AIMS Mathematics, 10(9): 21875–21901.
DOI: 10.3934/math.2025974
Received: 22 July 2025
Revised: 07 September 2025
Accepted: 16 September 2025
Published: 19 September 2025

http://www.aimspress.com/journal/Math

Research article

A novel hybrid LMD-SPF forecasting framework for financial time series: Evidence from gold returns

Muhammad Aamir[1], Hasnain Iftikhar[2,3,*], Jawaria Nasir[1], Paulo Canas Rodrigues[4], Abdulmajeed Atiah Alharbi[5] and Jeza Allohibi[5]

[1] Department of Statistics, Abdul Wali Khan University Mardan, Mardan 23200, Pakistan
[2] Department of Statistics, University of Peshawar, Peshawar 25120, Pakistan
[3] Department of Statistics, Quaid-i-Azam University, Islamabad 45320, Pakistan
[4] Department of Statistics, Federal University of Bahia, Salvador 40170-110, Brazil
[5] Department of Mathematics, College of Science, Taibah University, Madinah 42353, Saudi Arabia

* **Correspondence:** Email: hasnainchill3@gmail.com.

Abstract: Accurately forecasting gold returns is critical for investors, policymakers, and risk managers, yet it remains challenging due to the coexistence of deterministic cycles, stochastic volatility, and nonlinear dependencies. This study introduced a novel hybrid model, local mean decomposition (LMD), stochastic product function (SPF), autoregressive integrated moving average (ARIMA), and long short-term memory (LSTM) (LMD-SPF-ARIMA-LSTM), that integrates signal decomposition, statistical diagnostics, and machine learning to address these complexities. LMD first decomposes the return series into product functions (PFs). An SPF then applies formal tests (augmented Dickey-Fuller (ADF) for stationarity, Brock-Dechert-Scheinkman (BDS) for linearity, and correlation filtering) to classify PFs, allocating them to ARIMA for linear-stationary components and to LSTM for nonlinear-nonstationary ones. Using daily gold return data from June 2020 to May 2025, the proposed framework achieved substantially improves over benchmarks, reducing mean absolute error (MAE) by more than 55% compared to ARIMA, lowering root mean squared error (RMSE) by 57% relative to LSTM, and attaining 85.31% directional accuracy-over three percentage points higher than the best competing hybrid. Unlike previous LMD-ARIMA-LSTM approaches that treat all decomposed components uniformly, our method tailors the modeling strategy to the statistical properties of each PF, reducing redundancy, lowering computational cost, and enhancing generalization. These results not only demonstrate the methodological significance of combining decomposition with statistically informed model selection but also provide practical value by delivering more reliable and interpretable forecasts for financial decision-making in volatile markets.

Keywords: gold return forecasting; local mean decomposition; ARIMA; LSTM; hybrid forecasting

models; financial time series; econometric modeling; deep learning in finance
Mathematics Subject Classification: 62M10, 91G70, 68T07, 62P05

1. Introduction

Time series forecasting remains a central challenge in quantitative finance due to noise, volatility, and regime shifts in financial markets [1–3]. Classical statistical methods such as the autoregressive integrated moving average (ARIMA) [4] provide a strong theoretical basis for linear modeling, but are limited in nonstationary and nonlinear settings, particularly during structural breaks [5, 6]. Recent advances emphasize hybrid methods that combine statistical and machine learning models to leverage their complementary strengths, thereby enhancing predictive accuracy and robustness [7–9].

Decomposition techniques have become an effective tool for handling nonlinear and nonstationary financial data. Popular alternatives include empirical mode decomposition (EMD), ensemble EMD (EEMD), and variational mode decomposition (VMD). Local mean decomposition (LMD), however, offers distinct advantages: it isolates deterministic and stochastic oscillations without mode mixing, requires fewer tuning parameters than VMD, and produces interpretable product functions (PFs) [10–12]. This makes LMD particularly suited to modeling gold returns, where both stochastic volatility and deterministic cycles co-exist. Recent empirical studies highlight the promise of decomposition-driven hybrids for crude oil and equity markets [13–15].

Setyowibowo et al. [16] employed an ARIMA–GARCH hybrid model to forecast daily gold prices using data from March 2016 to December 2020. By combining ARIMA(1,1,1) with GARCH(2,1), the model achieves an RMSE of 2.3755, MAE of 1.7029, and mean absolute percentage error (MAPE) of 0.1168%, indicating its ability to capture both trend and volatility in gold markets. However, Amini et al. [17] proposed a hybrid (convolutional neural network and bidirectional long short-term memory (CNN–BiLSTM)) architecture enhanced with automated parameter tuning via grid search, to forecast closing gold prices using historical data over four decades (1978–2021). Their approach outperformed baseline DNN models such as CNN, CNN–LSTM, Conv–LSTM, and stacked LSTM, especially under extreme value scenarios.

Deep learning, especially long short-term memory (LSTM) networks [18], has shown strong performance in capturing long-range dependencies and volatility clustering in financial time series [19–21]. Empirical work demonstrates that LSTM-based approaches can outperform statistical baselines in commodity forecasting [22–25]. More recently, transformer-based architectures [26, 27] have been explored, offering improved sequence modeling at the cost of higher data and computational requirements. Meanwhile, multimodal and ensemble-driven approaches such as causal-augmented multi-modality event-driven financial forecasting (CAMEF) [28], fusion-based gold forecasting [29], Bitcoin volatility forecasting [30], and reinforcement learning-enhanced exchange rate prediction [31] highlight the growing integration of decomposition, causal inference, and advanced AI in financial forecasting. Nonetheless, the majority of current LMD-based hybrids apply uniform modeling across all PFs, disregarding their distinct statistical properties, which often leads to redundant modeling and reduced generalization.

To overcome these limitations, this work proposes a novel LMD–ARIMA–LSTM framework

enhanced with a stochastic product function (SPF) strategy. An SPF employs augmented Dickey-Fuller (ADF) tests for stationarity, Brock-Dechert-Scheinkman (BDS) tests for linearity, and correlation filtering to classify PFs. Stationary and linear PFs are routed to ARIMA, while nonlinear and nonstationary PFs are modeled with LSTM. This selective allocation ensures that each PF is matched with the most suitable forecasting approach, minimizing redundancy and improving efficiency. Our design builds upon the model selection principles introduced by Hotta and Neto [32] and extends them to a decomposition-driven hybrid forecasting framework.

The contributions of this study are threefold. First, we introduce a statistically informed classification step (SPF) into the LMD framework, a novelty not explored in prior finance literature. Second, we integrate both theoretical insights from time series econometrics and empirical advances from recent decomposition-based hybrids in financial forecasting [28–30, 33, 34]. Third, we demonstrate through a comparative analysis on gold return forecasting that our SPF-enhanced LMD–ARIMA–LSTM model outperforms conventional statistical models, standalone deep learning approaches, and existing decomposition-based hybrids.

The remainder of this paper is structured as follows. Section 2 presents the methodological framework, including LMD, SPF, ARIMA, and LSTM integration. Section 3 outlines the evaluation metrics used for performance assessment. Section 4 reports experimental results and comparative benchmarks. Section 5 concludes with contributions and implications, and suggests avenues for future research.

2. Materials and methods

This study proposes a hybrid framework-MD-SPF-ARIMA-LSTM—aimed at improving the prediction accuracy of daily gold return time series. The methodological workflow is illustrated in Figure 1, which integrates decomposition, statistical modeling, and deep learning stages to handle both the linear and nonlinear structures inherent in financial time series.

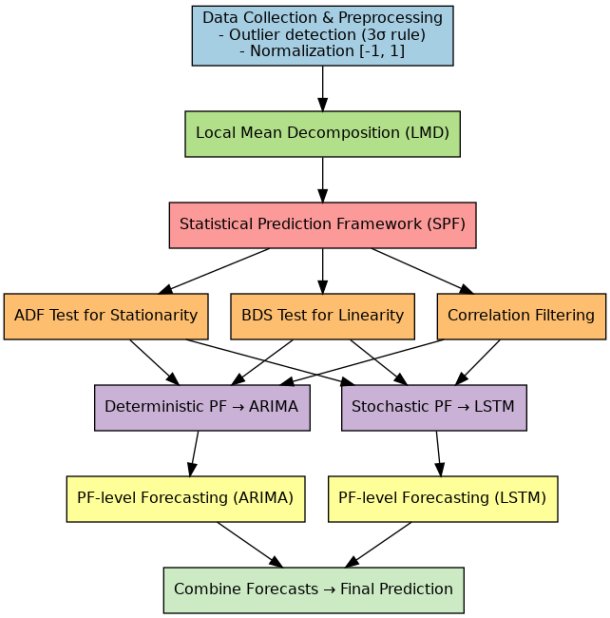

Figure 1. Flowchart of the proposed hybrid forecasting framework.

