Proceedings of ICFTBA 2025 Symposium: Data-Driven Decision Making in Business and Economics
DOI: 10.54254/2754-1169/2025.BL28705

Comparative Analysis of ARIMA, Multiple Linear Regression, and LSTM Models for Stock Price Prediction: Evidence from Starbucks and Luckin Coffee

Fujia Zhang

Stony Brook Institute at Anhui University, Hefei, China
R22314054@stu.ahu.edu.cn

Abstract. This paper compares three forecasting approaches-autoregressive integrated moving average (ARIMA), multivariate linear regression (MLR), and long short-term memory networks (LSTM)-for daily stock prices of Starbucks (SBUX) and Luckin Coffee (LKNCY). Trading calendars are aligned across tickers, missing observations are forward-filled, and technical indicators are engineered (log returns and lags, SMA/EMA, RSI, and volume change). A chronological split (80% train, 20% test) prevents look-ahead bias. Performance is evaluated using RMSE, MAE, and MAPE; Diebold-Mariano (DM) tests assess pairwise differences in forecast errors. On LKNCY, MLR attains the lowest error (RMSE 0.891; MAPE 2.74%), outperforming a baseline ARIMA (RMSE 7.155) and a univariate LSTM trained on prices with min-max scaling (RMSE 6.849). Results on SBUX show the same ranking. Diebold-Mariano tests show no statistically significant difference between LSTM and ARIMA forecast errors (SBUX: p=0.52; LKNCY: p=0.70). Diagnostics further indicate that, during downtrends, the price-target LSTM drifts toward the lower bound of the training range, a pattern consistent with sensitivity to scaling and distributional shift. To mitigate this issue, a robustness variant is introduced that predicts next-period log-returns using z-score standardization with a multi-layer LSTM. Taken together, the results emphasize the short-horizon strength of simple linear baselines and the central role of target choice, scaling, and evaluation protocol in financial time-series forecasting.

Keywords: stock price forecasting, ARIMA, multiple linear regression, LSTM

1. Introduction

Stock price prediction occupies a central place in financial economics and quantitative investing because near-term forecasts inform portfolio allocation, risk control, and corporate planning [1-3]. Daily equity series, however, are among the most challenging time series to model, as they are noisy, display volatility clustering, and experience structural breaks, resulting in model performance that can vary substantially across market regimes [2-3]. Within this context, applied work commonly confronts a choice among three modeling paradigms that encode different assumptions about dynamics and signal exploitation: ARIMA, multiple linear regression (MLR), and long short-term memory (LSTM) networks [4-7].

DOI: 10.54254/2754-1169/2025.BL28705

ARIMA captures persistence via differencing and low-order autoregressive/moving-average terms, offering a parsimonious and interpretable baseline when dependencies are short-lived and approximate stationarity holds [1,3]. MLR relates next-period price or return to observable covariates-lags of returns, moving-average signals, relative strength, and volume dynamics-yielding a transparent, factor-style specification that is amenable to auditing and stress testing [2,4].

LSTM, a gated recurrent architecture, addresses vanishing/exploding gradients and is capable of learning nonlinear interactions and long-range dependencies from sequences; empirical applications and surveys report strong performance when data volume is adequate, targets and scaling are chosen well, and horizons are sufficiently long to leverage its capacity [5-7]. Because these families rest on different statistical and algorithmic premises, direct comparison under a single pipeline is necessary for conclusions that generalize beyond a single asset or study.

In much of the applied literature, however, reported results are difficult to compare because preprocessing and evaluation protocols differ-for example, trading calendars are not aligned, gaps are handled inconsistently, targets alternate between prices and returns, and splits mix chronological and random partitions. Such heterogeneity blurs cross-model takeaways. Moreover, claims about superior accuracy should be corroborated by tests of predictive-accuracy differences, such as the Diebold-Mariano test [8]. Evidence from large-scale forecasting evaluations further shows that well-tuned statistical baselines and simple combinations can be highly competitive on short horizons with limited features [9]. For reproducibility and independent verification, widely used tooling, e.g., the forecast package for ARIMA in R, facilitates transparent model specification and replication [1,10].

