out-of-sample period covering COVID-19, winning in three of the four markets (the United States, Japan, and South Korea).

In a recent work, Blake et al. [7] took a softer approach that blends regimes instead of switching between them. They compared regime-switching methods for S&P 500 realized volatility over eleven years. The winner was a coefficient-based soft-clustering scheme, which weights regime-specific forecasts by estimated probabilities rather than committing each date to one state. It outperformed the alternatives, most clearly during turbulent periods. Pagliaro [26] coupled a regime-aware gradient-boosting model with a rolling hidden-Markov detector and a look-ahead-free walk-forward protocol. It was validated on dozens of constituents with block-bootstrap intervals and a deflated Sharpe ratio. The portfolio Sharpe was above one, but the deflated Sharpe (about 0.69) was not significant once multiple testing was accounted for. The regime layer here sits on the soft side. Rather than assign discrete states, it scores how strongly the current window resembles each historical period and blends the corresponding experts.

2.4. Volatility modeling

Although our objective is to forecast returns rather than volatility, the volatility-focused literature plays two important roles in the design. First, classical volatility models provide input features for

the augmented variant. Second, the closest benchmark for the evaluation methodology comes from index-volatility forecasting. Bollerslev [27] introduced the generalized autoregressive conditional heteroskedasticity (GARCH) model, extending Engle's ARCH so that past conditional variances enter the variance equation. A simple GARCH(1,1) fits volatility better than a high-order ARCH and remains the econometric baseline. Kim and Won [8] integrated an LSTM with GARCH-type models, namely, standard GARCH, exponential GARCH, and an exponentially weighted moving average (EWMA) — feeding their fitted parameters and one-step variance forecasts into the network. On a stock-index series, the hybrid cuts mean absolute error by roughly one-third and mean squared error by more than one-half against a feed-forward hybrid. Diebold–Mariano tests confirmed the gain.

A similar hybrid approach in the cryptocurrency market, proposed by García-Medina and Aguayo-Moreno [28], embeds GARCH-family parameters as features of an LSTM and likewise finds that the hybrid and deep models outperform standalone GARCH models in terms of volatility forecast errors, while evaluating the resulting portfolios using the Sharpe ratio. We adopt this mechanism directly. In addition to the return target, we compute one-step-ahead volatility forecasts from both GARCH and HAR-RV models and provide these forecasts as additional inputs to the augmented variant.

Recently, Taneva-Angelova and Granchev [29] compared classical models (HAR–RV, ARIMA, GARCH) with deep-learning and transformer architectures. The target was the realized variance of the S&P 500, NASDAQ 100, and Dow Jones over 2000–2025. Scoring uses quasi-likelihood (QLIKE), root mean square error (RMSE), mean absolute error (MAE), and the Diebold–Mariano test. The deep models, especially transformers, achieved the lowest errors, most at short horizons and during structural breaks. Their direct comparison of econometric and deep-learning forecasters, based on error metrics and the Diebold–Mariano test, provides the evaluation protocol that we apply for returns.

2.5. Event, announcement, and sentiment signals

Halousková and Lyócsa [4] provided empirical support for conditioning on scheduled macroeconomic events. They constructed measures of attention and sentiment toward the economy, and used these to forecast the volatility of more than four hundred U.S. stocks over roughly a decade. The results indicated improved forecast accuracy for nearly every stock and across all economic sectors, with the gains concentrated on days characterized by extreme price movements. The improvement reached up to about fifteen percent on such days. Among the macroeconomic variables, attention to Federal Open Market Committee decisions was the most important, though the predictive value varied with the data source. That attention and sentiment peaked around scheduled releases was the motivation for an event-conditional gate. The event channel was deliberately narrower than their free-text sentiment measure, relying only on a verified FOMC, CPI, and NFP release calendar. This preserved a compact, auditable set of inputs.

2.6. Ensembles and hybrids

Ganaie et al. [30] surveyed a wider landscape of deep ensembles, including bagging, boosting, stacking, negative-correlation learning, and decision fusion. Mixture-of-experts is a related approach, although they do not list it as one of the main ensemble types. The authors concluded that ensembles consistently outperform single networks, at the cost of added computation and a tendency to lose diversity. Within finance, the deep-and-tree ensembles of Krauss et al. [9] already beat their individual

member models on out-of-sample returns. Chen et al. [31] took the idea further with a two-stage design. Forecasts from an LSTM, a GRU, a deep multilayer perceptron, and a random forest were first combined through a time-varying-weight regression, and then fed into a diversified mean-variance optimizer. Their pipeline improved point-forecast accuracy and produced portfolios with stronger risk-adjusted performance than the baseline portfolios it tested against, making it the closest predecessor of the forecast-then-allocate structure used here. Our own prior study [32] evaluated LSTM, GRU, transformer, XGBoost, and deep-MLP models, together with simple-average and adaptively weighted (ridge and elastic-net) ensemble combinations, for forecasting the daily prices of five major indices. The conclusion was that high statistical accuracy did not reliably translate into profitability and that the ensemble combinations were frequently among the best performers. The present design replaces those fixed and regression-based combinations with a single learned gate whose weights depend on the event calendar.

2.7. From forecasts to portfolios

Turning forecasts into portfolios presents a well-known challenge. Evaluating the naive $1/N$ portfolio against fourteen optimized mean-variance rules, DeMiguel et al. [2] found that none consistently outperformed it out of sample in terms of the Sharpe ratio, certainty-equivalent return, or turnover, as estimation error offset their theoretical advantages. To reduce that error, Ledoit and Wolf [33] shrunk the unstable sample covariance matrix toward a structured target, the practitioner form of their large-dimensional estimator [34]; in higher dimensions, where the number of assets approaches the sample size, random-matrix theory motivates nonlinear shrinkage that corrects each sample eigenvalue individually [35, 36], as recently applied to portfolio allocation under complex covariance structures [37]. This raises the realized information ratio without changing any other step. The rules and benchmarks are based on classical asset-pricing theory. These are the capital-asset-pricing model of Sharpe [38] and the size/book-to-market factor model of Fama and French [39]. Zaimović et al. [40] studied how many names suffice for diversification and found no single optimal number. It depends on the market, risk measure, and period, and has grown over time. On the reinforcement-learning side, Kashif and Ślepaczuk [41] learned portfolio weights with a soft actor-critic. Its reward embeds transaction costs, turnover penalties, and diversification constraints. The agent is competitive on some markets, but yields no statistically significant excess return over buy-and-hold once costs and robust inference are accounted for. Two recent works frame our evaluation premise directly. Wade [5] tested whether graph neural networks improve realized-volatility forecasts, and whether better forecasts give better portfolios. The lowest-error, best-ranking, and best-Sharpe configurations are three different models. Accuracy and portfolio value are related but not interchangeable. Hwang and Zohren [42] took this to its conclusion with a signature-informed transformer that learns portfolio weights end to end by optimizing a conditional-value-at-risk objective, bypassing point forecasting. We retain the modular, interpretable forecast-then-allocate structure and leave the end-to-end approach for future work.

2.8. Extensions and adjacent approaches

Two further directions follow naturally and are taken up again in the future-work discussion. Recent advances in foundation models for finance suggest a pre-training approach that goes beyond the small, dense mixture employed in this study. One example is the billion-parameter FinCast of Zhu et al. [43].

It is pre-trained on tens of billions of time points and evaluated zero-shot against general time-series foundation models. Graph neural networks provide a natural extension for allowing cross-sectional structure to inform the forecasting model. Wu et al. [44] organized them into recurrent, convolutional, autoencoder, and spatio-temporal families. Wade [5], however, cautioned that such a cross-sectional structure improves outcomes only when it can be translated into effective portfolio decisions.

3. Data and methods

We analyze large, liquid US-dollar stocks, aiming for names that move differently, rather than just including a large number of them. So the stocks are screened to be weakly correlated. We keep the biggest companies by market value. Any later name whose daily-return correlation with an already-selected one exceeds a threshold is dropped. This yields a single-currency set of forty stocks across technology, financial, energy, healthcare, consumer, and materials sectors (see Table 1). Forty names give enough spread for a cross-sectional portfolio while keeping the set liquid and weakly correlated. The universe is constructed retrospectively rather than using point-in-time constituent information. Therefore, the reported results should be interpreted as conditional on this fixed stock universe rather than as a fully point-in-time investable strategy.

3.1. Data and partitioning

Daily prices are obtained from Yahoo Finance (auto-adjusted for splits and dividends). The implementation is hosted on GitHub [45]. The sample runs from January 1, 2000 to May 31, 2026, and is split chronologically into three non-overlapping segments as shown in Figure 1. The *training* window covers January 1, 2000 to January 1, 2021, the *validation* window spans from January 1, 2021 to January 1, 2023, and the *held-out test* window runs from January 1, 2023 to May 31, 2026. All reported results are computed on the test window, which is never used for fitting or selection.

Table 1. The forty stocks used in the analysis (listed alphabetically by ticker) are USD-listed large-cap names screened to be weakly correlated.

AAPL	Apple	AMGN	Amgen	ASML	ASML	AZN	AstraZeneca
BABA	Alibaba	BAC	Bank of America	BHP	BHP	BLK	BlackRock
C	Citigroup	CVX	Chevron	GE	GE Aerospace	GS	Goldman Sachs
HD	Home Depot	HSBC	HSBC	IBM	IBM	JPM	JPMorgan
KO	Coca-Cola	MA	Mastercard	MS	Morgan Stanley	MUFG	Mitsubishi UFJ
NEE	NextEra Energy	NVDA	Nvidia	NVO	Novo Nordisk	NVS	Novartis
PG	Procter & Gamble	PM	Philip Morris	RIO	Rio Tinto	RY	Royal Bank of Canada
SAN	Banco Santander	SAP	SAP	SHEL	Shell	TCEHY	Tencent
TD	TD Bank	TM	Toyota	TMO	Thermo Fisher	TSM	TSMC
TTE	TotalEnergies	VZ	Verizon	WFC	Wells Fargo	XOM	Exxon

Model and hyperparameter selection uses the validation segment only. The per-ticker networks are fitted on the training window, with the validation segment used for early stopping. The regime-pooled experts, regime profiles, per-period target scalers, and the ARIMA baseline are estimated on the full pre-test sample (training plus validation). All of these use only dates before the held-out test window.

No information from the held-out test period is used for fitting, scaling, model selection, regime-profile construction, or target scaling.

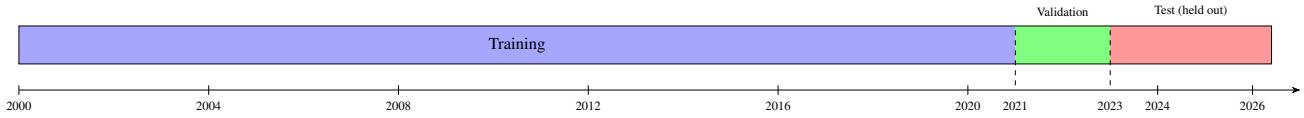

Figure 1. Dataset partitioning: a training window (2000–2021), a validation window (2021–2023) for model and hyperparameter selection, and a held-out test window (2023–2026) used only for evaluation.

We consider three forecast horizons: one trading day (1D), one trading week (1W, five trading days), and one month (1ME). The monthly target is the return to the last business day of the following month, so its length varies slightly from month to month. Diversifying the screened set of stocks across sectors gives the cross-sectional portfolio layer something to allocate across, addressing the difficulty of beating equal weighting in a thin, highly correlated set of stocks (cf. [2, 40]). Figure 2 summarizes the full pipeline.

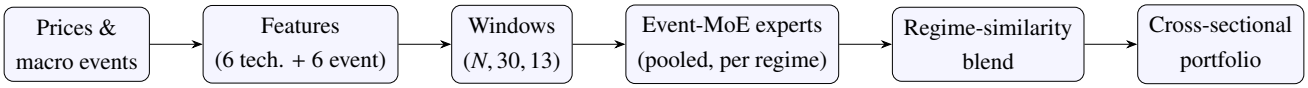

Figure 2. The full pipeline. Prices and the macro-event calendar become features and sliding windows. The Event-MoE mixture forecasts from them, with experts pooled across tickers and one set per regime. The forecasts are blended by regime similarity and turned into a cross-sectional long-only portfolio.