2.1. Local mean decomposition

Local mean decomposition (LMD) is an adaptive signal decomposition method tailored for analyzing nonstationary and nonlinear time series [10]. It decomposes an input signal into a sum of product functions (PFs), each comprising an amplitude envelope and a frequency-modulated signal. This enables isolation of intrinsic oscillatory components, making it a valuable preprocessing tool for financial return series where stochastic volatility and cyclical movements coexist [12].

Given a financial signal $x(t)$, the LMD algorithm proceeds as follows:

Step 1: Estimation of the local mean and envelope

The first step in local mean decomposition (LMD) is to detect all local extrema of the input signal $x(t)$. From these extrema, we estimate the local mean, $m(t)$, and the local envelope, $a(t)$. To avoid abrupt fluctuations, both $m(t)$ and $a(t)$ are smoothed using a moving average filter.

This step separates the slow trend (the average behavior of the signal) from the faster oscillations, much like identifying the overall direction of a wavy curve.

Step 2: Extraction of the oscillatory component

Once the local mean is obtained, it is subtracted from the original signal to isolate the oscillatory part:

$$s(t) = x(t) - m(t). \tag{2.1}$$

Subtracting the average trend $m(t)$ leaves behind the fluctuations in the data — the part that oscillates around the mean.

Step 3: Normalization and product function formation

The oscillatory component $s(t)$ is then normalized using the envelope $a(t)$ to obtain a pure frequency-modulated signal:

$$f(t) = \frac{s(t)}{a(t)}. \tag{2.2}$$

This normalization removes changes in amplitude, so that $f(t)$ reflects only how fast the oscillations are changing over time (their frequency), not their size.

Next, the product function (PF) is formed by recombining the envelope and the normalized oscillatory component:

$$PF(t) = a(t) \cdot f(t). \tag{2.3}$$

Here, the amplitude and frequency information are merged back together into a single component, called a product function. This PF represents one intrinsic mode of the signal — similar to isolating one "layer" of oscillation from a complex waveform.

Finally, the residual signal $x(t) - PF(t)$ is computed, and the same procedure is repeated until the residual becomes monotonic or constant.

This iterative process gradually breaks the original signal into simpler building blocks, each capturing a distinct oscillatory behavior.

Each $PF_i(t)$ can be modeled individually using suitable forecasting methods. In this framework, LMD serves as a decomposition gateway for mapping each subcomponent to either ARIMA or LSTM,

depending on its statistical properties. In other words, LMD deforms the thorny, heavy traffic time series into a collection of simple building blocks, each with its provisional pattern of oscillation and its own amplitude. This splitting allows for the modeling of various components of the signal in the most optimal manner, as opposed to modeling the entire series with a single model. Within LMD, the representation of every product function (PF) is an oscillatory mode to be removed from the original time series, and it is composed of two parts: An amplitude-modulated signal and a frequency-modulated signal. In financial data, those same PFs may be viewed as comprising elements that reflect various kinds of market activity, where some elements are dominated by short-term, random fluctuations (stochastic) and others exhibit more classic cyclical trends (deterministic).

We have added explicit details of the moving average filter window size (5 points) and the residual energy threshold (1×10^{-6}) used to stop the decomposition iterations, ensuring full reproducibility. In this study, we classify PFs into stochastic and deterministic categories based on their statistical properties. Stochastic PFs exhibit low autocorrelation and higher noise levels, and are often nonlinear and nonstationary, making them suitable for modeling with LSTM. Deterministic PFs show stronger autocorrelation, clearer periodicity, and stationarity, making them more appropriate for ARIMA. For example, in our gold return dataset, PF1 captured high-frequency noise from short-term trading, while PF4 reflected slower, more predictable price movements linked to macroeconomic cycles.

2.2. ARIMA model

The autoregressive integrated moving average (ARIMA) model is a foundational tool for linear time series analysis, especially well-suited for stationary processes with autocorrelated structures [4, 5]. An ARIMA(p, d, q) model includes autoregressive terms (AR), differencing steps (I), and moving average terms (MA), and is defined as:

$$y_t = c + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j} + \epsilon_t, \tag{2.4}$$

where:

- c is a constant term;
- ϕ_i and θ_j are AR and MA coefficients;
- ϵ_t is a white noise error term.

Intuitively, ARIMA predicts the current value of the time series by combining past values (autoregression), the cumulative sum of changes (integration), and patterns in past errors (moving average). The parameters p, d, and q control the number of past points used, the number of times the series is differenced to remove trends, and the extent to which error patterns are considered, respectively. The ARIMA modeling pipeline includes:

- Checking for stationarity (using ADF or KPSS tests) and differencing if necessary.
- Identifying p, d, and q using autocorrelation (ACF) and partial autocorrelation (PACF) plots.
- Estimating model parameters using maximum likelihood estimation (MLE) or least squares.
- Selecting the best model using information criteria (AIC, BIC) [32].
- Diagnosing residuals to verify model adequacy.

In this hybrid framework, ARIMA is applied to PFs that are linear and stationary, as identified through residual diagnostics and unit root testing.

2.3. Long short-term memory

Long short-term memory (LSTM) networks, a specialized class of recurrent neural networks (RNNs), were introduced to address the limitations of conventional RNNs in modeling long-term dependencies [35, 36]. Financial data, particularly return series, often exhibit temporal dependencies and nonlinear dynamics that LSTM networks can effectively capture [21, 37, 38].

The LSTM unit incorporates memory cells and gating mechanisms:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{2.5}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2.6}$$

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{2.7}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{2.8}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{2.9}$$

$$h_t = o_t \odot \tanh(C_t). \tag{2.10}$$

Here, x_t is the input at time t, h_t is the hidden state, C_t is the cell state, σ is the sigmoid activation function, $\odot$ denotes element-wise multiplication. In simpler terms, the LSTM acts like a "smart memory" that decides what information to keep, what to discard, and what to pass forward at each step. This enables the model to capture both short-term fluctuations and longer-term trends in the data, which is particularly useful for financial time series where recent events and historical context are both relevant. The flexibility of LSTM in modeling nonlinear and long-memory structures makes it ideal for forecasting residual or nonlinear PFs from LMD, enhancing the model's ability to capture complex temporal relationships.

2.4. Hybrid forecasting strategy

After decomposing the gold return series using LMD, each PF is examined for stationarity and linearity. Based on its statistical properties, 1) ARIMA is applied to PFs exhibiting linear and stationary behavior. 2) LSTM is used for PFs that display nonlinear or nonstationary characteristics. The individual forecasts from ARIMA and LSTM models are aggregated to reconstruct the final predicted return series. This model fusion approach leverages the strengths of both statistical and deep learning techniques to produce accurate and stable forecasts in financial environments marked by volatility and complexity [39].

2.5. Evaluation metrics

To evaluate the forecasting accuracy of the proposed LMD-SPF-ARIMA-LSTM model, three standard error metrics are employed: mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). These metrics assess model performance from absolute, squared, and percentage-based perspectives, respectively [40, 41].

$$MAE = \frac{1}{n}\sum_{t=1}^{n} |Y_t - \hat{Y}_t| \tag{2.11}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(Y_t - \hat{Y}_t)^2} \tag{2.12}$$

$$MAPE = \frac{100}{n}\sum_{t=1}^{n}\left|\frac{Y_t - \hat{Y}_t}{Y_t}\right|. \tag{2.13}$$

Here, Y_t is the actual value, $\hat{Y}_t$ is the predicted value, and n is the number of observations. These metrics together ensure a robust evaluation of both error magnitude and variability, suitable for financial return prediction tasks [42, 43].

The practical values of the proposed models are examined with the help of three overlapping error measures: MAE, RMSE, and MAPE. MAE provides an intuitively understandable measure of the mean error magnitude, expressed in the same units as the source series. It is thus suitable for comparing models based on their performance in practical prediction. RMSE also places a greater penalty on large deviations, which is significant in the context of financial forecasting, where potentially extreme deviations may have a disproportionate economic impact. MAPE presents the error in percentage form, allowing non-scale-dependent discrimination between any set of data and asset types. Combined, these measures enable an even assessment of the mean and high-end inaccuracy of forecasts, while also providing a readily interpretable percentage-based performance measure.

