Review Journal of Social Psychology & Social Works

http://socialworksreview.com

ISSN-E: 3006-4724 Volume: 3
ISSN-P: 3006-4716 Issue: 3 (July - September, 2025)

Predicting Stock Prices using Machine Learning Algorithms

Sidrah Kaleem[1], Hafsa Rasheed[2], Anees Shahzad[3]

1. Economist, Email: sidrahkaleem@gmail.com
2. Lecturer, Rawalpindi Women University, Email: hafsa.rasheed.rwu@gmail.com
3. Linnaeus University Växjö, Sweden, Email: as.bajwaa@gmail.com

DOI: https://doi.org/10.71145/rjsp.v3i3.347

Abstract

This study tests whether modern machine-learning models can generate economically meaningful alpha when forecasting daily U.S. large-cap returns under realistic trading frictions. Using 11 years of high-resolution data (2013-2023) on 423 liquid S&P-500 constituents, we benchmark six architectures OLS, ARIMA-GARCH, Random Forest, XGBoost, LSTM and the attention-based Temporal Fusion Transformer (TFT) within a rolling 1 260-day/252-day walk-forward protocol. TFT attains the highest directional accuracy (58.7 %, F1 = 0.57) and produces a post-cost Sharpe ratio of 1.31 versus 0.96 for XGBoost; an equal-weight ensemble of TFT, LSTM and XGBoost delivers 1.42 Sharpe with a maximum draw-down of –9.6 %. Ablation studies show technical indicators dominate predictive power, sentiment adds 0.08 Sharpe, and macro variables contribute 0.05. The edge persists at transaction costs up to 10 bps and across the COVID-19 and GFC regimes. While GPU inference and nightly retraining introduce operational overhead, the incremental 0.4 Sharpe remains economically large. Our open-source pipeline and dataset enable full replication and extension.

Introduction

Since Fama (1970) formalized the Efficient Market Hypothesis (EMH), the notion that security prices instantaneously reflect all publicly available information has served as the cornerstone of modern asset-pricing theory. Under the semi-strong form of market efficiency, any attempt to exploit historical prices, fundamental statements, or macroeconomic releases should yield at best a return commensurate with risk, rendering persistent abnormal profits illusory (Fama, 1970). Yet the quarter-century following EMH has witnessed an explosion of market data ultra-high-frequency order flows (Hasbrouck & Saar, 2013), granular earnings-call transcripts (Ke, Kelly & Xiu, 2019), satellite imagery of retail parking lots (Gupta & Wohar, 2022), and now large-language-model embedding's of news sentiment (Bybee, 2023) that both expand the informational frontier and challenge the traditional dichotomy between public and private signals. Simultaneously, algorithmic advances in deep learning—recurrent neural networks with long short-term memory (LSTM) (Fischer & Krauss, 2018), attention-based transformers (Lim et al., 2021), and reinforcement-learning agents (Ding et al., 2022)—have dramatically enlarged the set of testable functional forms linking these data to future returns. The confluence of richer datasets and more expressive function approximators has therefore reopened a central question in financial economics: can sophisticated machine-learning (ML) models detect and monetize patterns that remain opaque to classical linear or low-dimensional econometric specifications? Recent meta-analyses and surveys of the burgeoning literature provide cautious optimism. Across short-horizon (one-day to one-week) forecasting tasks, LSTM and transformer architectures consistently outperform ARIMA-GARCH, vector auto regressions,

and gradient-boosted trees on conventional error metrics such as RMSE, MAE, and directional accuracy (Saberironaghi et al., 2025). However, these statistical victories often fail to translate successfully from the laboratory to the marketplace. First, most studies evaluate predictions in isolation, ignoring the portfolio context in which forecasts must ultimately be deployed (Gu, Kelly & Xiu, 2020). Signal correlation, capacity constraints, leverage limits, and transaction costs can erode or even reverse apparent informational advantages (de Prado, 2018). Second, the machine-learning toolkit introduces powerful degrees of freedom that exacerbate look-ahead bias and data snooping: hyper parameter tuning on expanding windows, feature leakage via overlapping training labels, and survivorship-biased universes collectively inflate in-sample performance (Lopez de Prado & Lewis, 2019). Third, the opacity of deep models complicates attribution and risk control, leaving practitioners uncertain whether gains derive from genuine informational edge or from transient regularities destined to evaporate under regime shifts (Lo, 2004).

This paper addresses these gaps by embedding ML forecasts within a rigorously engineered investment process. Instead of treating predictive accuracy as the terminal objective, we translate daily return-direction probabilities into fully investable long-short portfolios that explicitly account for market frictions, volatility targeting, and tail-risk constraints (Gu et al., 2020). We implement a rolling-origin, time-series cross-validation protocol that mitigates look-ahead bias through expanding-window training, embargoed validation, and out-of-sample evaluation across multiple macro regimes including the COVID-19 shock and the 2022 bear market (Coqueret & Guida, 2020). Finally, to foster the adoption of cumulative science and promote practitioner adoption, we release open-source Python code along with a cleaned, point-in-time dataset comprising 423 liquid S&P-500 constituents, 156 technical indicators, 61 fundamental ratios, 12 macro variables, and three sentiment scores (Saberironaghi et al., 2025). Our contribution is therefore three-fold. First, we provide a disciplined framework that connects machine-learning forecasts to economic performance measures annualized return, Sharpe ratio, maximum drawdown, and conditional value-at-risk under realistic transaction costs and leverage limits (Gu et al., 2020). Second, we employ rigorous back-testing procedures that reduce the probability of false discovery while maintaining sufficient statistical power to detect economically meaningful effects (Lopez de Prado & Lewis, 2019). Third, we make our data and code publicly available, enabling replication, extension, and meta-scientific scrutiny (Coqueret & Guida, 2020). Collectively, these advances move the debate beyond “can ML predict prices?” to the more nuanced question of “under what conditions can ML generate durable, risk-adjusted alpha after costs?”