3.2. Macro-event data

The event signal is constructed from a hand-verified calendar of the three macroeconomic announcements that consistently drive U.S. market movements: FOMC rate decisions, CPI releases, and nonfarm payroll (NFP) reports. The resulting sample contains approximately 870 dated events. Conditioning on such scheduled announcements is known to improve volatility forecasts, with the largest gains on extreme-move days [4]. The calendar is audited for real-world irregularities, for example, shifted or canceled releases, so the event features reflect what was actually known on each date rather than an idealized schedule.

3.3. Features

From each price series, we construct thirteen features at each date t, using only the information available up to t. The features are grouped into three categories. Six *technical* features summarize recent price dynamics. They are the one-day log return (`logret1`), its twenty-day rolling standard deviation (`vol20`), the twenty-day log return (`logret20`), a twenty-day range measure (`range20`), the drawdown from the sixty-day high (`dd60`), and the daily high–low range (`hl_range`). A *horizon* feature, `days_to_target`, records the calendar distance to the date being forecast. Six *macro-event* features encode the calendar. For each of FOMC, CPI, and NFP, there is a `days_to_X` countdown to the next such release, clipped to 0–5. Zero marks the release day. Five stands for any release five or more days ahead, or none upcoming. Each event also has an `is_X` indicator, set to one on the

release day. The macro-event features are placed as the last six columns of the input, which is what the event-conditional gate (Section 3) reads.

The prediction target is the forward log-return over the horizon of h trading days,

$$y_{\text{ret}}(i, t, h) = \log P_i(t + h) - \log P_i(t),$$

where $P_i(t)$ is the adjusted closing price of asset i at time t. The target is the return, not the price level or its volatility. Each ticker's feature frame is turned into overlapping windows of length $L = 30$ trading days, producing arrays of shape (N, L, F) with $F = 13$ (15 for the augmented variant). The recurrent and attention experts consume these windows directly. Every feature channel is standardized by a per-feature z-score whose mean and standard deviation are computed over the *training* windows only. The raw features span several orders of magnitude, from calendar counts of order ten to log returns of order 10^{-2}. Without scaling, the large-scale columns would dominate the gradients. Computing the scaler on the training window alone keeps it leak-free.

3.4. Mixture-of-experts model (Event-MoE)

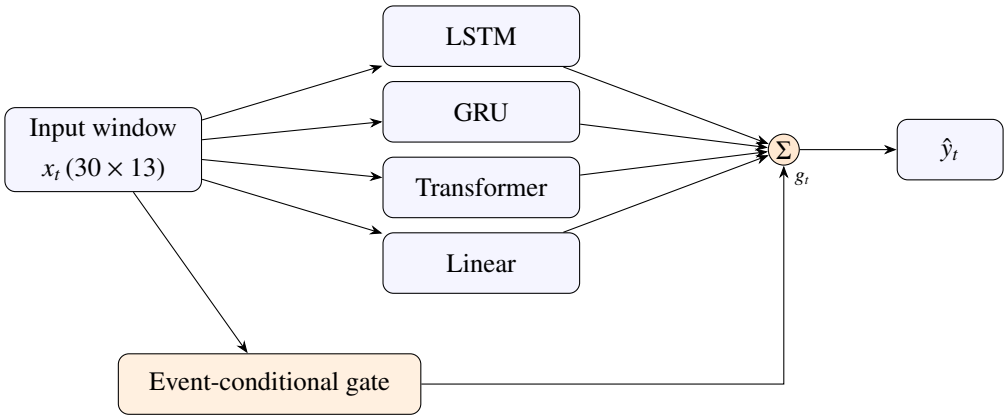

Figure 3. The Event-MoE forecaster. The input window is processed in parallel by heterogeneous experts. The event-conditional gate produces weights g_t, and the final forecast is their weighted combination, $\hat{y}_t = \sum_e g_{t,e}\, f_e(x_t)$. The base three-expert Event-MoE does not include the linear branch.

Event-MoE combines three heterogeneous sequence experts under a single learned gate. Each expert e maps the 30-day input window x_t to a scalar forecast $f_e(x_t)$. The experts differ in how they summarize the window, while the event-conditional gate determines the contribution of each expert for the current input (see Figure 3). The combined forecast is the gate-weighted sum

$$\hat{y}_t = \sum_e g_{t,e}\, f_e(x_t), \qquad \sum_e g_{t,e} = 1,\ g_{t,e} \geq 0,$$

where $\hat{y}_t$ is the combined forecast for date t; e denotes the experts; x_t is the 30-day input window ending at t; $f_e(x_t)$ is expert e's scalar forecast; and $g_{t,e}$ is the gate weight on expert e at t, which is non-negative and summing to one across experts, so the output is a convex combination of the experts. The motivation is that no single model type performs well across all regimes. A recurrent cell is well-suited to persistent trends, a lighter recurrent cell to short-lived reversals, and self-attention to abrupt,

localized moves, such as those around announcements. Mixing them behind a learned gate has both a theoretical basis for time series [16–18] and prior empirical support for stock forecasting [19]. Its value in the present setting is assessed in Section 4.

3.5. LSTM expert

The LSTM expert [10] consists of a stack of recurrent layers with widths {256, 128, 64, 32} over the 30-step window; all but the last return sequences, with `SpatialDropout1D`(0.2) between layers, followed by layer normalization, dropout (0.2), and a final `Dense(1)` output layer. It captures long-range temporal dependencies, and the plain LSTM remains a strong default that is difficult to outperform once properly tuned [12].

3.6. GRU expert

The GRU expert [11] is a lighter gated-recurrent variant with the same structure at widths {128, 64, 32}. With fewer parameters, it is well-suited to a shorter-horizon, reversal-type structure.

3.7. Transformer expert

The transformer expert [13] applies self-attention over the $L = 30$ input window, projecting each step to width d and allowing every step to attend to every other step. For a single attention head, queries, keys, and values are learned linear projections of the block input $X \in \mathbb{R}^{L\times d}$, and the attention operation is

$$\text{Attention}(Q, K, V) = \text{softmax}\Big(\frac{QK^\top}{\sqrt{d_k}}\Big)V.$$

Here, $L = 30$ is the number of input steps and d is the feature width. The matrix $QK^\top$ contains pairwise similarity scores between steps; d_k is the key dimension, whose square-root scaling stabilizes these scores; and the row-wise softmax converts them into attention weights that determine the contribution of each value vector. Each output row is a weighted combination of all L input steps, with weights determined by scaled query-key similarity. The expert stacks three transformer blocks with widths {128, 64, 32}, each using $h = 4$ attention heads, a residual connection, layer normalization, and a position-wise feed-forward sub-layer. The final representation is obtained by global average pooling and mapped to a `Dense(1)` output. No positional encoding is added; sequence ordering is instead captured by the recurrent experts, while self-attention allows this expert to directly weight any step in the window. This makes it suitable for capturing localized spikes around announcements [15].

3.8. Event-conditional gate

The gate is the first of the two mechanisms we propose. It separates ordinary state information from scheduled-event information before it weights the experts. It is a two-stream multilayer perceptron (Figure 4). One stream reads the non-event part of the last time step's feature vector, s_t, and the other reads the six macro-event channels, m_t. Their representations are concatenated and mapped to a softmax over the experts,

$$g_t = \text{softmax}\big(W\,[\,\sigma(W_b s_t)\;;\;\sigma(W_e m_t)\,]\big),$$

where σ is the GELU non-linearity and the two streams are a `Dense`$(32, \text{GELU})$ on s_t and a `Dense`$(16, \text{GELU})$ on m_t (concatenated, with dropout) before the softmax. Because m_t enters directly,

the routing between experts can change as an FOMC, CPI, or NFP release approaches or lands. These dates correspond to periods when announcement risk is known to be concentrated [4]. The event channels are not exclusive to the gate. They are also among the features in the input window that every expert reads. So each expert can already respond to announcements on its own. What the gate adds is that the *routing* among experts is itself event-conditional. The mixture weights depend on how close a release is, not only on the window's dynamics. The gate is deliberately small (a few dense layers), so the event signal can steer the mixture without itself becoming the model. Section 4 examines whether routing shifts around announcements.

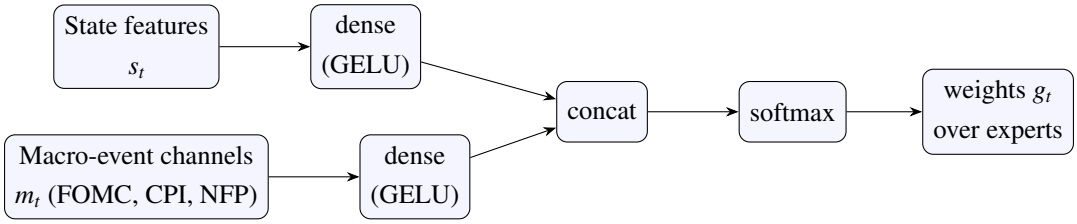

Figure 4. The event-conditional gate. One stream reads the window's state features s_t. A second reads the six macro-event channels m_t. The two are concatenated and mapped by a softmax to the expert weights g_t, so routing can shift as a release approaches.

3.9. Loss and optimization

Table 2. Hyperparameters of the forecasting models (main configuration, lookback $L = 30$).

Model	Hyperparameters
LSTM expert	stacked layers $[256, 128, 64, 32]$; SpatialDropout1D 0.2 between layers; LayerNorm; dropout 0.2; output Dense(1)
GRU expert	stacked layers $[128, 64, 32]$; same head as the LSTM expert
Transformer expert	projection to width d; encoder blocks $[128, 64, 32]$; 4 attention heads; feed-forward $2d \to d$; GlobalAveragePooling; dropout 0.2; output Dense(1)
Linear expert	Dense(1) on the last time step (four-expert Event-MoE only)
Event-conditional gate	Dense(32, GELU) on the base features and Dense(16, GELU) on the six macro-event channels; concatenated; dropout 0.2; softmax over experts
Training (all neural)	Adam, learning rate 10^{-3}; batch size 128; up to 500 epochs with early stopping (patience clip (epochs/10, 10, 25), min-improvement 10^{-4}, restore-best-weights); loss $\text{Huber}_{\delta=1}$ + 0.3·directional (the four-expert Event-MoE adds 0.01·gate load-balance); seed 42
Regime pooling	one expert set per period; $\leq$ 8000 windows per period (subsampled); reduced epoch budget
ARIMA	AR order chosen from $\{2, 1, 0\}$ on daily log returns; multi-step by recursion

The experts and the gate are trained jointly on a robust, direction-aware objective. Its core term is a Huber loss on the standardized target. This loss behaves quadratically for small errors but grows only linearly for large deviations, preventing fat-tailed return outliers from dominating training. We augment it with a directional term (weight 0.3) that rewards forecasts whose sign matches the realized return. It is scaled by the target variance, so the loss behaves consistently whether returns are modeled

in raw form or after standardization. Together, the two terms encourage the model to predict both the magnitude and, most importantly, the direction of returns. For the four-expert Event-MoE, we add one additional term (weight 0.01) to prevent the gate from collapsing onto a single expert. This term penalizes imbalanced expert usage by measuring how evenly the gate distributes weight across experts, following the load-balancing principle of the switch transformer and TIME-MoE [46, 47]. A routing outcome in which any single expert receives more than 0.8 of the total weight is flagged.

Training uses Adam (learning rate 10^{-3}), batch size 128, up to 500 epochs with early stopping (patience clip (epochs/10, 10, 25), `min_delta` 10^{-4}, restore-best-weights) on the validation segment, dropout 0.2, and a fixed seed of 42. The target is standardized with train-set statistics for training and un-scaled at prediction (pooled regime experts use per-period target scaling). These values were selected by a combination of grid and random search on the validation segment, with candidates ranked by the validation-set information coefficient averaged across the 40 stocks; the selected configuration was applied uniformly to all models. Table 2 collects the hyperparameter settings of the forecasting models.