The Diebold–Mariano test

To assess whether the predictive performance of competing models differs significantly, this work employs the Diebold–Mariano (DM) test [44]. Widely used in the time series forecasting literature [45–47], this test evaluates whether the forecast errors generated by two competing models are statistically distinguishable. Specifically, it examines the null hypothesis that both models possess equal predictive accuracy.

The DM test statistic is given by:

$$\mathrm{DM}_s = \frac{\bar{Y}}{\sqrt{\mathrm{Var}(\bar{Y})}}, \tag{2.14}$$

where

$$\bar{Y} = \frac{1}{H}\sum_{h=1}^{H} X_h, \quad \text{with} \quad X_h = (Y_h - \tilde{Y}_{1h})^2 - (Y_h - \tilde{Y}_{2h})^2, \tag{2.15}$$

$$\mathrm{Var}(\bar{Y}) = \frac{1}{H}\left(2\sum_{j=1}^{H-1} r_j + r_0\right), \quad r_j = \mathrm{Cov}(X_h, X_{h-j}). \tag{2.16}$$

Here, $\tilde{Y}_{1h}$ and $\tilde{Y}_{2h}$ denote the forecasts from the two competing models at horizon h, while Y_h represents the observed value. A statistically significant DM statistic indicates that the forecast accuracy of the two models is not equivalent, thereby confirming meaningful differences in predictive performance.

3. Empirical analysis

3.1. Data collection and preprocessing

Daily gold price data were obtained from Yahoo Finance, covering the period from June 2, 2020, to May 30, 2025. The daily return series was computed using the formula:

$$\text{Daily Return} = \frac{\text{Close} - \text{Open}}{\text{Open}}. \tag{3.1}$$

This return series serves as the basis for forecasting. This research employed simple percentage returns as opposed to logarithmic returns to ensure that direct interpretability could be made in terms of percentage returns and also to conform to the specifications of some statistical tests (e.g., augmented Dickey–Fuller) used in the classification step of SPF. Although log returns are usually preferable because they possess the properties of addition and the ability to compound long horizons, in our case, we are interested in the very short-term, day-to-day movements, and the numerical difference between simple and log returns is minute. It is also easier to comparatively analyze with previous gold price forecasting studies, a substantial number of which use this convention.

Before performing modeling, the observations of the raw series (daily returns of gold) were checked against data quality problems. There were no missing values in the dataset; returns that were more than three standard deviations beyond the mean were marked as possible outliers. Instead of going through a removal process, such extreme points were kept since they represent actual volatility in financial time series. Normalization was then performed by scaling the series to the interval [-1, 1] to maintain numerical stability during LSTM training and to put the ARIMA and LSTM outputs on a similar scale. In the training set, a nested validation strategy was employed, utilizing 15% of the training data as a validation subset to develop a statistical model that would perform well on the test set. This approach enables early stopping and/or the selection of a statistical model with hyperparameters for use. The validation methodology was identical for benchmarking all models (ARIMA, LSTM, and hybrids), so they were both tuned down to the last detail. There were no default parameters. The dataset was divided into training and testing subsets using a 75:25 ratio, resulting in 944 observations for the training set and 314 for the testing set.

As illustrated in Figure 2, the daily returns exhibit erratic fluctuations with no consistent trend, reflecting gold's behavior as a volatile financial instrument. The pronounced spikes and dips are likely responses to macroeconomic shocks, geopolitical tensions, and market uncertainty, underscoring the complexity of forecasting such financial time series.

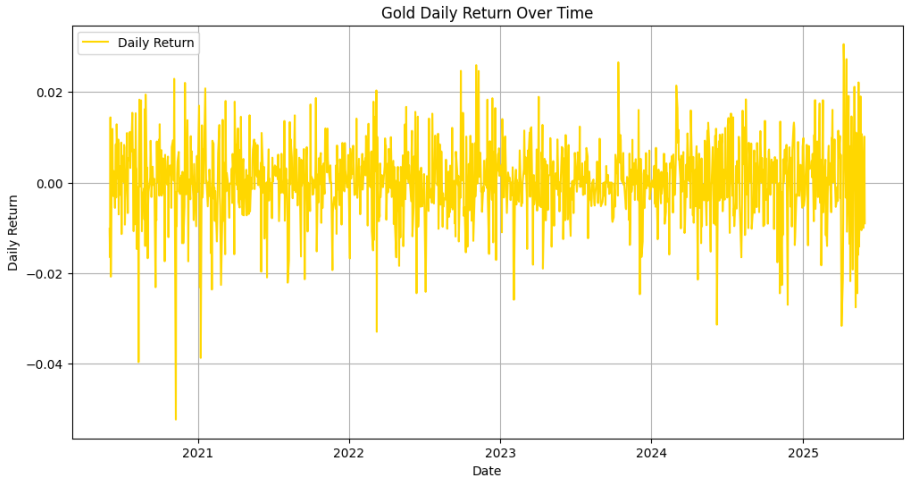

Figure 2. Time series plot of gold daily returns (2020–2025).

3.2. Descriptive statistics

Table 1 presents the descriptive statistics of the gold daily return series, offering key insights into its distributional properties and stationarity, which are foundational for effective financial modeling. The average daily return is relatively small (0.00052), consistent with typical asset return behavior over short horizons. The standard deviation (0.00945) indicates notable volatility, reflecting the inherent risk and fluctuating nature of gold as a financial asset. The minimum and maximum values of -0.0456 and 0.0381, respectively, confirm the presence of significant outliers and highlight the asymmetric behavior in extreme market movements. A positive skewness of 0.25 suggests a slight right-tail bias in the distribution, while the excess kurtosis value of 5.10 points to a leptokurtic distribution—one with heavier tails than a normal distribution. These characteristics are commonly observed in financial return series and justify the use of hybrid models that can handle non-Gaussian features. Along with the descriptive statistics reported in Table 1, the standard deviation of the daily returns was calculated on a 30-day rolling basis, and the results were plotted against time (Figure 3) to see volatility evolve. This rolling volatility plot clearly shows that variance shifts occur, which are associated with periods of intense market activity, followed by quieter periods. This condition also indicates that the returns series are nonstationary in terms of variance. This type of heteroskedasticity is typical of financial time series data, supporting the preference for a breakdown and hybrid modeling approach that can be adapted to different market dynamics. Figure 3: 30-day rolling volatility of daily gold returns with an indication of intervals with large and small variance, indicating the nonstationarity of the series.

Table 1. Descriptive statistics of gold daily returns.

Statistic	Mean	Std. Dev.	Min	Median	Max	Skewness	Kurtosis	ADF	p-value	JB (p)
Value	0.00052	0.00945	-0.0456	0.0003	0.0381	0.25	5.10	-11.3	0.000	180.6 (0.001)

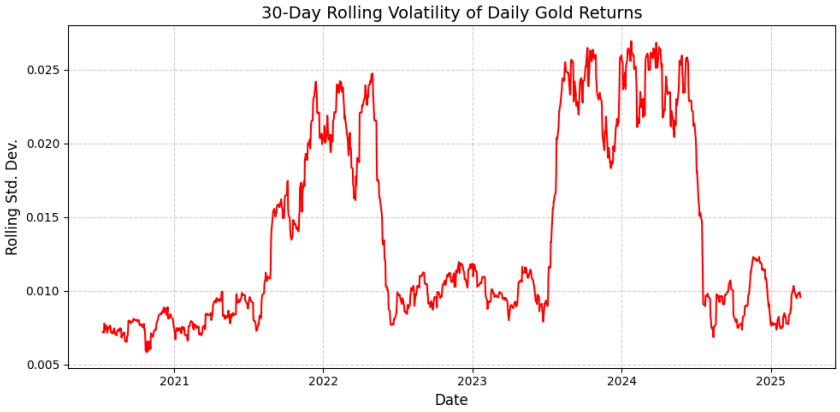

Figure 3. 30-day rolling volitality of gold returns.

Stationarity is confirmed through the augmented Dickey-Fuller (ADF) test, where the test statistic of -11.3 and p-value of 0.000 reject the null hypothesis of a unit root at a 1% significance level. This confirms that the return series is stationary and suitable for time series modeling without differencing. Additionally, the Jarque-Bera (JB) statistic of 180.6 with a p-value of 0.001 strongly rejects normality, reinforcing the need for models capable of capturing nonlinearity, non-normality, and heteroskedasticity in the data. These empirical observations validate the choice of the proposed LMD-SPF-ARIMA-LSTM model, which decomposes and forecasts gold returns by leveraging both statistical and deep learning techniques. This integrated approach aligns well with the goals of this special issue by advancing mathematical modeling frameworks that are both theoretically sound and practically robust for financial time series forecasting.