This study conducts an apples-to-apples comparison of ARIMA, MLR, and LSTM on two contrasting equities: Starbucks (SBUX), a mature multinational with a long trading history, and Luckin Coffee (LKNCY), a younger, more volatile issuer. Trading calendars are aligned across tickers, occasional gaps are forward-filled, and feature engineering is standardized (log-returns and lags, SMA/EMA, RSI, and volume-change proxies).

A strictly chronological split (80% train, 20% test) helps avoid look-ahead bias, and performance is summarized by RMSE, MAE, and MAPE. DM tests are applied to paired one-step errors to assess whether the differences between models are statistically meaningful [1,8,10]. Given that target definition and scaling strongly affect sequence models, a price-target LSTM (with min–max scaling) is complemented by a return-target robustness extension employing z-score standardization and regularization, consistent with best practices reported in the sequence-learning literature [5-7].

By placing these three paradigms under an identical data pipeline and evaluation protocol, the paper aims to provide generalizable guidance on when transparent linear baselines suffice and when flexible deep sequence models are likely to add value. The focus is on principled model choice-aligning methods with horizon, data richness, and regime stability—rather than on isolated headline numbers, thereby offering evidence that can inform both academic research and practical decision-making in financial time-series forecasting [1-3,8-10].

2. Literature review

Autoregressive Integrated Moving Average (ARIMA) models combine differencing (I) with autoregressive (AR) and moving average (MA) terms and continue to serve as a canonical baseline when dependencies are short-lived and approximate stationarity is plausible [1,3]. In equity applications, near-unit-root characteristics can lead ARIMA models to exhibit random-walk-like projections during trending regimes, while conditional heteroskedasticity and nonlinearity may compromise multi-step accuracy. Remedies such as ARIMAX and rolling re-estimation can alleviate these issues at the cost of additional specification work [1-3].

DOI: 10.54254/2754-1169/2025.BL28705

Multiple linear regression (MLR) provides a transparent specification that links next-period price or return to engineered predictors—including lags, moving-average signals, relative-strength measures, and volume dynamics-offering interpretability and straightforward diagnostics [2,4].

Good practice includes modeling returns to reduce nonstationarity, checking multicollinearity, and enforcing information-set timing to avoid look-ahead bias. Dynamic regression provides a principled extension when exogenous drivers are included [2,4]. LSTM addresses vanishing/exploding gradients in recurrent nets and can capture nonlinear interactions and long memory. Finance applications and surveys report favorable results when data volume and horizons are sufficient and when design choices (target = returns vs. prices; scaling; window; capacity; regularization) are appropriate [5-7].

Fair comparison requires consistent calendar alignment and gap handling, chronological train-test protocols, target and scaling choices (returns with z-scores vs. prices with min–max), and evaluation with both point metrics and statistical tests, such as DM [1,5,8].

3. Methodology

3.1. Data and preprocessing

The analysis uses two equities—Starbucks (SBUX) and Luckin Coffee (LKNCY)—at daily frequency with close and volume. Columns are auto-detected (date, price, volume), dates are parsed into a unified format, and nonpositive or nonnumeric prices are removed. To ensure a common timeline, the union of trading days across the two tickers is constructed, each series is left-joined to that calendar, and occasional gaps are forward-filled so that comparisons are not confounded by asynchronous holidays. After feature construction (Section 3.2), leading observations rendered undefined by indicator windows are discarded. A strictly chronological split allocates 80% of each series to training and 20% to testing (no shuffling) to avoid look-ahead bias. In the sample, this yields approximately 6,478/1,620 trading days for SBUX (train/test) and 1,060/266 for LKNCY.

Daily historical prices and volumes for Starbucks (SBUX) and Luckin Coffee (LKNCY) were taken from the Kaggle dataset Top 10 Fast Food Giants: Stock Price Dataset (2024) curated by Nguyen Tien Nhan [11]. The collection aggregates per-ticker CSV files originally compiled from Yahoo Finance and released under the CC0 Public Domain license. Each file contains the standard fields—Date, Open, High, Low, Close, Adjusted Close, Volume—from which Adjusted Close is used as the price series to account for splits and dividends. The specific files utilized are SBUX.csv and LKNCY.csv as provided in [11]; access dates are recorded in the reference entry.