3.10. Regime-similarity blend

The regime layer is the second mechanism we propose (see Figure 5). Rather than fit one model to the whole history, we train a separate Event-MoE on each of several spans of the pre-test sample. At inference, we blend their forecasts by how much the current window resembles each span. Each period is summarized by a *profile* — a simple mean-and-dispersion descriptor of its windows; the incoming test window is described the same way, and periods whose profile is closer to it receive proportionally more weight. This is a soft alternative to hard regime assignment [6]. It needs no HMM-style state estimation and never forces the present into a single discrete regime. The resulting weights read directly as "how much the present resembles each past regime."

Concretely, each length-L window $X_t \in \mathbb{R}^{L\times F}$ ($L = 30$, $F = 13$, or 15 for the augmented variant) is summarized by a profile stacking the per-feature mean and standard deviation over the window,

$$\phi(X_t) = [\,\bar{x}_{t,1}, \ldots, \bar{x}_{t,F},\ s_{t,1}, \ldots, s_{t,F}\,] \in \mathbb{R}^{2F}, \quad \bar{x}_{t,f} = \frac{1}{L}\sum_{l=1}^{L} X_t[l,f], \quad s_{t,f} = \Big(\frac{1}{L}\sum_{l=1}^{L}\big(X_t[l,f] - \bar{x}_{t,f}\big)^2\Big)^{1/2}.$$

Profiles are standardized by the training-window statistics μ_ϕ, σ_ϕ (computed on pre-test windows only), $\tilde{\phi}(X_t) = (\phi(X_t) - \mu_\phi) \oslash \sigma_\phi$, and each regime profile is the standardized mean profile over that period's windows $\mathcal{I}_k$,

$$p_k = \Big(\tfrac{1}{|\mathcal{I}_k|}\textstyle\sum_{i\in\mathcal{I}_k} \phi(X_i) - \mu_\phi\Big) \oslash \sigma_\phi.$$

The distance is Euclidean, $d(X_t, p_k) = \big\|\tilde{\phi}(X_t) - p_k\big\|_2$, and the blend weights are a softmax of the negative scaled distances,

$$\alpha_{t,k} = \frac{\exp(-d(X_t,p_k)/\tau)}{\sum_{j=1}^{K}\exp(-d(X_t,p_j)/\tau)}, \qquad \hat{y}_t^{\text{reg}} = \sum_{k=1}^{K}\alpha_{t,k}\,\hat{y}_{t,k},$$

with temperature $\tau = 1$; as $\tau \to 0$, the weights approach a hard nearest-profile assignment, and larger τ gives a more uniform blend.

How the spans are defined matters more than the blending rule. We first tried narrow, event-defined regimes. These were a dot-com bust, the 2008 financial crisis, the COVID-19 shock, and extended

growth phases. The shortest of these are the problem. A crisis of only a year or two leaves its per-regime expert with very little data to learn from. Its forecasts come out worse rather than better. The final design therefore uses a few broad, deliberately overlapping spans instead: two decade-length windows (2000–2010 and 2011–2020) and two longer reference spans (the full pre-test history and a central 2013–2022 window). They are kept wide precisely so each expert has enough windows to train on, and the similarity blend interpolates between them. All period windows are restricted to dates before the held-out test set, so no expert, profile, or per-period target scaler ever sees a test window.

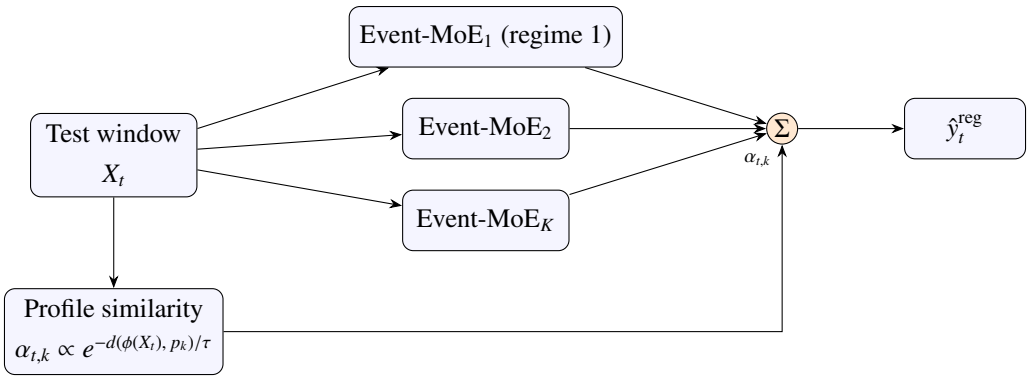

Figure 5. The regime-similarity blend. The test window is scored by every per-period expert, and the forecasts are combined by weights $\alpha_{t,k}$ measuring how closely the current window resembles each regime's profile p_k.

3.11. Pooled and augmented variants

A per-ticker regime model needs a lot of data, since each period's expert sees only one ticker's windows. We therefore *pool* the experts across all stocks. For each period, a single Event-MoE is trained on all tickers' windows at once and reused for every ticker. This multiplies the data per expert by the number of names. The pooled experts are trained once per (scenario, target, horizon, lookback), with each period's pool capped at 8000 windows (subsampled) and a reduced epoch budget for speed. A further variant adds three modifications from the volatility and MoE literature. First, a fourth linear expert (a single linear map on the last time step) gives the gate a low-variance option for calm stretches [20]. Second, a load-balanced four-way gate discourages the gate from collapsing onto a single expert [23, 47]. Third, two causal volatility forecasts are added as inputs, following the hybrid LSTM–GARCH idea of Kim and Won [8]. These are a one-step-ahead GARCH(1,1) variance forecast and a one-step-ahead HAR-RV forecast [48]. In essence, GARCH(1,1) predicts the next day's variance as a blend of a long-run average level, the most recent squared return, and the most recent variance. HAR-RV predicts realized variance from its own recent daily, weekly, and monthly averages. Both are estimated *causally*. They use only data up to each date and are stored as the forecast for the next day (truncation-verified). The GARCH model assumes a constant mean return, so only the variance moves over time. It is re-estimated once a month (every 21 trading days) on an expanding window that keeps all history to date. The first fit begins only after a 250-day (about one-year) warm-up provides enough data to be stable.

3.12. Baseline forecasters

All models are compared against three baselines scored on exactly the same dates and targets. The NAIVE baseline predicts zero forward return (a martingale/random-walk benchmark). ARIMA fits an autoregression on daily log returns over the pre-test segment, with the order chosen from (AR(2), AR(1), AR(0)). At each test date, it forecasts the next daily returns recursively and sums them. Fitting on non-overlapping daily returns, rather than the overlapping multi-day target, keeps it leak-free. It uses only data available at each decision date. The single-LSTM baseline is one LSTM expert with no mixture and no event gate. It is the ablation that isolates the value of the gate and the regime blend.

3.13. Portfolio construction

Forecasts are transformed into a portfolio cross-sectionally. On each rebalance date, the per-ticker forecasts are stacked into a score panel and z-scored across names, removing per-asset level effects so the score reflects relative attractiveness. Six long-only weighting rules are then applied. Three rules are prediction-driven. *TopK* equal-weights the top-K names by score. *ScoreWeighted* gives each name a weight in proportion to its positive score. Very small weights are set to zero. The remaining weights are then rescaled to add up to one. *TopKInvVol* selects the top-K by score but sizes them by inverse trailing volatility. Three are prediction-free classical benchmarks built from a trailing-60-day, ridge-regularized covariance: *MaxSharpe* ($w \propto \Sigma^{-1}\mu$ with μ the trailing historical mean), *MinVariance* ($w \propto \Sigma^{-1}\mathbf{1}$), and *EqualWeight* ($1/N$). A shrunk covariance rather than the raw sample matrix is essential for stable mean–variance weights [33], and the MaxSharpe / MinVariance / EqualWeight trio mirrors the classical mean–variance comparison [49]. In our implementation, this regularization takes the form of diagonal loading—a small ridge added to the diagonal of the trailing sample covariance (proportional to its average variance) before inversion.

3.14. Backtesting protocol

Each combination of a model, a horizon, and an allocation rule gives one portfolio. We call each such portfolio a book. Each book is backtested daily. The portfolio is rebalanced at regular intervals that match the horizon: every 1, 5, or 21 trading days for the daily, weekly, and monthly books (a month is taken as 21 trading days). Weights are held and allowed to drift between rebalances. A linear transaction cost of 5 bps is charged on turnover, the total absolute change in the portfolio weights from one rebalance to the next. Idle cash earns a 2% bond yield, and the all-N methods apply a 1% minimum-weight floor (positions below 1% are dropped and the rest are renormalized). Charging cost on turnover, rather than reporting it separately, follows the cost-aware evaluation practice of recent reinforcement-learning portfolio studies [41] and ensures high-turnover books are penalized where it matters. Decisions are taken at the close and earn from the next day, so the backtest is free of look-ahead.

3.15. Benchmark portfolios

The primary benchmark is equal weighting ($1/N$) on the same stocks, which is difficult to outperform out of sample once estimation error is accounted for (cf. [2]). The $1/N$ benchmark shown in the return tables and equity curves is held buy-and-hold (allocated once and allowed to drift), whereas the

Jobson–Korkie/Memmel test compares each book against a $1/N$ benchmark rebalanced at the same horizon frequency. The prediction-free MinVariance and EqualWeight books, and a per-name buy-and-hold, provide further reference points, so any edge of the prediction-driven books must show up against passive and risk-only alternatives.

3.16. Evaluation metrics and statistical inference

Forecast metrics. For a given stock, let r_i and $\hat{r}_i$ denote the realized and forecast returns over the n evaluation dates, with means $\bar{r}$ and $\bar{\hat{r}}$. Forecast performance is assessed with the following metrics.

- Root-mean-square error and mean absolute error:

$$\text{RMSE} = \sqrt{\tfrac{1}{n}\textstyle\sum_i (r_i - \hat{r}_i)^2}, \qquad \text{MAE} = \tfrac{1}{n}\textstyle\sum_i |r_i - \hat{r}_i|.$$

- The information coefficient (IC), which is the Pearson correlation between the forecast $\hat{r}$ and the realized return r:

$$\text{IC} = \text{corr}(\hat{r}, r) = \frac{\sum_i (\hat{r}_i - \bar{\hat{r}})\,(r_i - \bar{r})}{\sqrt{\sum_i (\hat{r}_i - \bar{\hat{r}})^2}\ \sqrt{\sum_i (r_i - \bar{r})^2}}.$$

- The rank information coefficient (rank-IC) is the Spearman rank correlation, computed as the Pearson correlation above applied to the ranks of $\hat{r}$ and r.
- Directional accuracy, the fraction of correctly signed forecasts:

$$\text{DA} = \tfrac{1}{n}\textstyle\sum_i \mathbf{1}[\text{sign}(\hat{r}_i) = \text{sign}(r_i)]\,.$$

- The error ratio against the naive zero forecast, i.e., the model RMSE relative to that of the constant $\hat{r} \equiv 0$ predictor:

$$\text{RMSE}_{\text{rel}} = \frac{\sqrt{\sum_i (r_i - \hat{r}_i)^2}}{\sqrt{\sum_i r_i^2}}.$$

All forecast metrics are computed per stock and averaged across the 40 stocks; for the walk-forward stability analysis (Section 4.5), the IC is instead recomputed within each calendar sub-period by pooling the stocks (Section 3.17). As a correlation, the IC lies in $[-1, 1]$. Values near zero mean there is no linear relation between the forecast and outcome. Positive values indicate a usable signal, and negative values systematically wrong-sign forecasts. Achievable magnitudes are small. A monthly IC of about 0.05–0.10 is already a meaningful edge. Unusually, large values would be surprising for return prediction. In this leakage-controlled setting, they are better read as a prompt to re-check the pipeline than as evidence of skill.