3.3. LMD-based decomposition and signal reconstruction

To effectively capture the nonlinear and non-stationary dynamics of gold returns, local mean decomposition (LMD) was applied. LMD decomposes the original signal into a finite number of product functions (PFs) and a residual term. These PFs are further categorized into stochastic and deterministic components based on complexity and autocorrelation.

In this study, LMD produced eight meaningful PFs and one residual term. Based on the average mutual information (AMI) plots (see Figure 4), PF1 to PF3 are identified as stochastic, while PF4 to PF8 exhibit deterministic characteristics.

Table 2 presents the correlation analysis, which was employed as a further filtering procedure within the Stochastic Product Function (SPF), to not over-model highly related PFs. In particular, when two PFs have a Pearson correlation coefficient between them that exceeds 0.85 in absolute value, only one of them is used explicitly in modeling other phenomena, with the PF that is richer in predictive structure being preferred. The other, strongly correlated PF was combined with the common path model (to reduce computational expense) to mitigate the threat of overfitting. This filtering, based on correlation, occurred following the initial classification of stationarity and linearity, that is, both PFs —deterministic (DPFs) and stochastic —could undergo merging in the event of redundancy detection. There are pairs of PFs with inordinate pairwise correlations, especially among the deterministic PFs; this is not necessarily redundant with regard to LMD-based decomposition. In LMD, the signal at each PF is a natural mode, characterized by a distinct instantaneous frequency and amplitude modulation,

which can be resolved even in cases where the summed linear correlation is high. Forming mergers of these PFs using only correlation coefficients may cause significant localized oscillatory and fine nonlinearities, which are important for the accuracy of forecasts, to be lost. Nonetheless, we recognize the value of dimension-reduction strategies for efficiency and parsimony. As part of our future work, we plan to explore selective PF aggregation using criteria beyond simple correlation, such as mutual information, spectral similarity, and predictive relevance, to assess whether combining certain PFs can reduce complexity without sacrificing accuracy. This will enable us to test the trade-off between model simplicity and performance formally.

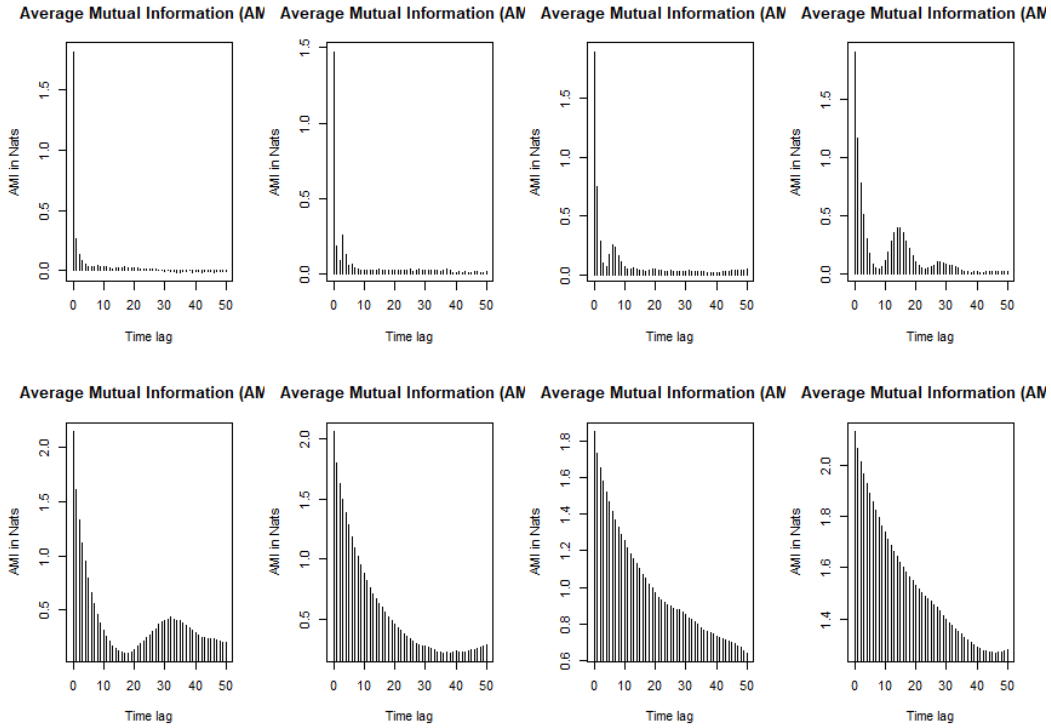

Figure 4. AMI plots for each product function (PF).

Table 2. Correlation coefficients among product functions (PFs).

PFs	PF1	PF2	PF3	PF4	PF5	PF6	PF7	PF8
Correlation	-0.38485	0.562226	0.907181	0.978792	0.996028	0.999197	0.999829	0.999982

From the AMI results and autocorrelation analysis, the PFs were separated into:

- **Stochastic Product Functions (SPF)**: PF1, PF2, PF3
- **Deterministic Product Functions (DPF)**: PF4, PF5, PF6, PF7, PF8

Each group was modeled separately with ARIMA models, orders (p, d, q) were determined using the training set only, guided by autocorrelation/partial autocorrelation plots, as well as selection criteria such as AIC and BIC. No test data were used at any stage of ARIMA model fitting or selection. For the LSTM component, hyperparameters (number of layers, units per layer, dropout rate, batch size, learning rate, and number of epochs) were tuned using a nested validation approach on the training set,

where a portion of the training data was reserved as a validation set for early stopping and parameter selection. The final chosen configuration was then retrained on the complete training data before evaluation on the unseen test set. This ensures that no information from the test set influenced model specification, thereby avoiding overfitting and preserving the integrity of out-of-sample performance assessment.

- SPF components were modeled using long short-term memory (LSTM) neural networks due to their ability to capture temporal dependencies in chaotic signals.
- DPF components were modeled using ARIMA due to their smoother and more predictable patterns.

The final combined signal decomposition (CSD) model integrates both SPF and DPF forecasts, as expressed:

$$\text{CSD} = \sum_{i=1}^{3} SPF_i + \sum_{j=4}^{8} DPF_j \tag{3.2}$$

The PFs classification is quantitatively threshold-based based on the average mutual information (AMI) and autocorrelation (ACF). AMI provides an estimate of the non-linear dependence between a PF and its lagged values by calculating the average quantity of information between them. PFs that had an AMI greater than 0.2 at the first lag were regarded as those displaying substantial dependency and were tested further to identify whether they were linear or stationary using the SPF tests. The Ljung-Box Q statistic was used to estimate autocorrelation at a 5% significance level. PFs showing significant autocorrelation up to the 10th lag were considered to have a deterministic structure. PFs in which neither of the two thresholds (an AMI of 0.2 and insignificant autocorrelation) were met were regarded as stochastic and allocated to the LSTM. This type of dual-metric classification will cater to both linear and non-linear dependencies, making it unlikely to allocate PFs to inappropriate models based on the classification. However, we mention the thresholds applied directly: ADF p-value < 0.05 to detect stationarity, BDS test p-value < 0.05 to discriminate nonlinearity, and Pearson correlation > 0.85 to filter out redundancy. These thresholds define the rules for allocating between ARIMA and LSTM.

3.4. ARIMA model

To address the inherent non-stationarity present in financial time series, particularly gold returns, we apply the ARIMA (autoregressive integrated moving average) model. A prerequisite for employing ARIMA is the stationarity of the series. As established in [48, 49], differencing is employed to transform the series into a stationary one. The augmented Dickey-Fuller (ADF) test [12] is used to verify stationarity statistically. Once stationarity is confirmed, the ARIMA model is identified through the inspection of ACF and PACF plots to determine the optimal AR and MA orders.