3.2. Feature engineering

From the price P_t log-returns $r_t = \ln(P_t/P_{t-1})$, and their lags r_{t-1} , r_{t-2} are computed. Standard technical indicators are added: simple moving averages SMA_5, SMA_{20} , exponential moving averages EMA_{12} ,relative strength index RSI_{14} ,and a volume-change proxy $\Delta \ln(1 + V_t)$. To reduce scale effects, price-like indicators (SMA/EMA/RSI) are expressed as ratios to P_t . All non-finite values are removed to prevent numerical artifacts in model fitting.

3.3. Model specifications

ARIMA (univariate). The price series is modeled using auto.arima with seasonal = FALSE, stepwise = TRUE, and approximation = FALSE. Model orders are selected by information criteria, and test-

DOI: 10.54254/2754-1169/2025.BL28705

set forecasts are generated in a single multi-step pass.

Multiple Linear Regression (MLR). The baseline specification predicts the next-period price from contemporaneous features:

$$P_{t+1} = \beta_0 + \beta_1 P_t + \beta_2 r_{t-1} + \beta_3 r_{t-2} + \beta_4 SMA_5 + \beta_5 SMA_{20} + \beta_6 EMA_{12} + \beta_7 RSI_{14} + \beta_8 \Delta \ln(1+V_t) + \varepsilon_t \quad (1)$$

This provides a strong, interpretable short-horizon baseline because P_t carries substantial information for P_{t+1}. For robustness, a return-target variant (predicting r_{t+1}) is also outlined for diagnostic comparisons.

LSTM (deep sequence model). The primary LSTM is implemented in R Torch as a univariate model: inputs are sliding windows of length lookback on the scaled price; the network comprises a single LSTM layer (hidden = 64) followed by a dense output; optimization uses Adam with a learning rate 10^{-3}; training is conducted for 40 epochs. Test-time prediction is closed-loop, i.e., the model's previous prediction is fed back to roll the window without injecting future ground truth. Because price-scale drift can bias min–max normalization in downtrends, a robustness extension is introduced that targets next-period log-returns under z-score standardization and can employ a multi-layer LSTM with dropout; results from this variant are reported for completeness.

3.4. Evaluation metrics and statistical testing

RMSE, MAE, and MAPE are reported on the test set:

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}\left(\hat{y}_t - y_t\right)^2}, \quad MAE = \frac{1}{N}\sum_{t=1}^{N}\left|\hat{y}_t - y_t\right|, \quad MAPE = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{\hat{y}_t - y_t}{y_t}\right| \quad (2)$$

where y_t denotes the ground-truth test value at time t, ŷ_t the model prediction, and N the number of test observations. To assess whether differences in forecast accuracy are statistically meaningful, the Diebold–Mariano (DM) test is applied to paired one-step-ahead errors with quadratic loss (power = 2). All models are evaluated under the same chronological train/test split and test window to ensure like-for-like comparison. DM tests use one-step-ahead errors (h = 1) with squared-error loss and HAC variance (Newey–West) with truncation lag q chosen as ⌊T^(1/3)⌋ (robustness: q∈{1,3,5} yields the same significance calls).

3.5. Implementation and reproducibility

All experiments are implemented in R using the tidyverse, lubridate, TTR, forecast, and torch/luz packages. Random seed is fixed at 2025. Transformations that require fitted parameters (e.g., scaling) are computed only on the training set and then applied to the test set to avoid information leakage. A lightweight hyperparameter scan for the univariate LSTM explores $lookback \in \{15,20,30\}$, hidden $units \in \{32,64\}$, epochs=20, and learning rate = 10^{-3}, to gauge sensitivity to window length and capacity. The pipeline mirrors the empirical section so that results are fully auditable and reproducible.