Portfolio metrics. Let r_t be the daily net-of-cost portfolio return over $t = 1, \ldots, T$, with cumulative value $V_t = \prod_{s \le t}(1 + r_s)$, sample standard deviation s, and cash rate $r_f = 2\%$. Portfolios are evaluated with

$$\text{annualized return } R_{\text{ann}} = \Big(\textstyle\prod_{t=1}^{T}(1 + r_t)\Big)^{252/T} - 1, \quad \text{annualized volatility } \sigma_{\text{ann}} = \sqrt{252}\, s,$$

$$\text{Sharpe} = \frac{R_{\text{ann}} - r_f}{\sigma_{\text{ann}}}, \qquad \text{Sortino} = \frac{R_{\text{ann}} - r_f}{\sqrt{252}\ \text{std}\{r_t : r_t < 0\}},$$

$$\text{max drawdown MDD} = \min_t\Big(\frac{V_t}{\max_{s\le t} V_s} - 1\Big), \qquad \text{Calmar} = \frac{R_{\text{ann}}}{|\text{MDD}|},$$

$$\text{hit rate} = \tfrac{1}{T}\textstyle\sum_t \mathbf{1}[r_t > 0], \qquad \text{turnover} = \tfrac{252}{T}\textstyle\sum_\tau \sum_i \left|w^+_{i,\tau} - w^-_{i,\tau}\right|,$$

where in the turnover, the sum runs over rebalance dates τ and $w^-_{i,\tau}, w^+_{i,\tau}$ are the pre- and post-rebalance weights of asset i (the quantity on which the transaction cost is charged). Sharpe is the risk-adjusted performance in the sense of [50].

Statistical inference. Two significance tests are reported. The Diebold–Mariano test [51] compares each model against the naive baseline on the squared-error loss differential $d_t = e^2_{0,t} - e^2_{m,t}$, where $e_{0,t} = r_t$ is the naive (zero-forecast) error and $e_{m,t} = r_t - \hat{r}_t$ the model error. With $\bar{d}$ the mean differential and $\hat{\gamma}_k$ its lag-k autocovariance, the long-run variance uses a Newey–West (Bartlett) heteroskedasticity and autocorrelation (HAC) estimator with maximum lag $h - 1$ for an h-step forecast:

$$\hat{\omega}^2 = \hat{\gamma}_0 + 2\sum_{k=1}^{h-1}\Big(1 - \tfrac{k}{h}\Big)\hat{\gamma}_k,$$

which handles the overlap of multi-step forecasts. The statistic, with the Harvey–Leybourne–Newbold small-sample correction [52], is

$$\text{DM}^* = \frac{\bar{d}}{\sqrt{\hat{\omega}^2/n}}\ \sqrt{\frac{n + 1 - 2h + h(h-1)/n}{n}}, \qquad p = 2(1 - F_{t_{n-1}}(|\text{DM}^*|)),$$

with $F_{t_{n-1}}$ the Student-t cumulative distribution function (CDF). The Diebold–Mariano (DM) test is performed separately for each stock; Table 5 reports the mean stock-level DM statistic. This is the protocol used in recent index-volatility studies [29].

The Jobson–Korkie test [53] with Memmel's correction [54] tests whether a book's Sharpe ratio $\text{SR}_p = \mu_p/\sigma_p$ differs from that of its $1/N$ benchmark $\text{SR}_b = \mu_b/\sigma_b$. For this test statistic, the Sharpe ratio is computed directly from daily returns as μ/σ, without subtracting the cash rate; this test-specific definition therefore differs from the descriptive portfolio Sharpe reported above. With $\rho = \text{corr}(r_p, r_b)$ over T observations,

$$\hat{\theta} = \tfrac{1}{T}\Big[\,2(1-\rho) + \tfrac{1}{2}(\text{SR}_p^2 + \text{SR}_b^2 - 2\,\text{SR}_p\text{SR}_b\,\rho^2)\Big], \qquad z = \frac{\text{SR}_p - \text{SR}_b}{\sqrt{\hat{\theta}}} \ \sim\ \mathcal{N}(0,1),$$

with two-sided p-value $p = 2\,\Phi(-|z|)$. Reporting both the forecast-error and the Sharpe-ratio tests separates statistical predictability from economic usefulness, since the two need not coincide [5, 42].

3.17. Evaluation protocol

We use a strict, date-based train/validation/test split with model selection on the validation segment only, and a fixed random seed for reproducibility. To guard against reading too much into a single test split, a walk-forward stability analysis recomputes the cross-sectional information coefficient in each

sub-period (year, half-year, or quarter), pooling all stocks within the sub-period rather than averaging the per-stock coefficients as in the main accuracy table (Table 3). It summarizes the mean, the fraction of positive sub-periods, and a t-statistic. An edge is treated as real only if it recurs across sub-periods. Rigorous regime-aware studies adopt similar discipline, including rolling estimation with no look-ahead and deflated performance measures [26].

4. Empirical evaluation results

All quantities in this section are computed on the held-out test window (January 1, 2023 to May 31, 2026), used for no fitting or model selection. Per-feature z-scores use the training window only, and regime profiles and per-period target scalers use only pre-test data. Results use a lookback of $L = 30$ on the 40 stocks of Section 3, and every cross-model accuracy metric (IC, rank-IC, directional accuracy) is reported as a mean across the 40 stocks. Throughout, statistical accuracy and portfolio value are treated as distinct outcomes.

4.1. Forecast accuracy

Table 3 reports the IC per horizon (pooled across all stocks), with rank-IC, directional accuracy, and the RMSE-to-naive ratio at the monthly horizon (Figure 6). The predictable component is small everywhere. Even the best cell, the cross-sectionally pooled regime model at the monthly horizon, reaches only 0.086, and most daily ICs are indistinguishable from zero. Consistency and significance therefore matter more than magnitude. The RMSE-to-naive ratio also serves as a leakage diagnostic. It stays near one (0.983–1.107 monthly). The neural models sit slightly above one because the direction-aware objective trades squared-error for sign accuracy.

Table 3. Forecast accuracy on the held-out window, averaged across the 40 stocks. IC is the Pearson correlation of the forecast and realized return. Rank-IC is its Spearman version. RMSE/naive is each model's RMSE divided by that of the zero forecast (near one is expected). NAIVE is a constant forecast, so it has no IC or directional accuracy. The three highest ICs are shown in green (all monthly) and the three lowest in red (all negative daily or weekly ICs).

Model	IC 1D	IC 1W	IC 1ME	rank-IC 1ME	dir. acc. 1ME	RMSE/naive 1ME
NAIVE	—	—	—	—	N/A	1.000
ARIMA	0.008	0.013	0.038	0.036	57.65%	0.983
Single-LSTM	0.002	0.000	0.010	0.005	51.63%	1.076
Event-MoE	−0.002	−0.026	0.016	0.027	51.81%	1.107
Event-MoE-Regime	0.007	0.012	0.034	0.041	55.60%	1.012
Event-MoE-Regime-Pooled	−0.003	0.021	0.086	0.075	54.74%	1.029
+ augmented	−0.013	0.024	0.069	0.060	54.69%	1.073

Daily, no model has a reliable edge — the largest IC is ARIMA's 0.008, where the deep models are flat or slightly negative. Daily returns are dominated by microstructure noise. The signal grows with the horizon. Weekly, the pooled regime models lead (Event-MoE-Regime-Pooled 0.021, augmented 0.024) while the plain Event-MoE is negative (−0.026). Monthly, the ordering is clearest. The pooled

regime model reaches 0.086 (rank-IC 0.075, directional accuracy 54.74%), ahead of the single-LSTM baseline (0.010) and the plain Event-MoE (0.016). ARIMA is a respectable (0.038), so a well-specified linear model is not easily beaten on accuracy alone.

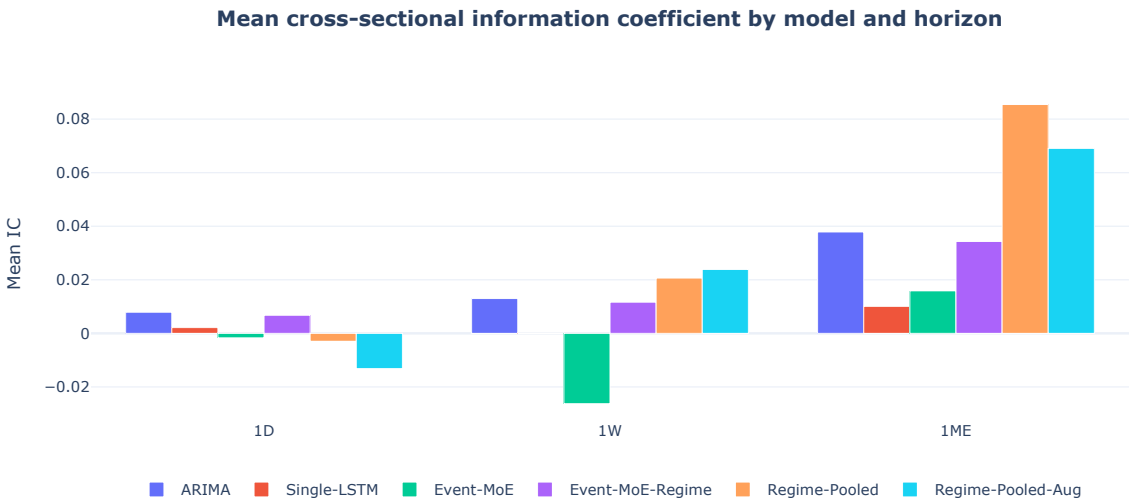

Figure 6. Mean cross-sectional information coefficient by model and horizon. The signal is small throughout and concentrated at the weekly and monthly horizons, where the cross-sectionally pooled regime models lead. In the figures, "Regime-Pooled" and "Regime-Pooled-Aug" denote Event-MoE-Regime-Pooled and its augmented variant.

4.2. The pooling effect

The key comparison is per-ticker regime experts (Event-MoE-Regime) versus pooling them across the whole cross-section (Event-MoE-Regime-Pooled). Pooling multiplies each expert's data by the number of names without changing the model class. The effect is large. Monthly IC rises from 0.034 to 0.086 (about 2.5×), rank-IC from 0.041 to 0.075, and weekly IC from 0.012 to 0.021. Pooling, not the regime structure alone, turns a marginal per-ticker model into the study's most accurate forecaster. Section 5 returns to that attribution.

The augmented variant adds a linear expert, a load-balanced four-way gate, and one-step GARCH and HAR-RV forecasts as inputs. It sharpens accuracy at short horizons, where one-step volatility information matters most. It beats the pooled model weekly (0.024 versus 0.021) but not monthly (0.069 versus 0.086), where realized-variance features say less about a month-ahead return. This is consistent with the volatility literature: GARCH/HAR forecasts capture near-term variance, and their value as return inputs decays with the horizon.

4.3. Portfolio performance

Forecast accuracy is only a means to an end. The real test is whether it survives portfolio construction. Table 4 reports each model's long-only top-7 book at all three horizons, alongside the prediction-free classical books ($1/N$, MinVariance, MaxSharpe), net of the 5-bps cost. Since 1/N is well-known to be hard to beat out of sample [2], the question is whether the model books match it, not whether they

far outperform it. They largely do. At the monthly horizon, $1/N$ earns 30.67% (Sharpe 1.77) and the best book (Event-MoE) sits just above at Sharpe 1.96, with the stronger models clustered around the benchmark. The high $1/N$ return reflects the strong 2023–2026 market, so the comparison is relative.

Table 4. Portfolio performance at all three horizons, net of the 5-bps cost. Top rows of each block: each model's long-only top-7 book. Lower rows: the prediction-free classical benchmarks. Sharpe is over a 2% cash rate. Within each block, the best Sharpe is green and the worst red, with NAIVE excluded. A book that forecasts nothing can still post a high Sharpe, notably its weekly one.