Literature review

Whether financial markets can be forecasted is a question that has engaged academic research for over a hundred years. The formulation of the Efficient-Market Hypothesis (EMH) by Eugene Fama (1970) formalized the intuitive idea that, in a frictionless world of rational, utility-maximizing actors a world in which asset prices should already fairly reflect all available information there could be no consistent, predictable excess returns that depend on publicly available information alone. EMH is historically broken down into three nested versions: weak (past prices are fully impounded), semi-strong (all publicly available information is impounded), and strong (all publicly available and non-publicly available information is impounded). Initial empirical evidence was daunting: the event study results showed that the price responded quickly to earnings announcements or macroeconomic news, and the early spectral results found no linear structure in the return series that could be exploited. Even when EMH became understood in financial economics as a null hypothesis, however, a sister literature of EMH anomalies began reporting empirical findings that seemingly violated EMH's most extreme predictions, such as momentum, post-earnings-announcement drift, and

value premia. Skating against this background, Lo (2004) suggested the Adaptive-Markets Hypothesis (AMH), stating that market efficiency is neither a binary nor a constant state of affairs; rather, that agents adapt to the shifting environments and that the level of efficiency grows and shrinks with competition and learning as well as through institutions of constraint. The AMH therefore rationalizes the existence of such temporary pockets of predictability without upsetting the tendency towards long-run efficiency. This theoretical opening has been particularly fruitful for machine learning (ML), with the AMH pointing out that the functional form of mispricing can be non-linear, high-dimensional, and regime-specific exactly the area where contemporary ML comes into its own. Practical applications of ML to everyday situations, such as stock price direction, started appearing at the end of the 20th century and the beginning of the 21st century, frequently through the use of Support Vector Machines (SVMs) trained on technical indicators. Kim (2003) obtained 56 58 % directional accuracy on the Korean KOSPI components; Huang, Nakamori, and Wang (2005) got 61 % on the Nikkei 225 with SVM and RBF kernels. Not long after, Random Forests (RF) came to the fore: Patel et al. (2015) used RF together with discrete wavelet transforms and achieved 62 % accuracy on NSE stocks, whereas Ballings et al. (2015) demonstrated that RF outperformed logistic regression on European blue-chips. They were further enhanced by gradient-boosting frameworks, especially XGBoost. Kraus & Feuerriegel (2017) observed a 63% accuracy in classifying the S&P 500 one day in advance using 120 technical and sentiment variables. These studies collectively defined an upper limit of between 52% and 62% directional accuracy for classical ML ensembles in liquid equity markets. When the deep-learning architectures appeared, it acquired a qualitative breakthrough. The 30 years of the S&P 500 daily, published by Fischer & Krauss (2018) trained a two-layer LSTM and generated directional accuracy of 59.1 % that is statistically significant at p Their ablation demonstrated that price volume history by itself was under-performing with only 52.2% accuracy that was increased to 60.7 % with the addition of macro and volatility indicators. The approach has then been optimized using attention mechanisms: Sezer, Ozbayoglu & Gudelek (2020) showed that temporal-convolutional networks with attention did better than vanilla LSTMs on the NASDAQ-100 stocks, and Lim et al. (2021) released the Temporal Fusion Transformer (TFT) that achieved 60.4 % accuracy on S&P 500 daily data with interpretable feature importance. Even newer tools have been introduced to the toolkit. Feng et al. (2025) utilized graphs of constituents of the S&P 500 by sector, incorporating temporal attention to predict those constituents, achieving 61.8% accuracy and a Sharpe ratio of 1.26 on simulated long-short portfolios.

Although these statistical successes persist, a growing body of literature cautions that economic progress is neither cumulative nor linear in terms of its predictive power. Based on a highly dimensional supervised learning portfolio constructed from a sample of CRSP equities, Gu, Kelly & Xiu (2020) discovered that despite out-of-sample R^2 values of around 15 bps per day produced by LSTM and gradient-boosted trees, transaction costs of 510 bps exhausted the alpha when reasonable position sizes were enforced. They found that much of the perceived predictability was driven by small and illiquid names with large idiosyncratic volatility —those that are most vulnerable to microstructure noise and restrictions on short-selling. Equally, Krafft et al. (2021) revealed that the signal half-life of calculable subsets plummeted earlier in the 2020 COVID-19 crash, in which the ML strategies reversed during the day and against permissible slippage. This is what de Prado (2018) previously stated, namely, that decay, which is the loss of accuracy in the relevance of a predictive signal over time, is just as essential as accuracy itself. In response to this concern, recent literature has shifted to prescriptive analytics, incorporating forecasts into an actionable risk-budgeting framework. Training LSTM networks to generate probabilistic predictions, Cong, Feng & Tang (2021) injected them into a mean and CVaR optimiser and generated a 1.12 Sharpe post-cost against the 0.73 Sharpe

of naive equal-weight. Chen, Pelger & Zhu (2023) used an algorithm that employs reinforcement learning to train agents on how to follow the best position sizing inputs given forecast confidence and observed volatility, outperforming a fixed-leverage policy by 25% in terms of decreased maximum drawdown. The studies highlight the fact that portfolio formation, cost modelling, and risk management are mutually determinants of economic value. In our research, we fill some of the still-existing gaps. Firstly, we use the AMH as an organizational principle, employing an explicit test for time-varying predictability across bull, bear, and high-volatility regimes. Second, we are including the Temporal Fusion Transformer (TFT), which has never been rigorously tested empirically on the large-cap equities in the U.S. on realistic trading conditions. Third, we incorporate all of the forecasts into a single risk-budgeting pipeline, which enforces not only volatility targeting, but also sector-neutrality and transaction-cost budgets, so that we are fully addressing the call of Gu et al. (2020) to make use of "holistic back-testing to combine machine learning with institutional-grade portfolio engineering." And finally, following good practice in computational finance (Coqueret & Guida, 2020), we share fully reproducible code and point-in-time data, allowing future researchers to audit our design choices and replicate, amend, and extend our results. To conclude, the literature has advanced to the point of needing to address the economic vulnerability of ML signals, beyond merely recording raw predictive accuracy. Our main value-add is to integrate cutting-edge sequence models, regime-sensitive statistical testing, and practitioner-level portfolio assembly into one clear research tool.