CSV files from [11] are parsed, nonpositive/nonnumeric prices are removed, and trading calendars are unified across tickers. Occasional gaps are forward-filled to avoid artificial lead–lag effects. Feature construction relies only on information available at time t: log returns

DOI: 10.54254/2754-1169/2025.BL28705

$r_t = \ln(P_t/P_{t-1})$ and their lags, SMA_5, SMA_{20}, EMA_{12}, RSI_{14}, and a volume-change proxy $\Delta \ln(1+V_t)$. Indicator ratios to P_t reduce scale sensitivity; non-finite observations are dropped. A chronological 80/20 split is then applied per ticker, after which ARIMA, MLR, and LSTM are estimated as specified in Section 3, and performance is evaluated by RMSE/MAE/MAPE with Diebold–Mariano tests on paired one-step errors.

4. Results

General patterns across both tickers.(1) Short horizons with limited features favor linear baselines. When forecasting one step ahead using a compact set of technical features, MLR consistently yields the strongest and most stable accuracy, reflecting the tight anchoring of P_{t+1} to P_t and the incremental signal from simple indicators [2,7-8].(2) ARIMA underreacts in trending windows. Differencing often makes ARIMA behave like a random-walk extrapolator, producing almost flat multi-step forecasts during pronounced trends; it remains a valuable diagnostic and quick baseline rather than a dominant forecaster [1-3].(3) Price-target LSTM is sensitive to scaling and distribution shift. With min–max normalization fit on the training window, predictions can “floor” near the historical range during test-period downtrends. LSTM’s advantages are more likely when the target is returns, scaling is z-score, data are larger, and horizons are longer-conditions that better exploit nonlinear memory [5-7,9].(4) Statistical significance matters. Diebold-Mariano tests on paired one-step errors frequently show no significant difference between ARIMA and price-target LSTM, highlighting that flexibility does not automatically deliver reliably better forecasts on daily horizons [8].

5. Discussion

Across both equities, the linear baseline (MLR) dominates ARIMA and the univariate, price-target LSTM. This outcome is consistent with short-horizon equity dynamics in which P_{t+1} is strongly anchored to P_t, and a small set of well-chosen technical covariates, add incremental signal. In contrast, ARIMA tends to collapse toward a random-walk–like extrapolation when differencing is required, producing nearly horizontal multi-step forecasts during pronounced trends. The price-based LSTM underperforms largely because target definition and scaling interact unfavorably with distributional shift: min–max normalization learned on the training window biases predictions toward the historical range, causing a “flooring” effect when the test period drifts downward.

First, target choice matters. Modeling next-period log-returns rather than prices reduces level effects and stabilizes training; a robustness design using a return-target LSTM with z-score scaling addresses this issue. Second, fair comparisons require a unified protocol. Holding calendar alignment, feature availability, splits, and evaluation constant ensures that differences reflect modeling capacity rather than pipeline artifacts. Third, capacity and window length for LSTM should be right-sized to the data. Evidence from the hyperparameter scan suggests that a modest capacity (hidden $\approx$ 64, lookback $\approx$ 20) is preferable in this sample; larger windows may introduce stale information and amplify smoothing.

Diebold–Mariano tests indicate no significant difference between ARIMA and LSTM errors, even though point metrics vary. This underscores the distinction between numerical improvements and statistically reliable gains. For practitioners, the decision criterion should combine accuracy, stability, interpretability, and operational cost.

The study uses two equities and daily frequency; results may differ at intraday horizons or across broader cross-sections. Exogenous drivers (macro announcements, earnings surprises, sentiment) are

DOI: 10.54254/2754-1169/2025.BL28705

not modeled explicitly in the baseline experiments. Hyperparameter tuning for LSTM is intentionally lightweight to avoid overfitting the test set; deeper searches might improve neural results but reduce interpretability and reproducibility if not appropriately nested.

For near-term forecasting on mature large caps and volatile growth names alike, a transparent MLR with carefully engineered features can be a competitive —and often superior—choice. ARIMA remains a quick-to-deploy baseline and a useful diagnostic tool for autocorrelation structure, while LSTM becomes attractive when the target is returns, external covariates are plentiful, and sample size allows for regularized training and walk-forward updating.