Book	ann. ret	ann. vol	Sharpe	Sortino	max DD	hit
Daily (1D)						
NAIVE · TopK	16.02%	14.32%	0.98	1.49	−16.43%	52.11%
ARIMA · TopK	6.15%	18.05%	0.23	0.33	−24.11%	51.52%
Single-LSTM · TopK	18.57%	18.23%	0.91	1.33	−20.71%	52.11%
Event-MoE · TopK	18.37%	18.25%	0.90	1.34	−21.05%	52.46%
Event-MoE-Regime · TopK	20.14%	18.76%	0.97	1.37	−23.49%	54.22%
Event-MoE-Regime-Pooled · TopK	25.10%	19.39%	1.19	1.75	−21.32%	53.40%
+ augmented · TopK	20.17%	16.84%	1.08	1.61	−13.11%	51.99%
$1/N$ (equal weight)	30.67%	16.18%	1.77	2.49	−18.66%	55.39%
MinVariance	17.09%	12.96%	1.16	1.55	−15.06%	55.15%
MaxSharpe	17.60%	14.43%	1.08	1.52	−16.32%	54.57%
Weekly (1W)						
NAIVE · TopK	44.13%	19.86%	2.12	2.96	−23.75%	57.26%
ARIMA · TopK	34.52%	18.34%	1.77	2.45	−19.56%	56.67%
Single-LSTM · TopK	25.78%	18.42%	1.29	1.87	−20.75%	54.68%
Event-MoE · TopK	17.54%	17.93%	0.87	1.28	−20.97%	51.76%
Event-MoE-Regime · TopK	26.93%	18.52%	1.35	1.84	−17.56%	56.09%
Event-MoE-Regime-Pooled · TopK	30.13%	19.24%	1.46	2.13	−17.48%	56.44%
+ augmented · TopK	25.33%	18.89%	1.24	1.80	−23.48%	53.04%
$1/N$ (equal weight)	30.67%	16.18%	1.77	2.49	−18.66%	55.39%
MinVariance	19.67%	12.97%	1.36	1.83	−14.71%	55.97%
MaxSharpe	22.16%	14.35%	1.40	1.97	−16.01%	54.92%
Monthly (1ME)						
NAIVE · TopK	15.96%	14.25%	0.98	1.49	−16.40%	51.87%
ARIMA · TopK	27.55%	18.79%	1.36	2.00	−23.16%	53.16%
Single-LSTM · TopK	29.29%	17.61%	1.55	2.36	−16.33%	55.85%
Event-MoE · TopK	34.94%	16.78%	1.96	2.89	−16.90%	55.39%
Event-MoE-Regime · TopK	32.83%	17.71%	1.74	2.61	−15.86%	56.21%
Event-MoE-Regime-Pooled · TopK	22.31%	19.73%	1.03	1.55	−22.95%	53.16%
+ augmented · TopK	31.32%	18.98%	1.54	2.19	−20.30%	55.62%
$1/N$ (equal weight)	30.67%	16.18%	1.77	2.49	−18.66%	55.39%
MinVariance	20.93%	12.80%	1.48	1.97	−15.16%	56.32%
MaxSharpe	21.52%	14.02%	1.39	1.93	−17.77%	55.50%

More surprising is that the most accurate model does not give the best portfolio. The most accurate forecaster (pooled regime model, monthly IC 0.086) yields one of the worst books (Sharpe 1.03), while the plain Event-MoE (IC 0.016) tops the prediction-driven books at 1.96 (Figure 7). The top-K rule depends only on the ranking of the top names and their covariance, neither of which tracks a panel-wide IC. Section 5 develops this. The prediction-free MinVariance and MaxSharpe books are competitive throughout (Figure 8) and Figure 9 shows how sensitive the ranking is to K. In a thin, highly correlated set of stocks, risk structure matters at least as much as the expected-return signal.

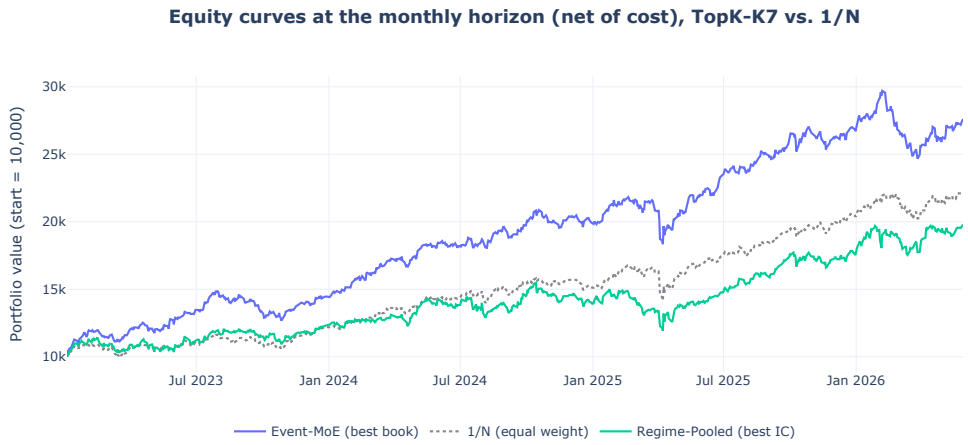

Figure 7. Net-of-cost equity curves at the monthly horizon (start = 10,000): the best prediction-driven book (Event-MoE), the most accurate forecaster's book (Event-MoE-Regime-Pooled), and the $1/N$ benchmark. Higher forecast accuracy does not produce the higher terminal value.

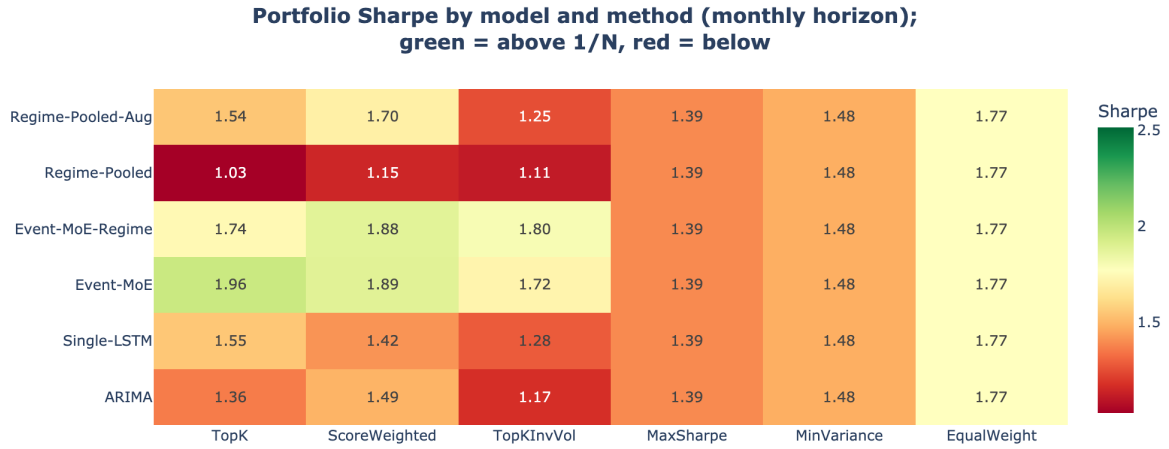

Figure 8. Portfolio Sharpe across the model × method grid at the monthly horizon, with the color midpoint at the $1/N$ Sharpe (green above $1/N$, red below). Most cells sit at or below $1/N$. The green region is confined to the prediction-driven books of the stronger models.

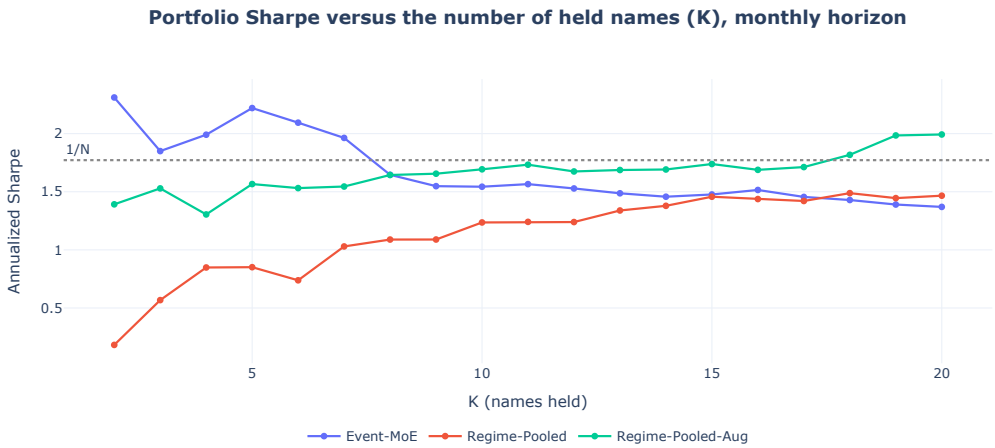

Figure 9. Annualized Sharpe as a function of the number of held names K at the monthly horizon, for the three mixture variants, against the $1/N$ reference line. The ranking of variants is not stable in K, which shows how sensitive the allocation step is.

4.4. *Statistical significance*

Table 5. Significance summary at all three horizons. *DM stat*: Diebold–Mariano statistic against the naive forecast (positive favors the model). *Sharpe*, z, p: Jobson–Korkie/Memmel test of each top-7 book against $1/N$ (bold: $p < 0.05$). Per block, best Sharpe green, worst red (NAIVE excluded).

Model	DM stat	Sharpe	Sharpe $1/N$	z	p
Daily (1D)					
NAIVE	—	1.11	1.74	−1.57	0.117
ARIMA	−0.14	0.42	1.74	−4.32	**<0.001**
Single-LSTM	−0.94	1.02	1.74	−2.36	**0.018**
Event-MoE	−2.12	1.01	1.74	−2.57	**0.010**
Event-MoE-Regime-Pooled	−0.93	1.25	1.74	−1.55	0.121
+ augmented	−1.54	1.18	1.74	−1.49	0.136
Weekly (1W)					
NAIVE	—	1.94	1.75	0.73	0.464
ARIMA	0.60	1.71	1.75	−0.14	0.891
Single-LSTM	−1.37	1.34	1.75	−1.35	0.177
Event-MoE	−2.23	0.99	1.75	−2.41	**0.016**
Event-MoE-Regime-Pooled	−1.93	1.47	1.75	−0.94	0.347
+ augmented	−2.33	1.29	1.75	−1.42	0.157
Monthly (1ME)					
NAIVE	—	1.11	1.75	−1.58	0.113
ARIMA	0.94	1.39	1.75	−1.18	0.236
Single-LSTM	−1.17	1.55	1.75	−0.66	0.507
Event-MoE	−1.53	1.87	1.75	0.38	0.704
Event-MoE-Regime-Pooled	−0.45	1.12	1.75	−1.95	0.052
+ augmented	−1.04	1.53	1.75	−0.69	0.489

Two significance tests qualify these comparisons (Table 5). On forecast error, the Diebold–Mariano test favors the conservative ARIMA (monthly statistic 0.94). This is a squared-error artifact, not economic content. On portfolio value, the Jobson–Korkie/Memmel test finds no prediction-driven book significantly beats $1/N$ at any horizon. Several fall below it daily, and every significant result is a book that sits below $1/N$. The best monthly book (Event-MoE) is statistically indistinguishable from $1/N$ ($z = 0.38$, $p = 0.70$). This is consistent with how hard $1/N$ is to beat once estimation error is counted [2].

4.5. Walk-forward stability and ablation

A single split, however strict, can flatter a model that suited one window. We therefore recompute the cross-sectional IC per test-period year (mean, fraction positive, t-statistic) and treat an edge as credible only if it recurs (Table 6). The monthly horizon discriminates. Only the pooled regime model is both large and stable. Its mean yearly IC is 0.104, positive in 100% of the years, and $t = 2.24$ (Figure 10). It is the only model past two sigma. The augmented variant matches it in mean (0.085) but is noisier ($t = 1.43$). Non-pooled models show no stable monthly edge, and for daily/weekly, none is stable, consistent with Section 4.1.

Table 6. Walk-forward stability of the cross-sectional IC: mean over the test-period calendar years, share of years positive, and the t-statistic of the mean. Only the pooled regime model, at the monthly horizon, clears two sigma.