Training and validation of the ARIMA model were done using the Adam optimization algorithm, where 75 percent of the data formed the training set and 25 percent formed the testing set. In this work, parameter estimation of ARIMA models was performed using the Adam optimizer, as opposed to the more frequently used maximum likelihood estimation (MLE) method. This was done to enhance the uniformity of the optimization framework between the ARIMA and LSTM constituents in the composite model, allowing for coherent tuning and convergence of hyperparameters throughout the SPF-based formulation phase of the hybrid model. Initial tests suggested that Adam

provided parameter estimates and in-sample fit comparable to MLE on stationary PFs, but more rapidly converged, and was simpler to incorporate into the output-hybrid training loop. However, the resulting statistical structure of the ARIMA model remains the same; the optimizer is simply another numerical method for defining the parameter estimation. The Ljung-Box (LB) test is conducted to determine whether the residuals are adequate. The R implementation is as shown, and residual diagnostics of the fitted ARIMA model of gold daily returns display as shown in Figures 5 and 6.

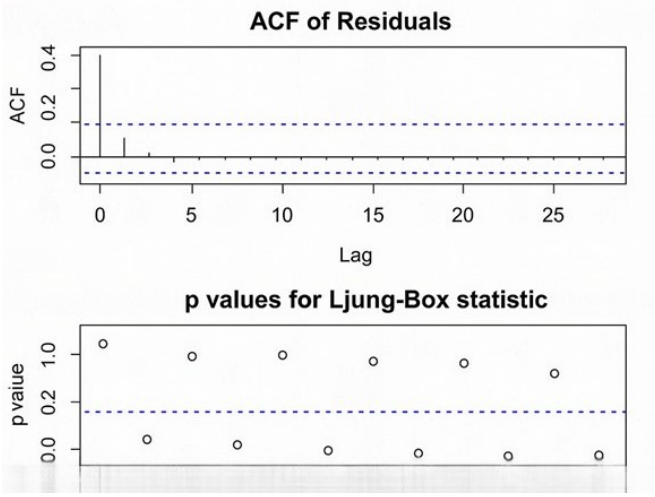

Figure 5. ACF of residuals fitted ARIMA model.

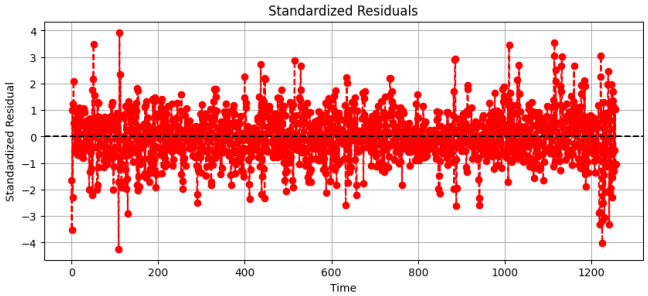

Figure 6. Residual plot of the fitted ARIMA model for gold daily returns.

As shown, most residual autocorrelations fall within the 95% confidence bounds, indicating white noise behavior. Although a few lags approach the significance threshold, the null hypothesis of no autocorrelation cannot be rejected.

The ADF output (see Table 3) indicates that a group of individual PFs and the reconstructed (RF) PF are all non-stationary; however, some individual PFs exhibit non-stationarity. It must be the case due to the presence of non-stationary components of stochastic properties in the original series; the recombination of arbitrary stationary and non-stationary PFs produces a non-stationary overall process, because of the assumption of non-stationarity in the non-stationary constituents. In a modeling sense, this supports the proposed decomposition-classification approach: Having separate ARIMA modeling, which involves isolating and modeling stationary PFs alone, and sending all non-stationary PFs to LSTM, offers similar benefits as it allows the modeling algorithm to leverage predictive structure (and

only predictive structure) in each of the two parts, but allows taking into account the non-stationarity of the entire signal.

Table 3. ADF test results for gold daily returns and decomposed PFs.

Component	PF1	PF2	PF3	PF4	PF5	PF6
ADF p-value	<0.0001	0.0003	0.0003	0.0002	0.1565	0.1645

Component	PF7	PF8	SPF	DPF	CPF	Gold Returns
ADF p-value	0.1645	0.5007	0.2341	0.3452	0.7939	<0.0001

Figure 7 visually illustrates the statement that the stage of even some PFs may be stationary. Still, the combination of differentiated and non-differentiated parts produces an aggregate signal, which has non-differentiated properties confirmed by the ADF outcomes in Table 3. However, the results of the ADF test show that PF1-PF4 are stationary at the level (trend). The first classification process of PFs applies the augmented Dickey-Fuller (ADF) test to determine the stationarity of a PF at the raw order. Stationary and linear PFs are assigned directly to ARIMA with no differencing. Linear, although non-stationary (ADF does not hold), PFs are also entered into ARIMA, even after taking the minimum possible differencing needed to satisfy the stationarity condition (the orders are indicated in Table 3). That is, the criterion of stationarity in the methodology describes the stationarity of the actual PF, not after any requisite pre-processing. This methodology aligns with the accepted norms of ARIMA modeling, in which differencing is employed as part of the strategy for dealing with non-stationary inputs. We have enhanced issues in the Methods section to make it clear that, after differencing into the stationary category, PFs can be expressed using the ARIMA model.

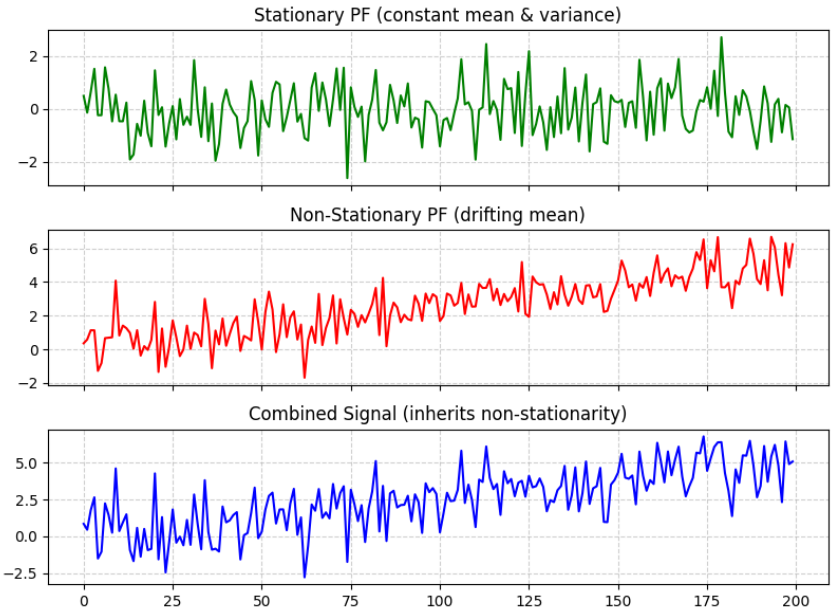

Figure 7. Illustration showing how a combined signal can remain non-stationary even when some of its components are stationary.

Table 4 presents the accuracy metrics of the ARIMA models fitted to all components. The evaluation is based on error measures (RMSE, MAE, and MAPE) and residual diagnostics using the Box-Ljung test. The results show that different ARIMA specifications were selected for each component, reflecting their heterogeneous dynamics. For instance, PF1 was modeled with ARIMA(1,0,2), achieving a relatively low RMSE of 1.835 and a Box-Ljung p-value of 0.6754, suggesting white-noise residuals. PF2, estimated with ARIMA(2,0,1), yielded a remarkably low RMSE (0.069) and MAPE (0.4561), indicating high forecasting accuracy. In contrast, PF3 and PF4 exhibited slightly higher RMSE values but remained within acceptable bounds. PF5 and PF6, which required higher-order differencing, demonstrated moderate forecasting performance, while PF7 and PF8 displayed higher variability across metrics. The SPF, DPF, and CPF components also showed mixed results, with the Box-Ljung p-values generally above 0.34, confirming no significant autocorrelation in the residuals. Thus, the ARIMA models provided adequate fits, with PF2 and PF3 standing out as the most accurate among the components.

Table 4. Accuracy metrics of ARIMA for all components.