Use MLR when the horizon is short, interpretability is required, and only compact engineered features are available; prefer return targets, check multicollinearity, and control look-ahead [2,4,7].

Use ARIMA/ARIMAX as a fast baseline and for autocorrelation diagnostics; expect underreaction in strong trends unless exogenous signals are included and models are re-estimated regularly [1-3].

LSTM should be employed in settings with larger samples, richer covariates, and/or longer horizons; adopt return targets with z-score scaling, incorporate dropout/weight decay, and evaluate via walk-forward protocols to mitigate regime shifts [5-7,9]. Complexity does not guarantee better performance. Align model choice with data characteristics (sample size, regime stability, horizon) and forecasting goals (accuracy vs. interpretability vs. operational cost), and report both point metrics and DM tests to separate numerical gains from statistically reliable improvements [1,5,8-9].

6. Conclusion

This study compared ARIMA, multiple linear regression (MLR), and LSTM within a single, transparent pipeline for two contrasting equities: Starbucks (SBUX) and Luckin Coffee (LKNCY). Trading calendars were aligned, gaps were forward-filled, features were standardized, and a chronological split prevented look-ahead bias. Three general lessons emerge.

First, linear baselines dominate at short horizons with compact feature sets. With one-step targets and a small, carefully engineered set of indicators, MLR provided the most accurate and stable forecasts. This outcome is consistent with the near-unit-root nature of daily prices, which P_{t+1} is strongly anchored to P_t , and a few technical indicators add incremental signal. In practice, a well-specified regression is an appropriate default starting point before increasing model complexity.

Second, ARIMA remains a valuable yet cautious baseline. In trending regimes, differencing tends to push ARIMA toward random-walk-like trajectories that underreact to persistent moves, so leadership on multi-step test windows is uncommon. Even so, ARIMA is fast to fit, offers diagnostics for autocorrelation and seasonality, and serves as a useful benchmark or scaffold when exogenous drivers are available.

Third, LSTM's gains are conditional on target definition and scaling. A price-target LSTM trained with min-max scaling tended to drift toward the training range during downtrends, indicating scale anchoring under distributional shift. LSTM becomes more promising when the target is returns, scaling follows z-score standardization, regularization is present, the sample is larger, and the horizon is longer-conditions that allow nonlinear memory to be exploited. Empirically, Diebold–Mariano comparisons did not show a statistically significant accuracy gap between the price-target LSTM and ARIMA, underscoring that additional flexibility does not automatically translate into reliably better daily forecasts.MLR is recommended for short-horizon forecasting when interpretability, stability, and rapid iteration are priorities; return targets, multicollinearity checks, and strict information-set timing are advisable. ARIMA/ARIMAX is best used as a rapid baseline and diagnostic tool, with rolling re-estimation and exogenous regressors to mitigate trend

DOI: 10.54254/2754-1169/2025.BL28705

underreaction. LSTM is most appropriate when richer covariates (e.g., macro, earnings, sentiment), longer horizons, and sufficient data are available; return targets with z-score scaling and regularized training evaluated in walk-forward backtests are preferable.

The analysis covers two equities at daily frequency, a fixed single-step horizon, and a compact set of technical features. Macro, news, and sentiment were not modeled explicitly, and the neural hyperparameter search was intentionally lightweight to avoid test-set overfitting. These choices support reproducibility but may understate the upside of deep sequence models in data-rich settings.Promising directions include: (i) expanding to broader cross-sections and multiple market regimes; (ii) studying multi-horizon and intraday forecasts; (iii) integrating exogenous information (macro releases, earnings surprises, order flow, and sentiment/news embeddings) via dynamic regression and multi-channel LSTMs; (iv) adopting walk-forward retraining with nested hyperparameter tuning and change-point detection; (v) moving from point forecasts to probabilistic ones (quantiles and intervals) and evaluating economic utility under transaction costs and risk constraints; and (vi) testing hybrid/ensemble strategies such as linear-neural stacking and simple combinations.