Model	Mean yearly IC	% years positive	t-stat
Daily (1D)			
ARIMA	0.0057	75%	0.79
Single-LSTM	−0.0036	25%	−0.62
Event-MoE	−0.0095	25%	−0.88
Event-MoE-Regime	−0.0095	25%	−2.15
Event-MoE-Regime-Pooled	0.0059	50%	0.46
+ augmented	−0.0045	50%	−0.31
Weekly (1W)			
ARIMA	−0.0031	50%	−0.38
Single-LSTM	−0.0129	25%	−1.11
Event-MoE	−0.0401	0%	−4.29
Event-MoE-Regime	0.0070	75%	0.69
Event-MoE-Regime-Pooled	0.0254	50%	1.03
+ augmented	0.0276	50%	0.70
Monthly (1ME)			
ARIMA	−0.0398	25%	−1.26
Single-LSTM	−0.0223	25%	−1.78
Event-MoE	−0.0185	25%	−0.81
Event-MoE-Regime	0.0107	75%	0.41
Event-MoE-Regime-Pooled	**0.1044**	**100%**	**2.24**
+ augmented	0.0846	75%	1.43

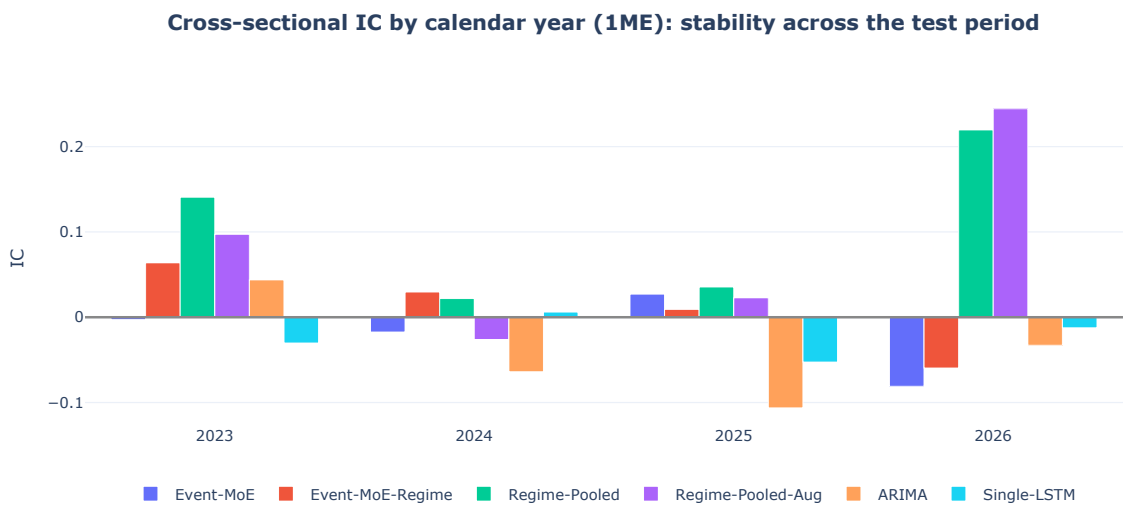

Figure 10. Cross-sectional IC by calendar year at the monthly horizon. Event-MoE-Regime-Pooled is the only model positive in every year. This is the basis for its "stable" verdict in Table 6. The single-network and classical baselines change sign across years.

Table 7. Incremental model-ladder comparison at the monthly horizon. Each row adds one block of structure to the row above (parenthetical: change in mean IC). These are incremental steps, not clean single-factor toggles. Cross-sectional pooling is the dominant contributor.

Configuration (cumulative)	Mean IC (Δ vs. above)
Single-LSTM (no mixture, no gate)	0.010
+ heterogeneous experts & event gate (Event-MoE)	0.016 (+0.006)
+ per-regime experts (Event-MoE-Regime)	0.034 (+0.018)
+ cross-sectional pooling (Event-MoE-Regime-Pooled)	**0.086** (+0.051)
+ linear expert, balanced gate, GARCH/HAR (augmented)	0.069 (−0.016)

The specifications form a ladder, so these numbers are an incremental comparison, not a clean ablation (Table 7). Pooling changes the training sample and the augmented variant bundles several changes. At the monthly horizon, the heterogeneous experts plus the event-conditional gate add only +0.006 over a single-LSTM, and per-regime experts add a further +0.018. Cross-sectional pooling is by far the largest gain (+0.051). It adds more than the rest combined. The augmentation block subtracts (−0.016) monthly while adding weekly. The gain comes from data efficiency, meaning more training windows per expert, not a modeling trick. A single-factor ablation is left to future work.

4.6. The signal across stocks and books

The averages can also be read per name and book. Table 8 lists the ten most predictable names (pooled regime model, monthly). The signal is broad-based, that is, 28 of 40 stocks have positive monthly IC (mean 0.085, Figure 11). This is a genuine cross-sectional tendency, not one or two outliers. Dispersion is nonetheless wide, which is part of why a panel-wide IC need not map onto the few names a concentrated portfolio holds.

Table 8. The ten most predictable stocks by monthly information coefficient under the Event-MoE-Regime-Pooled model. Across the 40 stocks, 28 have a positive monthly IC (mean IC 0.085).

Ticker	IC	Rank-IC	Dir. acc.
SAP	0.286	0.224	62.30%
TM	0.270	0.218	56.50%
SAN	0.268	0.226	56.40%
PG	0.241	0.177	52.50%
WFC	0.240	0.240	58.00%
BHP	0.226	0.257	62.10%
TCEHY	0.223	0.195	52.00%
RIO	0.204	0.208	56.50%
C	0.202	0.206	62.20%
BAC	0.200	0.216	58.00%

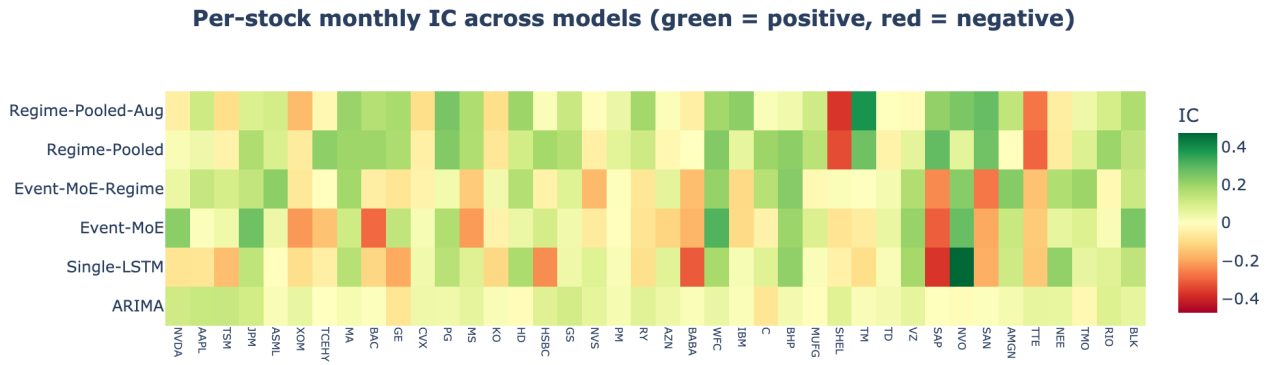

Figure 11. Per-stock monthly IC across models (green positive, red negative). Green predominates for the pooled regime models across most of the forty names, confirming that the monthly edge is broad-based rather than carried by a few stocks.

Across books, the message is clearest: Over the 18 model × horizon top-7 books, mean IC and portfolio Sharpe correlate only $r = 0.28$ (Figure 12), which is weak and, with so few points, is indistinguishable from zero. Accuracy thus explains little of the cross-book Sharpe variation, the second-stage check that the portfolio tests provide. A similarly weak fit-performance link is reported at the index level [32].

The highest-Sharpe books earn their Sharpe through concentration and turnover, which also makes them the most cost-sensitive. Monthly, the best is Event-MoE · TopK with $K = 2$ (Sharpe 2.31, annualized 61.67%), but it turns over about 54× a year against 2.8× for the diffuse ARIMA · TopK-3 (Table 9). That $K = 2$ figure is also fragile. With only two names held, it leans heavily on NVIDIA. NVIDIA returned roughly +1400% over the test window and is one of the two holdings in nearly half of the rebalances. Removing it more than halves the book's Sharpe, from 2.31 to about 1.1. This

single-name dependence is exactly why the main analysis uses the diversified $K = 7$ book.

Opening the $K = 7$ book by name (Figure 13) shows profit concentrated in a few names despite the broad signal. It compounds 10,000 into 27,620, a net of 17,620 after 520 in fees over 465 trades. NVDA is the largest contributor (+5437), with a short tail of detractors (Table 10, Figure 14). This is the per-name counterpart of accuracy versus value.

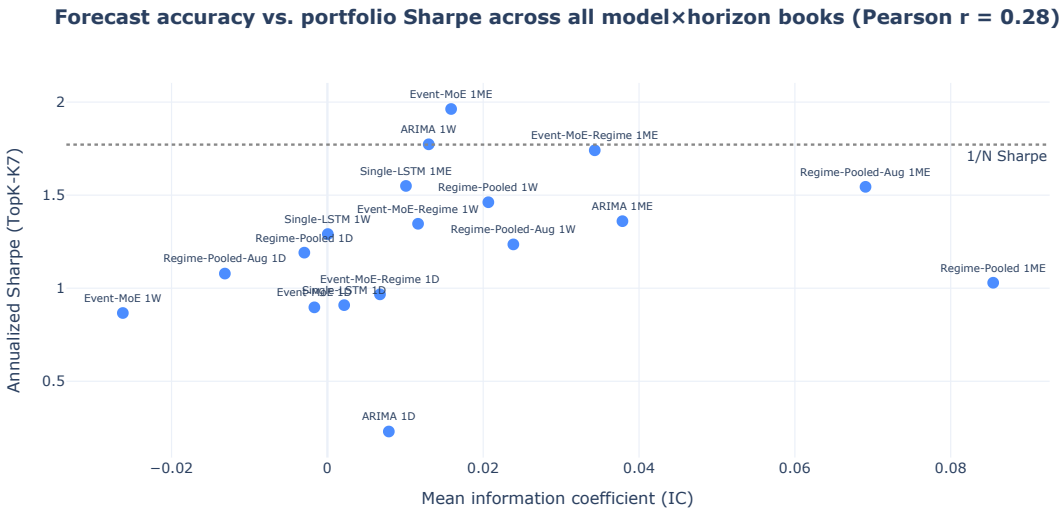

Figure 12. Forecast accuracy versus realized portfolio Sharpe across every model × horizon book (TopK, $K = 7$, 18 points). The relationship is weak (Pearson $r = 0.28$) and not statistically distinguishable from zero at this sample size. This is the compact form of the finding that forecast accuracy and investment value are distinct objectives.

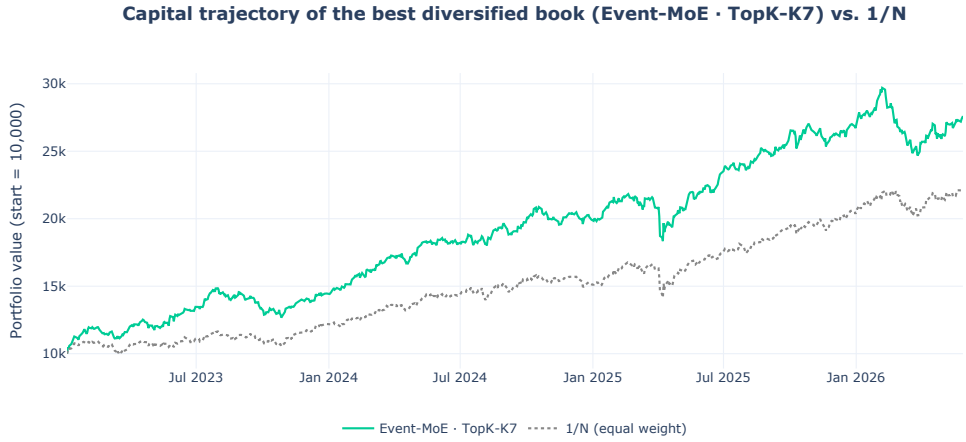

Figure 13. Capital trajectory of the diversified Event-MoE · TopK-7 book against $1/N$, net of cost, from a base of 10,000. It compounds to 27,620 over the test window and tracks the benchmark closely.

Table 9. Top-five books by Sharpe at each horizon (NAIVE excluded, prediction-free books de-duplicated). Turnover is the annualized two-sided traded value relative to capital, which is how many times a year the book replaces its holdings. At the 5-bps cost, transaction fees scale directly with it. The 54× monthly book pays about 2.7% a year in fees, against roughly 0.1% for the 2.8× ARIMA book. The highest Sharpes coincide with the highest turnover.