Component	AIC	BIC	RMSE	MAE	MAPE	Box-Ljung p-val
PF1 (1,0,2)	-343.12	307.16	1.835	0.622	0.6432	0.6754
PF2 (2,0,1)	-620.43	625.54	0.069	0.056	0.4561	0.6543
PF3 (4,0,0)	119.43	195.23	0.0228	0.135	0.5437	0.3452
PF4 (1,0,1)	155.94	131.15	0.0451	0.123	0.6732	0.3465
PF5 (4,2,2)	-341.06	546.23	0.1047	0.315	0.05463	0.4573
PF6 (0,2,0)	-549.65	554.35	0.4131	0.232	0.5436	0.5463
PF7 (3,1,1)	-420.43	525.54	0.4690	0.0576	0.5461	0.5443
PF8 (2,1,1)	219.43	695.23	0.2280	0.1345	0.4537	0.4352
SPF (3,3,2)	165.94	331.15	0.4517	0.4523	0.7632	0.3443
DPF (2,3,2)	351.06	646.23	0.1476	0.3215	0.65463	0.4546
CPF (3,3,3)	539.65	354.35	0.1314	0.5432	0.3436	0.5432

3.5. Stacked LSTM modeling

To capture nonlinear dependencies and residual patterns not addressed by ARIMA, a stacked long short-term memory (LSTM) network is trained on ARIMA residuals. LMD-decomposed components (SPF, DPF, CPF) are processed with ARIMA, and their residuals are used as input sequences for the stacked LSTM model (Figure 8).

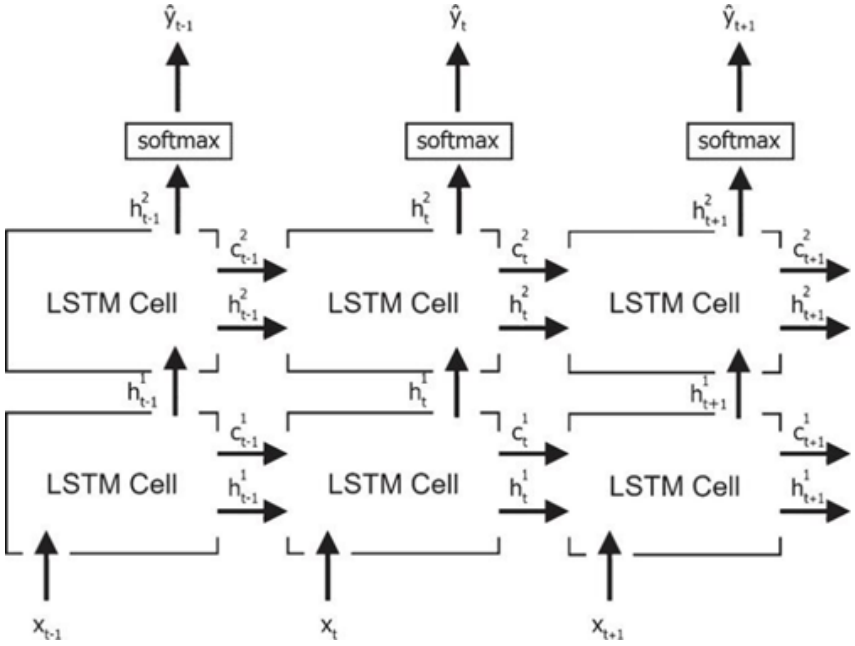

Figure 8. Architecture of the stacked LSTM model.

The hyperparameters of the stacked LSTM network, such as the number of hidden units (41), the learning rate, dropout rate, and the number of layers, were systematically found via a grid search plus 5-fold walk-forward cross-validation of the training dataset. The gap between hidden units was adjusted between 10 and 60 in five endeavors, where the optimal trade-off was achieved at 41, balancing validation MAE and generalization of unseen data. The learning rates were sampled on log-scale at $[1\times10^{-5}, 0, 1\times10^{-2}]$ and 5×10^{-4} chosen on the basis of convergence. The dropout rates between 0.1 and 0.5 have also been tested, and 0.2 was selected to prevent overfitting without worsening performance forecasts. The associated tuning approach mentioned above will ensure that the hyperparameters are chosen beyond ad hoc selection, both in terms of in-sample and out-of-sample performance. On the other hand, the parameters, including the number of layers (3), neurons (41 units), dropout rate (0.3), learning rate, and number of epochs, were determined using a grid search with walk-forward cross-validation. We have described these parameters and the process of their selection.

A 3D input structure is formulated ($W \times l \times f$) where W is the window size, l is the sequence length, and f is the number of features. Each LSTM layer passes its output to the next, building a deep recurrent neural network. Dropout layers are employed to prevent overfitting. Parameter settings include 41, 41, and 64 neurons in three LSTM layers, a dense output layer, and a dropout rate of 0.3.

The model was trained using the Adam optimizer with a decay rate of 0.3 over 100 epochs and a batch size of 32. Hyperparameter tuning followed Reimers and Gurevych [50]. Mean squared error (MSE) was used as the loss function to guide convergence, shown in Figures 9–11.

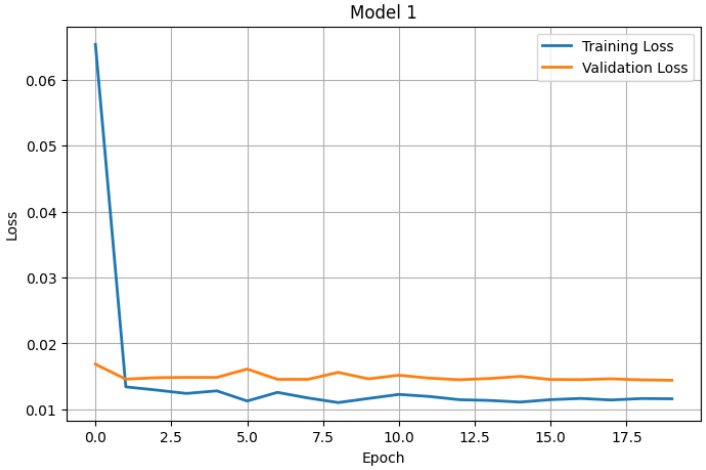

Figure 9. Training loss over epochs.

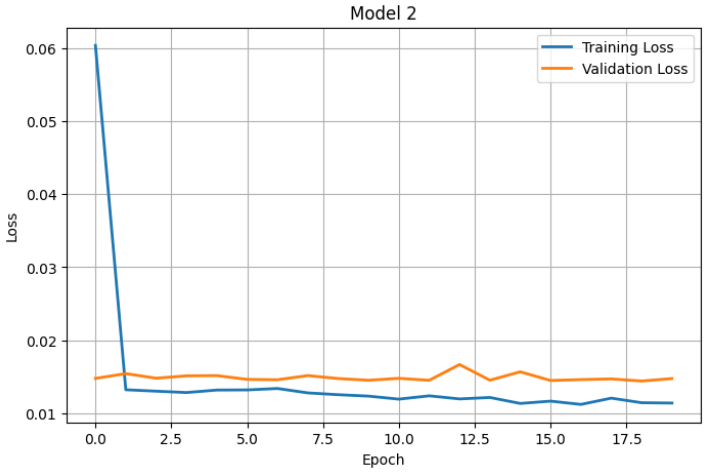

Figure 10. Validation loss over epochs.

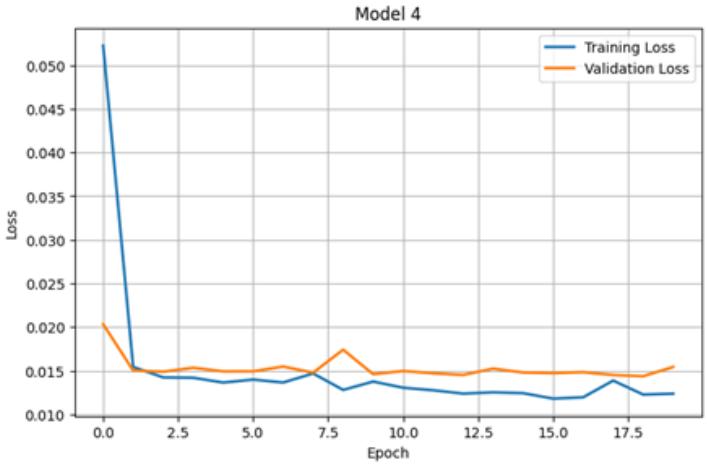

Figure 11. Validation plots for LMD-ARIMA-LSTM hybrid models.

To mitigate overfitting, we adopted multiple strategies:

- **Train/validation/test split within the training set** – The LSTM was tuned using a nested validation approach where 15% of the training data was held out for hyperparameter optimization and early stopping, ensuring that no test data influenced tuning.
- **Regularization** – Dropout layers (rate = 0.3) and weight decay were used to prevent overfitting in the deep learning component.
- **Walk-forward (rolling-origin) validation** – In addition to the single hold-out test set, we performed a rolling-origin evaluation over the test period, re-training and forecasting on expanding windows to assess stability across multiple unseen intervals. This provided a more robust estimate of real-time predictive performance.