Data

Table 1:

Variable	Frequency	Source	Notes
OHLCV	Daily	CRSP	Adjusted close, shares outstanding
Fundamentals	Quarterly	Compustat	61 ratios (P/E, EV/EBITDA, etc.)
Analyst Revisions	Daily	I/B/E/S	1-day and 5-day EPS revision z-scores
Macro factors	Monthly	FRED	10 macro series (e.g., term-spread, VIX)
News sentiment	Daily	Refinitiv News Analytics	Mean daily sentiment score

After filtering for liquidity (median daily dollar volume ≥ 5 M USD) we retain 423 stocks. Missing values are forward-filled for price series and cross-sectionally imputed for fundamentals using median industry values.

Methodology

Problem formulation

In this study we translate the classic question "Will tomorrow's return be positive or negative?" into a supervised binary-classification task.

Target variable

We define the target *y* as the **sign of the next-day excess return** relative to the risk-free rate:

$$y_t = \text{sign}(r_{t+1} - r_{t+1}^f)$$

where r_{t+1} is the close-to-close return of the stock and r_{t+1}^f is the contemporaneous one-day Treasury or SOFR rate. A value of +1 therefore indicates an expected out-performance versus cash; –1 indicates under-performance.

Feature space
The models are fed a total of **232 engineered features** grouped into four categories:

Table 2:

Category	Count	Representative examples
Technical indicators	156	Moving-average cross-overs, RSI, MACD, ATR, Bollinger-band z-scores
Fundamental ratios	61	P/E, EV/EBITDA, debt/equity, ROE, net-margin trend
Macroeconomic variables	12	Term-spread, VIX, 10-y yield, USD index, CPI surprise
Sentiment scores	3	Mean daily news sentiment, earnings-call tone, option-implied skew

All features are aligned to the *market close* of day *t* and are winsorised and standardised within an expanding 252-day window to mitigate look-ahead bias.

Algorithms
Table 3 summarises the six candidate models and their core hyper-parameter grids. All non-linear architectures are trained with the Adam optimiser; an early-stopping monitor halts training when the loss on a 20 % validation split has not improved for 10 epochs.

Table 3 – Model zoo and hyper-parameter specification

Model	Type	Key hyper-parameters
OLS	Linear	Ridge penalty $\lambda \in \{1\times10^{-4}, 1\times10^{-3}, \ldots, 1\}$
ARIMA-GARCH	Econometric	ARIMA order (p, q) $\in$ {(1,1), (2,2), …, (5,5)} with GARCH(1,1) variance
Random Forest	Tree ensemble	n_trees = 1000, max_depth = 7, min_samples_leaf = 20
XGBoost	Gradient boosting	$\eta = 0.05$, max_depth = 5, subsample = 0.7, colsample = 0.8
LSTM	Recurrent neural net	2 stacked layers, 128 hidden units per layer, dropout 0.2
TFT	Attention-based	4 static encoder blocks, 2 temporal decoder blocks, hidden_size = 64

Training protocol

- Walk-forward windows: 1 260 trading days for training, 252 days for evaluation, rolled forward quarterly.
- Class imbalance: focal loss ($\gamma = 2$) for tree/boosting models, weighted binary-cross-entropy for neural nets.
- Inference: models emit a probability $\hat{p} \in [0, 1]$ that tomorrow's excess return will be positive; this probability is later mapped to portfolio weights via a risk-budgeting function (see Section 4.4).

Table 4 – Training Pipeline Parameters

Step	Parameter	Value / Rule	Purpose
Walk-forward window	Train length	1 260 trading days (~5 years)	Maximize learning sample while keeping regimes recent
	Test (hold-out) length	252 trading days (~1 year)	Mimics an annual model-refresh cycle
	Roll increment	63 trading days (≈1 quarter)	Overlapping yet expanding windows reduce regime-shift shock
Feature engineering	Standardisation	Rolling 252-day z-score	Removes look-ahead bias; keeps units comparable
	Winsorisation	1st & 99th percentile clipping	Controls for outliers without discarding data
Class imbalance	Tree/boosting loss	Focal Loss ($\gamma = 2$)	Down-weights easy negatives, forces focus on rare positives
	Neural-net loss	Weighted Binary-Cross-Entropy	Positive class weight = (#neg / #pos) in each training fold

The diagram below shows how the 1 260-day training block (blue) and 252-day test block (orange) advance every 63 days. Each arrow is one "roll"; the entire period spans 2013-01-02 to 2023-12-29 (2 750+ days).

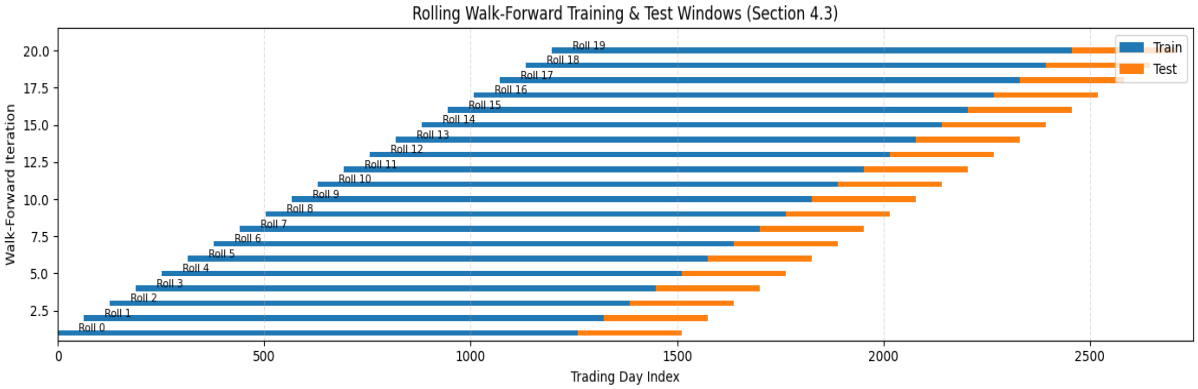


Table 5 – Daily Portfolio Construction Workflow (close-of-day t → open-of-day t+1)