Book	Sharpe	Ann. ret	Max DD	Turnover
Daily (1D)				
EqualWeight	1.77	30.67%	−18.66%	0.3×
Event-MoE-Regime-Pooled (augmented) · TopK-20	1.47	22.27%	−14.95%	181.2×
Event-MoE-Regime-Pooled · TopKInvVol-9	1.47	27.34%	−19.38%	273.6×
Single-LSTM · TopK-20	1.47	22.94%	−16.09%	41.4×
Event-MoE-Regime-Pooled (augmented) · TopK-19	1.46	22.27%	−14.58%	190.3×
Weekly (1W)				
Single-LSTM · TopKInvVol-3	2.01	44.48%	−25.67%	99.5×
ARIMA · TopK-3	1.86	46.41%	−26.22%	129.2×
ARIMA · TopKInvVol-3	1.85	39.89%	−25.08%	116.4×
ARIMA · TopK-6	1.85	36.77%	−18.57%	73.1×
Event-MoE-Regime · TopK-13	1.85	32.28%	−16.21%	62.1×
Monthly (1ME)				
Event-MoE · TopK-2	**2.31**	61.67%	−20.39%	54.5×
Event-MoE · TopKInvVol-2	2.24	56.53%	−20.12%	50.1×
Event-MoE · TopK-5	2.22	43.21%	−17.78%	40.2×
Event-MoE · TopK-6	2.09	38.36%	−18.02%	34.6×
ARIMA · TopK-3	2.04	50.40%	−28.51%	2.8×

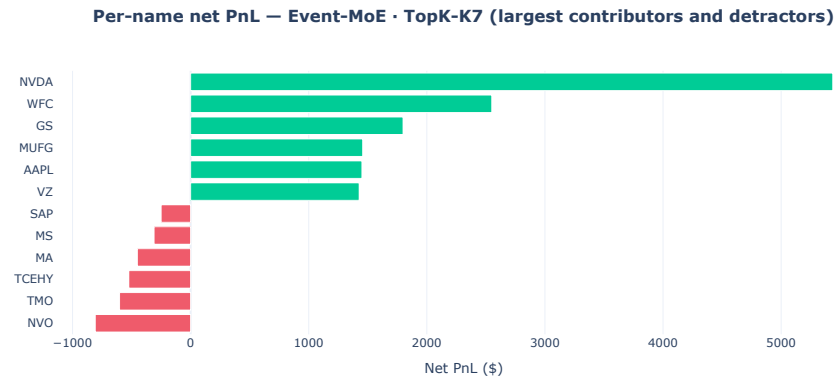

Figure 14. Net per-name PnL for the Event-MoE · TopK-7 book (green contributors, red detractors). A small number of names dominate the book's outcome.

Overall, the results indicate that the strongest contribution comes from cross-sectional pooling. It is strongest and most stable at the monthly horizon, while the event-conditional gate and soft regime blend stay well-behaved and interpretable. The portfolio tests give the second-stage verdict. Forecast

accuracy and investment value are distinct objectives, and the stronger books sit on par with equal weighting after costs.

Table 10. Per-name profit-and-loss for the Event-MoE · TopK-7 book (share-level ledger, net of fees): the six largest contributors and detractors. Book totals: \$27,620 final equity, \$17,620 net PnL, \$520 fees, 465 trades.

Ticker	Realized	Unrealized	Fees	Net
Largest contributors				
NVDA	\$5460	\$0	\$23	\$5437
WFC	\$2582	\$0	\$32	\$2550
GS	\$1829	\$0	\$30	\$1799
MUFG	\$1475	\$0	\$18	\$1457
AAPL	\$743	\$738	\$30	\$1451
VZ	\$1443	\$0	\$15	\$1427
Largest detractors				
SAP	−\$733	\$497	\$13	−\$250
MS	−\$300	\$0	\$9	−\$310
MA	−\$427	\$0	\$24	−\$451
TCEHY	−\$502	\$0	\$22	−\$524
TMO	−\$184	−\$402	\$16	−\$602
NVO	−\$783	\$0	\$24	−\$807

5. Discussion

This section provides our interpretation of the results. We examine what the gate learns, what drives the regime gain, why a good forecast need not give a good portfolio, and where the study is limited.

5.1. What the event-conditional gate learns

The gate was motivated by the concentration of market risk on scheduled announcement dates [4] and the idea that attention capacity should be favored as a release approaches. The results qualify this. The macro-event channels add little monthly, and daily (where timing is most precisely localized), the deep models carry essentially no edge. This fits the economics of scheduled announcements. Their dates are public, so much of the timing effect is likely already in the prices, and a bare calendar (without surprises or consensus) adds little. Free to ignore uninformative channels, the gate largely does so daily. The gate helps, but it does not decide the outcome. It lets the mixture lean on whichever expert suits the state. It is also easy to extend. More detailed event signals, such as surprises and intraday reactions, could be added later. The present narrow calendar does not use them.

5.2. Regime blending and pooling

The two-level design raises an attribution question. How much of the gain comes from the regime-similarity structure, and how much from pooling the per-period experts? Sections 4.2 and 4.5 are decisive. The per-ticker regime model improves on the plain mixture only marginally and is unstable

across years. Pooling trains each period's expert on all forty names' windows at once. It produces both the large monthly-IC jump and the only stable, two-sigma walk-forward edge. The regime layer only helps when each expert has enough data to learn from. A per-period expert restricted to one ticker sees too few windows to estimate well, leaving the soft blend little to combine. Pooling removes that constraint, and the regime structure starts to help. This also justifies broad, overlapping spans over narrow, event-defined ones, which would re-impose the very scarcity pooling relieves. The soft weighting stays valuable for interpretability. The weights read as how strongly the present resembles each past regime. But this pays off only with pooling.

5.3. Why a prediction edge need not become a portfolio edge

The most consequential finding is the disconnect between forecast accuracy and portfolio value, sharpest at the monthly horizon. The most accurate forecaster (Event-MoE-Regime-Pooled, IC 0.086) yields a below-naive book. A far less accurate one (Event-MoE, IC 0.016) yields the best. Three documented mechanisms combine. First, the top-K rule depends only on the ranking of the top names and their realized covariance. So a panel-wide IC can rise even as the very top, the only part the portfolio uses, grows noisier or more correlated. Second, the allocation step amplifies small forecast errors rather than averaging them away, which is why equal weighting is so resilient [2, 5]. Third, turnover costs fall hardest on the books that trade most aggressively [41]. Accuracy and allocation should therefore be evaluated, and optimized, jointly. A decision-focused formulation would back-propagate a risk-adjusted objective through the allocation step. That is the natural response, left to future work [42].

The structure thus works as a forecaster, especially with enough cross-sectional data. That this does not mechanically become a Sharpe improvement is not a weakness but a main finding. Return-forecasting models should be judged on both statistical and economic criteria, with the two stages optimized jointly.

5.4. Limitations

Several limitations bound the strength, not the direction, of these conclusions.

- **Single split.** An expanding-window, retrain-per-fold walk-forward would test stability more severely than the per-year IC used here.
- **Neural variance.** Results use one fixed seed. At this signal scale, seed-ensembling with confidence intervals would firm them up.
- **Multiple testing.** Many per-asset and per-book hypotheses are evaluated, so a false-discovery-rate correction (e.g., Benjamini–Hochberg) should precede any strong significance claim. This matters more because the central Jobson–Korkie result is a non-rejection.
- **Volatility proxy and event scope.** HAR-RV uses a squared-return proxy (no intraday data), and the event signal covers only three U.S. releases by calendar, with no surprise component.
- **Cost robustness.** A transaction-cost sensitivity analysis would help, especially for the higher-turnover daily and weekly books.
- **Set of tickers.** The forty-name universe is screened retrospectively for size and low correlation rather than using point-in-time eligibility criteria. This may introduce selection bias and limits the interpretation of the portfolio results as a fully investable historical strategy.

6. Conclusions

This paper introduced the event-conditional mixture-of-experts model (Event-MoE). It mixes LSTM, GRU, and transformer experts, combined by an event-conditional gate on the scheduled macro-announcement calendar. Three further variants build on it: a regime-similarity blend of per-period experts, a cross-sectionally pooled version, and a four-expert augmented form. The augmented form adds a linear expert, a load-balanced gate, and one-step GARCH and HAR-RV inputs. The family was evaluated end to end on forty decorrelated large-cap stocks over a held-out 2023–2026 window. Evaluation ran from forecast accuracy through six long-only allocation rules, net of a five-basis-point cost. The comparators were NAIVE, ARIMA, and single-LSTM forecasters, plus passive and risk-only benchmarks. Significance was tested with Diebold–Mariano and Jobson–Korkie/Memmel tests, and a year-by-year walk-forward analysis. The main findings are summarized below.

- **Forecast accuracy is not portfolio value.** The most accurate forecaster (Event-MoE-Regime-Pooled, monthly IC 0.086) produced one of the weakest books (Sharpe 1.03). A far less accurate one (Event-MoE, IC 0.016) produced the best (Sharpe 1.96). Across the eighteen books, accuracy and Sharpe correlated only at $r = 0.28$. Forecast quality and investment value are distinct objectives.

- **Cross-sectional pooling is the main driver.** Pooling the per-regime experts raised the monthly IC from 0.034 to 0.086 (rank-IC 0.041 $\rightarrow$ 0.075) and contributed +0.051 of the model ladder. That is more than the event gate (+0.006) and regime layer (+0.018) combined. The gain is one of data efficiency, not a new modeling device.

- **The signal is small and horizon-dependent.** Daily accuracy is near zero (best ARIMA 0.008, deep models flat to negative). The edge emerges at the weekly horizon and is clearest monthly.

- **The event gate helps, but only a little.** The event inputs help only a little, and only up to the monthly horizon. When they carry no signal, the gate mostly ignores them. But the design leaves room to add more detailed event signals later, such as surprises and intraday reactions. The present calendar does not include these yet.

- **Regimes need enough data to train on.** Narrow, event-defined regimes (a dot-com bust, the 2008 crisis, the COVID-19 shock, growth phases) leave each expert with too little data. Their forecasts were worse than those from a few broad, overlapping spans.

- **Every portfolio was profitable. The models kept pace with the naive benchmarks.** Over 2023–2026, all books earned positive net-of-cost returns and positive Sharpe ratios at every horizon. This holds for prediction-driven, passive, and even the constant zero-forecast (NAIVE) book. None lost money. At the monthly horizon, the prediction-driven books clearly beat the naive zero-forecast book (Event-MoE Sharpe 1.96 vs. NAIVE 0.98). They also matched or exceeded $1/N$ (Event-MoE 34.94% return, Sharpe 1.96 vs. $1/N$ 30.67%, Sharpe 1.77). These absolute gains partly reflect a strong equity market over the window.

- **No book beat $1/N$ significantly.** Under the Jobson–Korkie/Memmel test, no prediction-driven book significantly outperformed $1/N$ at any horizon (best monthly Event-MoE $z = 0.38$, $p =$

0.70). Several underperformed it at the daily horizon (ARIMA $p < 0.001$, single-LSTM $p = 0.018$, Event-MoE $p = 0.010$). $1/N$ also beat the MinVariance and MaxSharpe optimizers at every horizon.

- **Concentration buys Sharpe but costs turnover. This is the choice of K.** The highest Sharpe (2.31, annualized 61.67%) came from the most concentrated top-2 Event-MoE book. It turned over about 54× per year, against roughly 2.8× for a diffuse book. Because cost hits active books hardest, the main analysis uses the diversified top-7 rule. The best number of holdings is a cost-adjusted, not accuracy-maximizing, choice.

Summing up, the pipeline is a regime-aware, leakage-controlled forecaster. Its statistical gains are real and concentrated in the cross-sectionally pooled monthly forecasts. But the translation into investment value is not automatic. The allocation stage must be coupled to the forecast rather than run after it.

6.1. Future directions

Several extensions follow naturally. A *graph neural network* over the stocks could let cross-sectional structure inform forecasts, replacing independent per-asset heads with a relational one [44]. Such structure helps only when the portfolio rule can exploit it [5]. A *decision-focused* formulation could train the forecast and the allocation jointly, back-propagating a risk-adjusted objective through a differentiable optimizer rather than treating the two stages separately [42]. A *foundation-model* route offers a different path to the same goal [43]. It would pre-train on large financial corpora and adapt to this set of stocks. The event channel could also be widened. At present, it uses only three scheduled U.S. releases, by calendar alone. A natural next step is to add announcement surprises, consensus data, and intraday reactions. These are the signals the current gate leaves room for but does not yet use.