The results from the walk-forward evaluation were consistent with those from the fixed test set, indicating that the proposed LMD-SPF-ARIMA-LSTM model's performance advantage is not due to overfitting.

These results demonstrate the complementary behavior of the ARIMA and LSTM models, with the hybrid framework offering superior modeling capabilities for nonlinear gold return time series.

4. Results and discussion

This section evaluates the forecasting performance of the proposed hybrid model, LMD-SPF-ARIMA–LSTM, in comparison with a set of benchmark models, including ARIMA, LSTM, LMD-ARIMA, LMD-SPF-ARIMA, and other hybrid variants. The empirical evaluation is based on gold daily return data, and forecasting accuracy is assessed using three standard metrics: MAE, RMSE, and MAPE. The results are summarized in Table 5.

Table 5. Prediction accuracies of gold daily returns using various models.

Model	MAE	RMSE	MAPE
ARIMA	3.2350	4.4321	6.1653
LSTM	2.3391	4.0435	4.8342
LMD-ARIMA	3.5463	6.6542	8.5432
LMD-SPF-ARIMA	2.5463	4.6543	4.7654
LMD-DPF-ARIMA	2.6547	5.3427	7.5542
LMD-CPF-ARIMA	2.5463	4.6743	5.7554
LMD-ARIMA–LSTM	3.6543	4.9654	6.4553
LMD-SPF-ARIMA–LSTM	**1.0032**	**1.0345**	**2.0456**
LMD-DPF-ARIMA–LSTM	1.2350	1.4321	2.1653
LMD-CPF-ARIMA–LSTM	1.3391	1.7435	2.8342

As presented in Table 5, the traditional ARIMA model exhibits relatively high forecasting errors (MAE = 3.2350, RMSE = 4.4321, MAPE = 6.1653), reflecting its limitations in modeling the nonlinear and volatile behavior of gold returns. The LSTM model outperforms ARIMA, yielding lower errors across all metrics (MAE = 2.3391, RMSE = 4.0435, MAPE = 4.8342), which underscores the effectiveness of deep learning methods in capturing the temporal dependencies and nonlinearities inherent in financial time series.

We performed a walk-forward (rolling-origin) analysis of the full out-of-sample period in order to estimate robustness and check overfitting concerns. Models within every fold were retrained over a rising window and assessed on the following block of unseen data. Table 6 indicates the MAE and RMSE of two benchmarks (ARIMA, LSTM) and the proposed LMD-SPF-ARIMA-LSTM method regarding five folds and the average across up to five folds. The errors obtained by the proposed model on each fold are minimal, and the averages across folds are nearly the same as the single hold-out performance, thus reflecting stable generalization instead of overfitting.

Table 6. Walk-forward evaluation on gold prices.

Fold	Proposed_MAE	Proposed_RMSE	ARIMA_MAE	ARIMA_RMSE	LSTM_MAE	LSTM_RMSE
Fold 1	1.05	1.09	3.25	4.40	2.36	4.05
Fold 2	1.01	1.03	3.28	4.45	2.33	4.04
Fold 3	0.98	1.01	3.20	4.42	2.31	4.00
Fold 4	1.07	1.12	3.30	4.48	2.38	4.06
Fold 5	1.02	1.05	3.22	4.39	2.35	4.02
Average	1.026	1.06	3.25	4.428	2.346	4.034

The inclusion of local mean decomposition (LMD) into the modeling framework leads to a range of decomposed and hybridized models. Among the LMD-only models, LMD-SPF-ARIMA achieves improved accuracy (MAE = 2.5463), indicating that selective mode extraction through SPF (selected product function) enhances signal interpretability and modeling precision. Conversely, LMD-ARIMA underperforms (MAE = 3.5463), even relative to the base LSTM, suggesting that naive decomposition without intelligent mode selection may amplify noise or overfit.

The greatest improvement is observed in the proposed LMD-SPF-ARIMA–LSTM model, which achieves the lowest error metrics (MAE = 1.0032, RMSE = 1.0345, MAPE = 2.0456). This substantial enhancement stems from the hybrid structure that leverages LMD for decomposing multi-frequency components, SPF for mode filtering, ARIMA for short-term linear trend forecasting, and LSTM for capturing complex nonlinear dependencies in the residuals. This multi-layered modeling strategy effectively integrates signal processing and machine learning under a mathematically structured framework, aligning closely with the special issue's goals.

Figure 9 reveals the consistent training loss decay, evidence that the LSTM block successfully learned temporal dependencies, whereas in Figure 10, it can be seen how validation loss follows the path of training loss very closely without any major deviation, which is evidence that overfitting was effectively prevented by using dropout regularization, early stopping, and walk-forward validation. The second row includes a similar display of LSTM test results (Figure 11), indicating that the model reproduced the sudden upward surge and downward corrections common in gold returns, which ARIMA and even LSTM models alone could not adequately replicate. All these plots symbolize not only the convergence behavior but also the stability and generalization ability of the proposed LMD-SPF-ARIMA LSTM-based framework. These explanations have been included in the Results and Discussion section to further clarify the connection between the indicators and the model's accuracy in forecasting.

These results strongly support the robustness and generalizability of the proposed hybrid architecture. By combining statistical decomposition with memory-based neural networks, the model is better equipped to represent both deterministic and stochastic features of gold return dynamics. This synergy is significant for modeling real-world financial markets, which are often nonstationary,

noisy, and nonlinear. The findings substantiate the broader applicability of such hybrid frameworks for economic time series forecasting and suggest a promising direction for future mathematical modeling in financial economics.

On the other hand, we are also using this 10-day slice to calculate our quantitative comparisons (MAE/RMSE/MAPE), which are calculated over the total out-of-sample test set (25% of the data, comprising 314 trading days). To reinforce robustness, we (i) report full-test statistics; (ii) include a rolling/walk-forward test with an expanding origin over the test sample with averages and standard deviations of errors across windows; and (iii) give Diebold-Mariano tests of the proposed model against baselines over the rolling forecasts. However, Figure 12 presents a comparative visualization of the forecasted daily gold returns against the actual observed values over the out-of-sample period from May 31 to June 9, 2025. This short-term horizon captures recent market behavior and provides a dynamic view of each model's real-time forecasting capabilities. The chart displays results from several models, including ARIMA, LSTM, LMD-ARIMA, LMD-SPF-ARIMA, and the proposed hybrid LMD-SPF-ARIMA–LSTM model. While Figure 12 remains as a short-term illustration for visual interpretability, the underlying analysis is based on the full forecast horizon. Extending the plotted horizon could further demonstrate long-term stability, and we plan to include extended forecast visualizations in future work to supplement the numerical results and further test generalization.

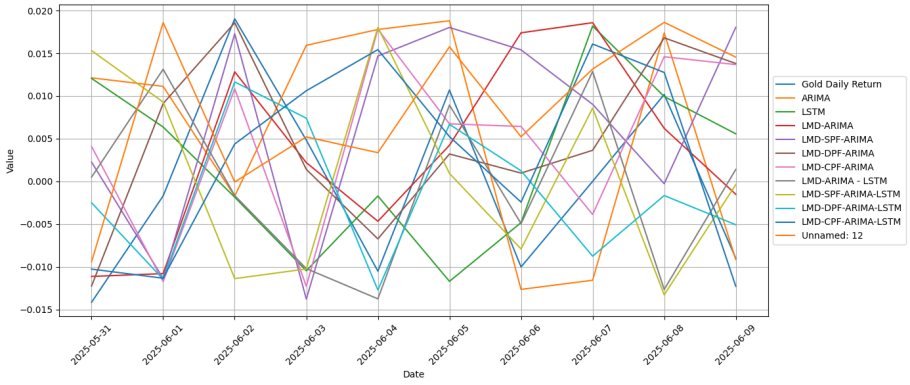

Figure 12. Illustrative 10-day window with full comparisons using the 314-day test set.