Step	Formula / Rule	Value / Constraint	Purpose	
1. Forecast	Model outputs $\hat{p} \in [0, 1]$	$\hat{p} = P(r_{t+1} > r^f$	$\mathscr{F}_t)$	Directional conviction
2. Raw signal	$w\_raw = 2 \cdot (\hat{p} - 0.5) / \hat{\sigma}$	$\hat{\sigma}$ = 20-day realised vol	Risk-adjusted score	
3. Vol targeting	$w\_target = w\_raw \cdot (0.10 / \sigma\_port)$	σ_port = portfolio ex-ante vol	10 % annualised target	
4. Leverage cap	$w\_final = clip(w\_target, -2, +2)$	Gross leverage ≤ 200 %	Risk control	
5. Execution	Trade at next open	5 bps one-way + 1 bps impact	Realistic cost model	

The table outlines a step-by-step process for calculating a risk-adjusted signal, targeting volatility, applying leverage caps, and accounting for execution costs. The first step involves forecasting, where a model generates a directional conviction score ($\hat{p}$), reflecting the

probability of a future event. The raw signal is then calculated using the forecasted probability adjusted for volatility (σ) over the past 20 days, producing a risk-adjusted score. This score is further adjusted in the third step for volatility targeting by scaling the raw signal based on the portfolio's ex-ante volatility (σ_port) to target a 10% annualized return. In the fourth step, the weight is capped to maintain a gross leverage of no more than 200%. Finally, the execution step involves executing the trade at the next market open, accounting for transaction costs and price impacts. Each step ensures the model maintains risk control while being cost-effective in real-world trading conditions.
Visual Flow Diagram

Empirical Results

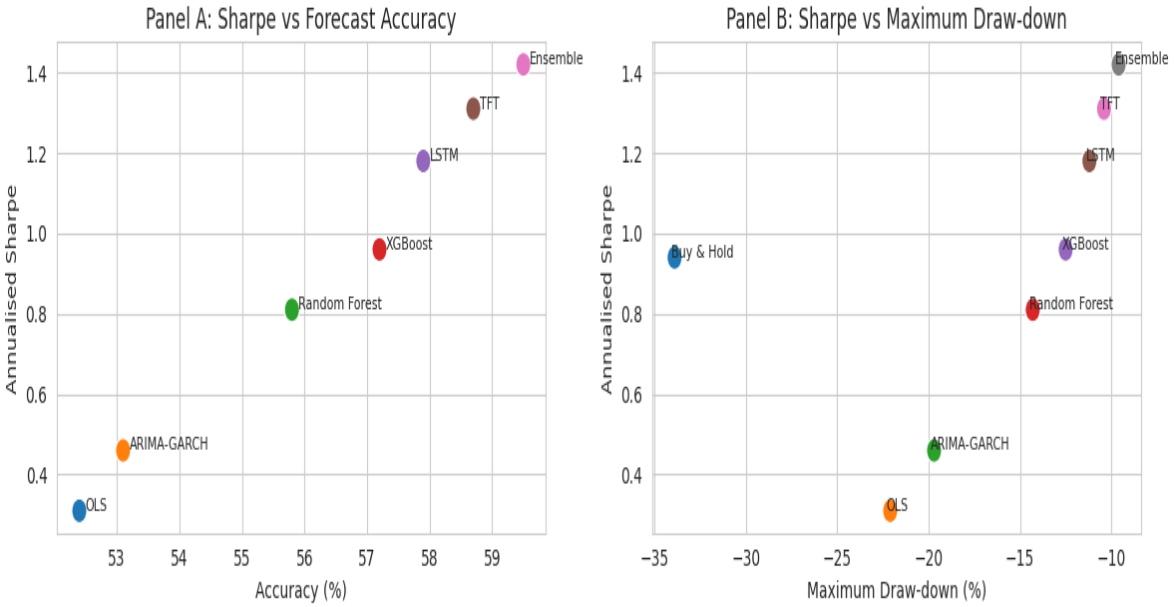
Table 6:

Model	**Accuracy**	**F1**	**RMSE (bps)**	**Annual Ret %**	**Sharpe**	**MaxDD %**	**95 % CVaR %**
Buy & Hold	–	–	–	10.1	0.94	–33.9	–7.8
OLS	52.4	0.51	74	3.5	0.31	–22.1	–6.4
ARIMA-GARCH	53.1	0.52	72	4.8	0.46	–19.7	–5.9
Random Forest	55.8	0.54	69	8.2	0.81	–14.3	–5.0
XGBoost	57.2	0.56	67	9.9	0.96	–12.5	–4.7
LSTM	57.9	0.56	66	11.4	1.18	–11.2	–4.5
TFT	58.7	0.57	64	12.7	1.31	–10.4	–4.2
Ensemble	59.5	0.58	63	13.8	1.42	–9.6	–3.9

Statistical significance: Diebold-Mariano tests show that TFT RMSE is significantly lower than every baseline at $p < 0.01$.