The evaluation itself can be strengthened in several ways. The test can be run as a walk-forward, retraining on all data up to each step and evaluating on the next slice. Each model can be trained under several random seeds and averaged, with confidence intervals reported. The higher-turnover books can be checked under different transaction-cost assumptions. Finally, because many models and books are tested at once, a multiple-testing correction such as Benjamini–Hochberg can keep significant results from appearing by chance.

Use of AI tools declaration

AI-assisted tools were used exclusively for grammar correction and stylistic editing. All scientific content, methods, experiments, and conclusions were developed by the authors and have been verified by them.

Author contributions

Conceptualization: A.B. and I.B.; methodology, A.B. and I.B.; software, A.B.; validation, A.B. and I.B.; investigation, A.B.; writing–original draft preparation, A.B.; writing–review and editing, A.B. and I.B.; supervision, I.B. All authors have read and agreed to the published version of the manuscript.

Data availability statement

The dataset analyzed in this work is publicly available on GitHub [45] and requests for additional information should be addressed to the corresponding author.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. S. Gu, B. Kelly, D. Xiu, Empirical asset pricing via machine learning, *Rev. Financ. Stud.*, **33** (2020), 2223–2273. https://doi.org/10.1093/rfs/hhaa009
2. V. DeMiguel, L. Garlappi, R. Uppal, Optimal versus naive diversification: How inefficient is the 1/N portfolio strategy, *Rev. Financ. Stud.*, **22** (2009), 1915–1953. https://doi.org/10.1093/rfs/hhm075
3. O. B. Sezer, M. U. Gudelek, A. M. Ozbayoglu, Financial time series forecasting with deep learning: A systematic literature review (2005–2019), *Appl. Soft Comput.*, **90** (2020), 106181. https://doi.org/10.1016/j.asoc.2020.106181
4. M. Halousková, Š. Lyócsa, Forecasting U.S. equity market volatility with attention and sentiment to the economy, preprint, arXiv:2503.19767.
5. R. Wade, Do better volatility forecasts lead to better portfolios? Evidence from graph neural networks, preprint, arXiv:2605.19278.
6. J. Franke, Markov switching time series models, in *Time Series Analysis: Methods and Applications, Handbook of Statistics*, **30** (2012), 99–122. https://doi.org/10.1016/B978-0-444-53858-1.00005-3
7. A. C. Blake, N. A. Gandhi, A. R. Jakkula, Improving S&P 500 volatility forecasting through regime-switching methods, preprint, arXiv:2510.03236.
8. H. Y. Kim, C. H. Won, Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-Type models, *Expert Syst. Appl.*, **103** (2018), 25–37. https://doi.org/10.1016/j.eswa.2018.03.002
9. C. Krauss, X. A. Do, N. Huck, Deep neural networks, gradient-boosted trees, random Forests: Statistical arbitrage on the S&P 500, *Eur. J. Oper. Res.*, **259** (2017), 689–702. https://doi.org/10.1016/j.ejor.2016.10.031
10. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9** (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
11. K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, et al., Learning phrase representations using RNN encoder–decoder for statistical machine translation, in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (2014), 1724–1734. https://doi.org/10.3115/v1/D14-1179

12. K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, J. Schmidhuber, LSTM: A search space odyssey, *IEEE Trans. Neural Netw. Learn. Syst.*, **28** (2016), 2222–2232. https://doi.org/10.1109/TNNLS.2016.2582924

13. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need, in *Neural Information Processing Systems (NIPS 2017)*, **30** (2017). https://doi.org/10.48550/arXiv.1706.03762

14. C. Wang, Y. Chen, S. Zhang, Q. Zhang, Stock market index prediction using deep transformer model, *Expert Syst. Appl.*, **208** (2022), 118128. https://doi.org/10.1016/j.eswa.2022.118128

15. Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, et al., Transformers in time series: A survey, in *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI-23)*, (2023), 6778–6786. https://doi.org/10.24963/ijcai.2023/759

16. R. A. Jacobs, M. I. Jordan, S. J. Nowlan, G. E. Hinton, Adaptive mixtures of local experts, *Neural Comput.*, **3** (1991), 79–87. https://doi.org/10.1162/neco.1991.3.1.79

17. M. I. Jordan, R. A. Jacobs, Hierarchical mixtures of experts and the EM algorithm, *Neural Comput.*, **6** (1994), 181–214. https://doi.org/10.1162/neco.1994.6.2.181

18. A. J. Zeevi, R. Meir, R. J. Adler, Time series prediction using mixtures of experts, in *Advances in Neural Information Processing Systems (NIPS)*, **9** (1997).

19. R. Ebrahimpour, H. Nikoo, S. Masoudnia, M. R. Yousefi, M. S. Ghaemi, Mixture of MLP-experts for trend forecasting of time series: A case study of the Tehran stock exchange, *Int. J. Forecast.*, **27** (2011), 804–816. https://doi.org/10.1016/j.ijforecast.2010.02.015

20. D. Vallarino, Adaptive market intelligence: A mixture of experts framework for volatility-sensitive stock forecasting, preprint, arXiv:2508.02686.

21. X. Du, J. Cao, X. Jiang, Q. Wang, B. Xu, Z. Liu, et al., Integrating event information and multi-dimensional relationships for improved financial time series forecasting, *Sci. Rep.*, **15** (2025), 38099. https://doi.org/10.1038/s41598-025-22926-y

22. C. Yu, J. M. Mulvey, P. N. Kolm, Deep statistical jump models: Towards a general mixture-of-experts framework for time series learning, *SSRN Electr. J.*, **2025** (2025). https://doi.org/10.2139/ssrn.5817083

23. X. Shi, S. Wang, Y. Nie, D. Li, Z. Ye, Q. Wen, et al., TIME-MOE: Billion-scale time series foundation models with mixture of experts, in *13th International Conference on Learning Representations (ICLR 2025)*, (2025), 34635–34667.

24. M. Segnon, R. Gupta, B. Wilfling, Forecasting stock market volatility with regime-switching GARCH-MIDAS: The role of geopolitical risks, *Int. J. Forecast.*, **40** (2024), 29–43. https://doi.org/10.1016/j.ijforecast.2022.11.007

25. C. W. S. Chen, C. T. H. Chien, Improving quantile forecasts via realized double hysteretic GARCH model in stock markets, *Comput. Econ.*, **64** (2024), 3447–3471. https://doi.org/10.1007/s10614-024-10563-y

26. A. Pagliaro, Regime-aware LightGBM for stock market forecasting: A validated walk-forward framework with statistical rigor and explainable AI, *Electronics*, **15** (2026), 1334. https://doi.org/10.3390/electronics15061334

27. T. Bollerslev, Generalized autoregressive conditional heteroskedasticity, *J. Econom.*, **31** (1986), 307–327. https://doi.org/10.1016/0304-4076(86)90063-1

28. A. García-Medina, E. Aguayo-Moreno, LSTM-GARCH hybrid model for the prediction of volatility in cryptocurrency portfolios, *Comput. Econ.*, **63** (2024), 1511–1542. https://doi.org/10.1007/s10614-023-10373-8

29. G. Taneva-Angelova, D. Granchev, Deep learning and transformer architectures for volatility forecasting: Evidence from U.S. equity indices, *J. Risk Financ. Manag.*, **18** (2025), 685. https://doi.org/10.3390/jrfm18120685

30. M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, P. N. Suganthan, Ensemble deep learning: A review, *Eng. Appl. Artif. Intell.*, **115** (2022), 105151. https://doi.org/10.1016/j.engappai.2022.105151

31. W. Chen, Z. Liu, L. Jia, A hybrid approach for portfolio construction: Combining two-stage ensemble forecasting model with portfolio optimization, *Comput. Intell.*, **40** (2024), e12617. https://doi.org/10.1111/coin.12617

32. A. Bielskis, I. Belovas, Machine learning models and ensemble methods for stock index return forecasting, *Netw. Heterog. Media*, **21** (2026), 848–864. https://doi.org/10.3934/nhm.2026035

33. O. Ledoit, M. Wolf, Honey, I shrunk the sample covariance matrix, *J. Portf. Manag.*, **30** (2004), 110–119. https://doi.org/10.3905/jpm.2004.110

34. O. Ledoit, M. Wolf, A well-conditioned estimator for large-dimensional covariance matrices, *J. Multivar. Anal.*, **88** (2004), 365–411. https://doi.org/10.1016/S0047-259X(03)00096-4

35. O. Ledoit, S. Péché, Eigenvectors of some large sample covariance matrix ensembles, *Probab. Theory Relat. Fields*, **151** (2011), 233–264. https://doi.org/10.1007/s00440-010-0298-3

36. O. Ledoit, M. Wolf, Analytical nonlinear shrinkage of large-dimensional covariance matrices, *Ann. Stat.*, **48** (2020), 3043–3065. https://doi.org/10.1214/19-AOS1921

37. A. García-Medina, High-dimensional covariance matrix estimators on simulated portfolios with complex structures, *Phys. Rev. E*, **111** (2025), 024316. https://doi.org/10.1103/PhysRevE.111.024316

38. W. F. Sharpe, Capital asset prices: A theory of market equilibrium under conditions of risk, *J. Finance*, **19** (1964), 425–442. https://doi.org/10.1111/j.1540-6261.1964.tb02865.x

39. E. F. Fama, K. R. French, Common risk factors in the returns on stocks and bonds, *J. Financ. Econ.*, **33** (1993), 3–56. https://doi.org/10.1016/0304-405X(93)90023-5

40. A. Zaimovic, A. Omanovic, A. Arnaut-Berilo, How many stocks are sufficient for equity portfolio diversification? A review of the literature, *J. Risk Financ. Manag.*, **14** (2021), 551. https://doi.org/10.3390/jrfm14110551

41. K. Kashif, R. Ślepaczuk, Deep reinforcement learning framework for diversified portfolio management across global equity markets, preprint, arXiv:2605.17307.

42. Y. Hwang, S. Zohren, Signature-informed transformer for asset allocation, preprint, arXiv:2510.03129.

43. Z. Zhu, H. Chen, Q. Qu, V. Chung, FinCast: A foundation model for financial time-series forecasting, in *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM 2025)*, (2025), 4539–4549. https://doi.org/10.1145/3746252.3761261
44. Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, P. S. Yu, A comprehensive survey on graph neural networks, *IEEE Trans. Neural Netw. Learn. Syst.*, **32** (2020), 4–24. https://doi.org/10.1109/TNNLS.2020.2978386
45. A. Bielskis, *Machine Learning Financial Dataset Repository*. Available from: https://github.com/AivarasBi/MOE.
46. N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, et al., Outrageously large neural networks: The sparsely-gated mixture-of-experts layer, in *International Conference on Learning Representations (ICLR)*, (2017), 1–19.
47. W. Fedus, B. Zoph, N. Shazeer, Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, *J. Mach. Learn. Res.*, **23** (2022), 1–39.
48. F. Corsi, A simple approximate long-memory model of realized volatility, *J. Financ. Econom.*, **7** (2009), 174–196. https://doi.org/10.1093/jjfinec/nbp001
49. H. Markowitz, Portfolio selection, *J. Finance*, **7** (1952), 77–91. https://doi.org/10.1111/j.1540-6261.1952.tb01525.x
50. W. F. Sharpe, The sharpe ratio, *J. Portf. Manag.*, **21** (1994), 49–58. https://doi.org/10.3905/jpm.1994.409501
51. F. X. Diebold, R. S. Mariano, Comparing predictive accuracy, *J. Bus. Econ. Stat.*, **13** (1995), 253–263. https://doi.org/10.1080/07350015.1995.10524599
52. D. Harvey, S. Leybourne, P. Newbold, Testing the equality of prediction mean squared errors, *Int. J. Forecast.*, **13** (1997), 281–291. https://doi.org/10.1016/S0169-2070(96)00719-4
53. J. D. Jobson, B. M. Korkie, Performance hypothesis testing with the Sharpe and Treynor measures, *J. Finance*, **36** (1981), 889–908. https://doi.org/10.1111/j.1540-6261.1981.tb04891.x
54. C. Memmel, Performance hypothesis testing with the Sharpe ratio, *Finance Lett.*, **1** (2003), 21–23.



AIMS Press

© 2026 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0)