Overall, the evidence supports a discipline-first approach to equity forecasting: begin with strong linear baselines, document the pipeline rigorously, and escalate complexity only when the data environment and forecasting objective justify it. This pathway converts empirical accuracy into reliable, decision-useful forecasts for financial time series.

References

[1] R. J. Hyndman and G. Athanasopoulos, Forecasting: Principles and Practice, 3rd ed. OTexts, 2021. Available at: https: //otexts.com/fpp3/

[2] R. S. Tsay, Analysis of Financial Time Series, 3rd ed. Hoboken, NJ: Wiley, 2010. Available at: https: //www.wiley.com/en-us/Analysis%2Bof%2BFinancial%2BTime%2BSeries%2C%2B3rd%2BEdition-p-9780470414354

[3] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, Time Series Analysis: Forecasting and Control, 5th ed. Hoboken, NJ: Wiley, 2015. Available at: https: //www.wiley.com/en-us/Time%2BSeries%2BAnalysis%3A%2BForecasting%2Band%2BControl%2C%2B5th%2BEdition-p-9781118675021

[4] A. Pankratz, Forecasting with Dynamic Regression Models. New York: Wiley, 1991. doi: 10.1002/9781118150528. Available at: https: //onlinelibrary.wiley.com/doi/book/10.1002/9781118150528

[5] S. Hochreiter and J. Schmidhuber, “Long short-term memory, ” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735. Available at: https: //direct.mit.edu/neco/article/9/8/1735/6109/Long-Short-Term-Memory

[6] T. Fischer and C. Krauss, “Deep learning with long short-term memory networks for financial market predictions, ” European Journal of Operational Research, vol. 270, no. 2, pp. 654–669, 2018. doi: 10.1016/j.ejor.2017.11.054. Publisher page: https: //www.sciencedirect.com/science/article/pii/S0377221717310652 ; Open-access preprint: https: //econpapers.repec.org/RePEc: eee: ejores: v: 270: y: 2018: i: 2: p: 654-669

[7] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, “Financial time series forecasting with deep learning: A systematic literature review (2005–2019), ” Applied Soft Computing, vol. 90, art. 106181, 2020. doi: 10.1016/j.asoc.2020.106181. Open-access preprint: https: //arxiv.org/abs/1911.13288

[8] F. X. Diebold and R. S. Mariano, “Comparing predictive accuracy, ” Journal of Business & Economic Statistics, vol. 13, no. 3, pp. 253–263, 1995. doi: 10.1080/07350015.1995.10524599. Publisher page: https: //www.tandfonline.com/doi/abs/10.1080/07350015.1995.10524599 ; OA version: https: //www.ssc.wisc.edu/~bhansen/718/DieboldMariano1995.pdf

[9] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The M4 Competition: Results, findings, conclusion and way forward, ” International Journal of Forecasting, vol. 34, no. 4, pp. 802–808, 2018. doi: 10.1016/j.ijforecast.2018.06.001. Publisher page: https: //www.sciencedirect.com/science/article/pii/S0169207018300785 ; OA post-print (example): https: //researchportal.bath.ac.uk/files/192035719/IJF_2019_M4_Editorial_post_print_.pdf

DOI: 10.54254/2754-1169/2025.BL28705

[10] R. J. Hyndman and Y. Khandakar, “Automatic time series forecasting: The forecast package for R, ” Journal of Statistical Software, vol. 27, no. 3, pp. 1–22, 2008. doi: 10.18637/jss.v027.i03. Available at: https: //www.jstatsoft.org/v27/i03

[11] N. T. Nhan, “Top 10 Fast Food Giants: Stock Price Dataset (2024), ” Kaggle Datasets, 2024. License: CC0 Public Domain. Files used: SBUX.csv, LKNCY.csv. Available at: https: //www.kaggle.com/datasets/nguyentiennhan/stock-prices-of-the-10-largest-fast-food-companiesAccessed: Sept. 10, 2025.