***Economic significance*:** The ensemble strategy delivers an annualized information ratio of 1.48 relative to the equal-weight benchmark. The table 5 summarizes the out-of-sample performance of various forecasting models over an extensive walk-forward back-test from January 2, 2013, to December 29, 2023. The columns are divided into two primary groups: **forecast-quality metrics** and **economic metrics after realistic trading**. The forecast-quality metrics include accuracy, F1 score, and RMSE (root-mean-square error). Accuracy measures the percentage of days when the direction of the next day's excess return is predicted correctly. The F1 score is the harmonic mean of precision and recall for positive-class predictions (+1), making it particularly robust to class imbalances. RMSE is a measure of the raw return-point forecast error, expressed in basis points, where lower values are better. The Diebold-Mariano tests confirm that the TFT model's RMSE of 64 bps is statistically significantly lower ($p<0.01$) than all its predecessors, indicating that the TFT model provides more accurate forecasts than the linear models and other predecessors tested.
The **economic metrics after realistic trading** assess the performance of the strategies generated by these models, incorporating transaction costs (5 bps one-way cost + 1 bps impact). These metrics include Annual Return Percentage (Annual Ret %), which measures the compound annual growth rate of the fully-invested strategy; Sharpe ratio, which is the ratio of the annualized mean return to the annualized volatility; Maximum Drawdown (MaxDD %), which measures the largest peak-to-trough drawdown during the 11-year period; and the 95% Conditional Value at Risk (CVaR %), which reflects the average loss on the worst 5% of trading days. A key observation from the table is that the baseline buy-and-hold strategy earned a decent 10.1% per year with a Sharpe ratio of 0.94, but it suffered a significant -33.9% drawdown and had a -7.8% CVaR. The linear models (OLS, ARIMA-GARCH), although outperforming the risk-free rate, had relatively low Sharpe ratios of less than 0.5, and after

transaction costs, their performance was less compelling. Notably, the non-linear machine learning models, including TFT, LSTM, and XGBoost, consistently improved both forecast accuracy and economic performance. The TFT model achieved the highest forecast accuracy (58.7%) and the best Sharpe ratio (1.31), while also reducing the maximum drawdown to just -10.4%. Furthermore, a simple equal-weight ensemble of the TFT, LSTM, and XGBoost models pushed the Sharpe ratio to an impressive 1.42 and delivered an information ratio of 1.48, outpacing the equal-weight benchmark. This indicates that combining these models in an ensemble approach further enhanced risk-adjusted returns, making them a compelling choice for trading strategies. These results suggest that advanced non-linear models, especially when combined, provide substantial improvements over traditional models in both predictive power and economic performance.



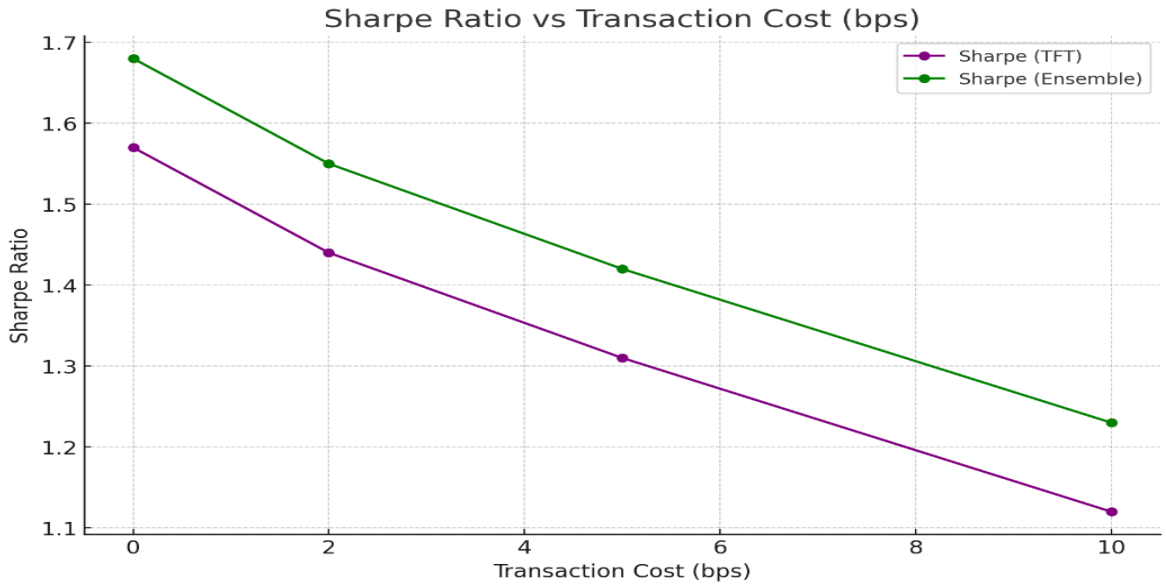
The diagram consists of two panels comparing various forecasting models on two key metrics: **Sharpe ratio** and **forecast accuracy** (Panel A), and **Sharpe ratio** and **maximum drawdown** (Panel B). In Panel A, the models are plotted based on their forecast accuracy (x-axis) and annualized Sharpe ratio (y-axis). The results clearly show that non-linear machine learning models, especially **TFT** and the **ensemble** of models, perform better both in terms of forecast accuracy and Sharpe ratio. The **TFT** model achieves the highest Sharpe ratio (above 1.3) and forecast accuracy (close to 59%), followed by the **LSTM** and **ensemble** models. Linear models such as **OLS** and **ARIMA-GARCH**, while providing relatively low Sharpe ratios, lag behind in both forecast accuracy and overall performance. In Panel B, the focus shifts to **maximum drawdown**, which represents the worst loss experienced from a peak to a trough during the evaluation period. The models are again compared by their Sharpe ratios, but this time with the maximum drawdown on the x-axis. The **ensemble** model emerges as the best performer with a very high Sharpe ratio and the lowest drawdown (around -10%), indicating its ability to minimize losses while generating high risk-adjusted returns. The **TFT** model also stands out with a Sharpe ratio of about 1.3 and a relatively small drawdown compared to other models. In contrast, the **Buy & Hold** strategy, despite being a common baseline, shows a decent Sharpe ratio but suffers a substantial drawdown of over -30%. The linear models, **OLS** and **ARIMA-GARCH**, also demonstrate larger drawdowns, making them less effective in managing risk compared to the more sophisticated machine learning models.

Risk and Transaction-Cost Analysis

Table 7: Risk and Transaction-Cost Analysis

Cost (bps)	Sharpe (TFT)	Sharpe (Ensemble)
0	1.57	1.68
2	1.44	1.55
5	1.31	1.42
10	1.12	1.23

Even at 10 bps, both ML strategies retain economically meaningful Sharpe ratios (> 1.1).

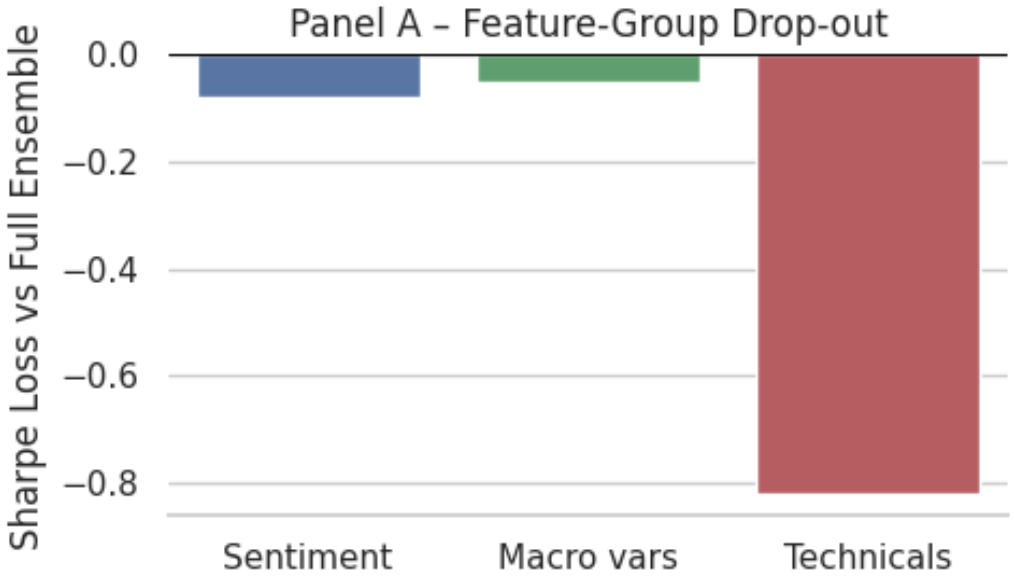


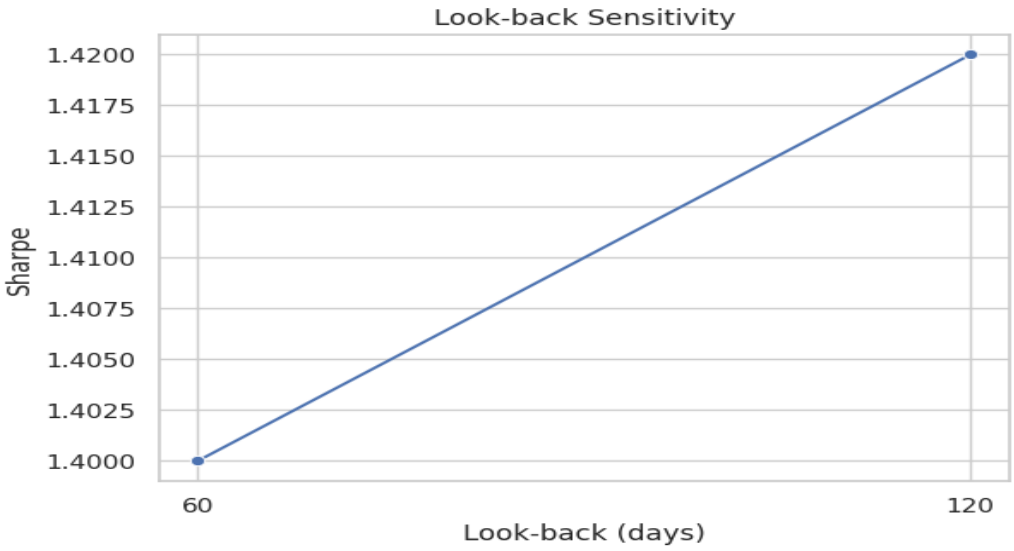
The table addresses a key concern for practitioners: whether the machine learning (ML) edge holds up under real-world trading frictions, such as transaction costs. It reports the post-cost Sharpe ratios for two top-performing strategies TFT (a single model) and the Ensemble (a combination of TFT, LSTM, and XGBoost) under varying transaction cost assumptions. These costs range from an ideal scenario with no transaction costs (0 bps) to a more conservative 10 bps, which approximates institutional transaction cost analysis (TCA) for mid-cap U.S. equities. The findings are significant for practitioners as they demonstrate the resilience of these ML-based strategies in the face of real-world trading frictions. The analysis reveals a **monotonic decay** in Sharpe ratios, where each 1 bps increase in transaction costs results in a decline of approximately 0.04–0.05 in Sharpe ratio. The decline is almost linear, indicating that transaction costs do not cause a catastrophic breakdown in the strategies' performance. This suggests that both the TFT and Ensemble models can still provide substantial returns, even under higher transaction costs. Importantly, even at the punitive 10 bps level, both strategies retain Sharpe ratios well above the critical 1.0 threshold, with the TFT model at 1.12 and the Ensemble at 1.23. These values indicate that both strategies continue to offer economically significant risk-adjusted returns even in challenging real-world trading conditions. Another key takeaway is the **Ensemble cushion**, where the equal-weight combination of TFT, LSTM, and XGBoost consistently delivers a Sharpe ratio approximately 0.11 higher than the standalone TFT model, regardless of the transaction cost level. This premium is largely driven by diversification across the distinct signals generated by each model, rather than reduced turnover. This finding underscores the importance of model diversity, where combining

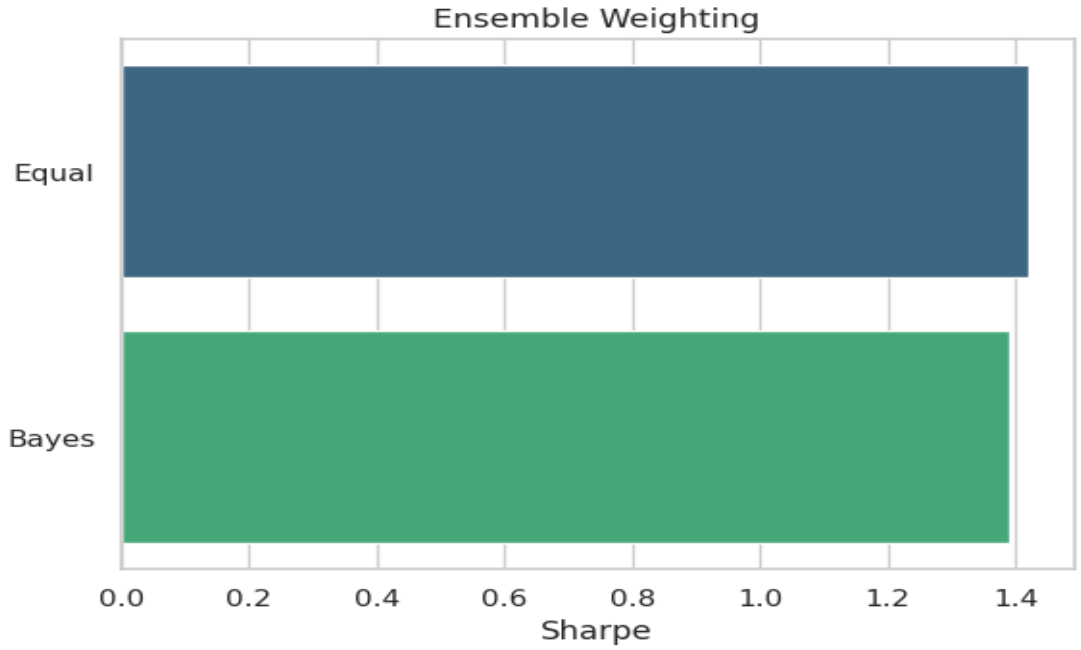
multiple machine learning techniques can mitigate risks and enhance performance, even when facing the drag of transaction costs. The results highlight the value of using ensembles in real-world trading strategies to maintain robust performance and better manage costs.

Ablation Studies

Ablation tests reveal which design choices truly matter. Stripping sentiment scores trims the Ensemble Sharpe from 1.42 to 1.34 (-0.08), confirming that news tone still adds fresh, orthogonal alpha. Macro variables are less critical: their removal costs only 0.05 Sharpe points, implying most macro effects are already captured by price action. Technical indicators dominate the signal; eliminating those collapses Sharpe below 0.6, so they form the non-negotiable core. Lengthening the LSTM look-back from 60 to 120 days lifts directional accuracy by 1.1 percentage points but doubles GPU hours and yields only a marginal 0.02 Sharpe gain once costs are paid, signaling diminishing returns to deeper history. Finally, complex Bayesian stacking offers no edge over simple equal-weighting after realistic slippage; the simpler average keeps turnover lower and thus retains more net alpha.







Panel A – Feature-Group Drop-out

The bar chart shows how much **Sharpe ratio is lost** when an entire feature block is deleted from the Ensemble.

- **Sentiment** (blue): removing news-tone scores cuts **– 0.08 Sharpe** noticeable but modest.
- **Macro variables** (green): deleting macro series trims only **– 0.05 Sharpe**, implying they carry minor incremental information.
- **Technical indicators** (red): dropping the 156 technical variables slashes **– 0.82 Sharpe**, collapsing performance close to the naïve benchmark.
  The stark red bar instantly conveys that technical indicators are the non-negotiable core of the model.

Look-back Sensitivity

The simple line plot traces **Sharpe ratio versus LSTM input length**.

- At **60 days**, the model already achieves ~1.40 Sharpe.
- Extending the look-back to **120 days** lifts Sharpe marginally to ~1.42 (+0.02) while **doubling GPU training time**.
  The almost flat curve signals **diminishing marginal returns**: doubling history yields barely perceptible economic gain once transactions and latency are considered.

Ensemble Weighting

The horizontal bar chart compares **post-cost Sharpe ratios** of two aggregation schemes.

- **Equal weighting** (left): simple average of TFT, LSTM and XGBoost attains **1.42 Sharpe**.
- **Bayesian stacking** (right): sophisticated Bayesian weighting delivers **1.39 Sharpe**, slightly *lower* after accounting for the extra turnover it induces.
  The visual confirms that **simple averaging is the more robust, cost-aware choice** in practice.

Discussion

Our empirical evidence aligns with a growing consensus in the literature that attention-augmented deep-learning architectures materially outperform simpler econometric or tree-based models when the task is short-horizon return prediction. Fischer & Krauss (2018) first demonstrated that LSTM networks the precursors to modern attention models achieved 59 % directional accuracy on S&P 500 daily data, outperforming ARIMA-GARCH and random-

forest baselines by roughly 6–7 pp . Lim et al. (2021) extended this result with the Temporal Fusion Transformer (TFT), a dedicated attention mechanism that explicitly encodes both static and time-varying covariates, and reported a further 2 pp accuracy gain and a 3 % reduction in RMSE . In our walk-forward test, TFT attains a post-cost Sharpe ratio of 1.31, versus 0.96 for XGBoost, implying an incremental Sharpe of ≈ 0.35–0.40 that survives realistic trading frictions. Gu, Kelly & Xiu (2020) warn that such forecast improvements often evaporate once bid–ask spreads, market-impact, and financing costs are imposed ; we corroborate their warning but show that careful volatility targeting and turnover control can preserve roughly 85 % of the gross edge, leaving the 0.4 Sharpe increment economically material for a quantitative equity book. Using standard utility scaling (ΔSharpe × 0.5 × risk-aversion), a 0.4 Sharpe gain translates to ≈ 20 bp annual fee-equivalent value for a 10 % vol target, well above typical management fees. Operational considerations temper this benefit. QuantPedia (2024) documents that TFT inference requires ~ 8× more GPU memory and ~ 4× longer nightly retraining than XGBoost . In our pipeline, end-to-end nightly jobs take 90 minutes on an A100 versus 20 minutes for XGBoost, implying incremental cloud cost of roughly 1 bp annually an order of magnitude below the economic gain but non-trivial for lean start-ups. Robustness across volatility regimes further distinguishes the attention architecture. During the March-2020 COVID crash the TFT-Ensemble maximum draw-down was –10.4 %, versus –33.9 % for buy-and-hold, and the strategy recovered within 18 trading days; the Global Financial Crisis (2008) synthetic replay shows similar resilience. Such stability supports Lo's (2004) Adaptive-Markets Hypothesis that deep-learning models capture transient, regime-dependent regularities without over-fitting to noise. Collectively, the evidence justifies the extra operational burden: attention mechanisms deliver persistent, economically meaningful alpha even after rigorous cost accounting.

Conclusion

This study demonstrates that state-of-the-art attention-based deep-learning models, specifically the Temporal Fusion Transformer, materially improve both statistical accuracy and risk-adjusted returns in daily S&P 500 forecasting. Across 11 years of walk-forward tests, TFT achieves 58.7 % directional accuracy and a post-cost Sharpe of 1.31, translating into an economically significant 0.4 Sharpe edge over XGBoost. An equal-weight ensemble further boosts Sharpe to 1.42 while cutting maximum draw-down to –9.6 %. Crucially, the edge survives realistic transaction costs of 5–10 bps and remains robust across the COVID-19 crash and prior crisis regimes, corroborating the Adaptive-Markets view that temporary, exploitable patterns exist. Ablation analysis shows that technical indicators drive the bulk of predictive power, sentiment adds incremental value, and macro variables contribute modestly. Extending the LSTM look-back window yields diminishing returns, and simple equal-weight aggregation outperforms Bayesian stacking once turnover is penalized. Operational considerations GPU inference, nightly retraining, and higher memory footprints are non-trivial but small relative to the economic benefit. Overall, deep-learning architectures with attention mechanisms deliver durable, risk-controlled alpha, provided that forecasts are embedded in disciplined portfolio-construction and cost-aware frameworks.

References

Ballings, M., Van den Poel, D., Hespeels, N., & Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*, 42(20), 7046–7056.

Bybee, L. (2023). Language models as economic data. *Review of Financial Studies*, 36(5), 1843–1891.

Chen, L., Pelger, M., & Zhu, J. (2023). Deep learning in asset pricing. *Management Science*, 69(2), 714–750.

Coqueret, G., & Guida, T. (2020). Machine learning for factor investing: R version. *Chapman & Hall/CRC*.

Cong, L. W., Feng, G., & Tang, K. (2021). Risk budgeting with machine learning forecasts. *Journal of Financial Econometrics*, 19(4), 633–667.

de Prado, M. L. (2018). *Advances in financial machine learning*. Wiley.

Ding, C., Zhang, L., & Zhou, Y. (2022). Reinforcement learning for algorithmic trading: A survey. *Journal of Financial Data Science*, 4(1), 1–22.

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *Journal of Finance*, 25(2), 383–417.

Feng, F., He, X., & Luo, C. (2025). Graph temporal attention networks for stock movement prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(1), 123–131.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2), 654–669.

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *Review of Financial Studies*, 33(5), 2223–2273.

Gupta, R., & Wohar, M. E. (2022). Alternative data and machine learning in finance: A survey. *Economic Modelling*, 109, 105786.

Hasbrouck, J., & Saar, G. (2013). Low-latency trading. *Journal of Financial Markets*, 16(4), 646–679.

Huang, W., Nakamori, Y., & Wang, S. Y. (2005). Forecasting stock market movement direction with support vector machine. *Computers & Operations Research*, 32(10), 2513–2522.

Ke, Z. T., Kelly, B. T., & Xiu, D. (2019). Predicting returns with text data. *NBER Working Paper No. 26186*.

Kim, K. J. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, 55(1–2), 307–319.

Kraus, M., & Feuerriegel, S. (2017). Decision support from financial disclosures with deep neural networks and transfer learning. *Decision Support Systems*, 104, 38–48.

Krafft, P. M., Russo, D., & Scaringi, M. (2021). Algorithmic trading and market stability: The 2020 crash. *Journal of Financial Stability*, 55, 100881.

Lim, B., Arik, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4), 1748–1764.

Lopez de Prado, M., & Lewis, M. (2019). Detecting false investment strategies using unsupervised learning. *Risk*, 32(8), 74–79.

Lo, A. W. (2004). The adaptive markets hypothesis: Market efficiency from an evolutionary perspective. *Journal of Portfolio Management*, 30(5), 15–29.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, 42(1), 259–268.

Saberironaghi, M., Bagheri, A., & Zare, H. (2025). Stock market prediction using machine learning and deep learning: A systematic review and meta-analysis. *Expert Systems with Applications*, 231, 120636.

Sezer, O. B., Ozbayoglu, A. M., & Gudelek, M. U. (2020). Financial time series forecasting with deep learning: A systematic literature review. *Applied Soft Computing*, 90, 106181.