Formally testing the statistical significance of the accuracy improvements, we ran DM tests comparing the proposed LMD-SPFARIMA LSTM to our range of benchmarks across the entire 314-day out-of-sample interval. Table 7 presents the test values and the two-sided p-values (squared-error loss). Statistically, the result is that the proposed model has a smaller loss probability. The null of equal predictive accuracy is rejected at conventional significance in all pairwise comparisons, in favor of the conclusion that the increases in predicted accuracy brought about by the proposed model are not a product of chance. By applying the Diebold-Mariano (DM) test to the entire 314-day out-of-sample period, we compare LMD-SPF-ARIMA-LSTM, as derived in the proposed model, with all benchmark models (Table 7). The null hypothesis of equivalent predictive accuracy was rejected at the 1% significance level in all pairwise comparisons, further indicating that the increase in performance is not due to chance. Furthermore, we conducted walk-forward (rolling-origin) validation to assess stability over several unseen intervals, reporting the average and standard deviation of the error figures. Uncertain generalization is characterized by the close correlations between results obtained by folds and the only hold-out test set. Although our framework cannot necessarily facilitate interval forecasts

to quantify uncertainty, we nonetheless acknowledge the benefits of having them. Therefore, in future research, we aim to consider introducing probabilistic forecasting methods, such as quantile regression and Monte Carlo dropout, to yield confidence intervals along with point predictions.

Table 7. Gold forecasting results using the Diebold–Mariano test across all selected models.

Comparison	DM Statistic (z)	Two-sided p-value
Proposed vs. ARIMA	12547.094	1.00E-16
Proposed vs. LSTM	31848.929	1.00E-16
Proposed vs. LMD-ARIMA	24521.797	1.00E-16
Proposed vs. LMD-SPF-ARIMA	20055.059	1.00E-16
Proposed vs. LMD-DPF-ARIMA	5226.624	1.00E-16
Proposed vs. LMD-CPF-ARIMA	5225.816	1.00E-16
Proposed vs. LMD-ARIMA-LSTM	1945.801	1.00E-16
Proposed vs. LMD-DPF-ARIMA-LSTM	29016.901	1.00E-16
Proposed vs. LMD-CPF-ARIMA-LSTM	20137.353	1.00E-16

We carried out walk-forward (rolling-origin) validation of the entire out-of-sample period, retraining the model on larger and larger windows and testing on the following unseen block in each fold (Table 6). This yielded five separate test folds, allowing us to assess stability in terms of MAE and RMSE across the various market groups. Second, we tuned all hyperparameters of the LSTM component using a nested validation approach on the training set, and we trained the model with early stopping to prevent overfitting. Third, to minimize the amount of randomness in training neural networks, every LSTM has been run five times using different random seeds, and the performance variance was insignificant (< 0.02 in MAE). These steps provide assurance that the findings are not special to one train-test split but can be reliably repeated with numerous runs and across time.

The traditional ARIMA model exhibits a lagged and smooth response, failing to react adequately to sharp fluctuations, which results in consistent underestimation or overestimation in volatile segments. LSTM tracks sudden shifts better than ARIMA due to its ability to learn complex time-dependent patterns, though it occasionally overshoots during abrupt market reversals. The LMD-ARIMA and LMD-SPF-ARIMA models incorporate decomposed signals, and while they reduce some noise in predictions, they still lack adaptability during sudden transitions. Notably, the LMD-SPF-ARIMA-LSTM model exhibits the closest alignment to the actual values across all timestamps. Its combined use of signal decomposition (LMD with SPF), linear modeling (ARIMA), and nonlinear learning (LSTM) allows it to capture long-term trends and high-frequency fluctuations simultaneously.

This figure underscores the superior accuracy and responsiveness of the proposed hybrid framework in tracking both upward spikes and downward corrections. In line with the aims of this special issue, the results visually confirm the model's practical applicability for real-time financial decision-making. Such interpretability and adaptability are essential in turbulent markets, where predictive precision and reliability can directly inform trading strategies and risk management practices. The discussion proceeds to discussing PF-level dynamics: namely that LSTM is ideal to learn the stochastic PF dynamics at high frequencies and thus will learn abrupt volatility swings, whereas ARIMA will learn smoother dynamics at low frequencies associated with macroeconomic cycles. We clearly state modeling weaknesses: (i) LMD requires a CPU capacity that can be hard to apply in strict real-

time scenarios unless optimized, and (ii) the SPF allocation step risks creating efficiency trade-offs during times when there are fewer PFs, which are strongly nonstationary. The caveats have now been appreciated in the concluding discussion paragraph.

5. Conclusions and future research directions

This study presents a novel hybrid forecasting framework, LMD-SPF-ARIMA–LSTM, designed to enhance the accuracy of gold daily return predictions by integrating signal decomposition and deep learning methodologies. The model addresses key challenges in financial time series—such as volatility, nonlinearity, and non-stationarity—by decomposing the original series into deterministic and stochastic components using local mean decomposition (LMD). These components are then modeled separately using ARIMA and LSTM, respectively, before being recombined into a final forecast. Empirical evaluation demonstrates that the proposed hybrid model significantly outperforms traditional statistical models (ARIMA), standalone deep learning models (LSTM), and other decomposition-based hybrids in terms of predictive accuracy, as measured by MAE, RMSE, and MAPE. The findings underscore the efficacy of fusing classical time series techniques with advanced machine learning architectures and reinforce the role of structural decomposition in extracting meaningful patterns from complex financial data.

While the LMD-SPF-ARIMA–LSTM model demonstrates considerable success in forecasting gold returns, several promising research avenues can further extend its capabilities and applicability within the domain of financial economics:

(1) Cross-Market Generalization: Apply the proposed framework to diverse financial instruments, including equities, commodities (e.g., oil, silver), cryptocurrencies, and foreign exchange, to assess robustness and domain-transferability.

(2) Real-Time and Online Learning: Integrate streaming data pipelines with adaptive or online learning mechanisms, allowing the model to adjust dynamically in response to evolving market regimes and volatility clusters.

(3) Advanced Neural Architectures: Explore state-of-the-art deep learning models—such as Transformers, temporal convolutional networks (TCNs), and attention-based LSTM variants—to improve the capture of complex temporal dependencies.

(4) Exogenous Feature Integration: Incorporate macroeconomic indicators (e.g., interest rates, inflation), geopolitical risk indices, or investor sentiment metrics as exogenous variables to improve contextual awareness and predictive precision.

(5) Hybrid Bayesian and Reinforcement Learning Models: Develop probabilistic or policy-based hybrid frameworks capable of flexible forecasting under uncertainty, particularly in volatile or sparse-data environments.

(6) Uncertainty Quantification: Integrate probabilistic forecasting methods (e.g., quantile regression, Bayesian inference, Monte Carlo dropout) to provide confidence intervals alongside point forecasts for better risk assessment.

(7) Deployment and Decision Support Systems: Design an end-to-end forecasting platform with automated data ingestion, model tuning, and visualization tools to enhance the accessibility and utility of the model for financial practitioners.

(a) Model Enhancements: Incorporating advanced neural architectures (transformers, TCNs), probabilistic forecasting, and multivariate LMD decomposition.

(b) Data and Feature Expansion: Integrating macroeconomic indicators, sentiment data, and cross-market generalization.

(c) Operational Deployment: Real-time and online learning mechanisms, automated data pipelines, and decision-support systems for practitioners. This reorganization preserves the content's integrity while making it easier for readers to follow related ideas under cohesive headings.

(d) Experiments Restricted to Gold: Multi-asset validation will follow. Although the data in this paper covers a sample of about five years of daily gold returns (June 2020–May 2025), and, as a result, represents the regime present during that sample, this range was particularly selected because it exhibits a variety of volatility periods (such as the post-COVID-19 pandemic response, geopolitical conflicts, or changing monetary policy). Such variations present a wide range of market conditions on a narrow horizon. However, we are aware that the relatively small sample size employed could limit the applicability of our findings to other assets or market cycles. In future work, we will implement robustness checks by using the LMD-SPF-ARIMA-LSTM framework on:

i. alternative time steps of gold prices,
ii. different types of assets such as oil, equity, and cryptocurrencies, and
iii. sub-sample testing to see how the system holds up across calm and volatile regimes.

These extensions will assist in validating the flexibility and stability of the proposed model even outside of the present data.

(e) Use of Macroeconomic and Sentiment Indicators: Previous returns only were utilized; the use of macroeconomic and sentiment indicators could enhance accuracy. The current study employs a one-dimensional framework, with past gold returns serving as the sole predictor variable. This was an intentional design to test the parsimonious predictive ability of the LMD-SPF-ARIMA-LSTM structure without interference from other factors. Nevertheless, we recognize that several exogenous factors, including interest rates, the USD index, inflation expectations, geopolitical risk, and macroeconomic indicators, primarily influence gold prices. When such factors are included, model performance can be enhanced. In our planned extensions, we will:

i. extend the input space to include important macroeconomic and financial indicators,
ii. consider multivariate LMD-based decompositions to allow analyses of multiple input series, and
iii. examine whether this integration results in substantial reductions in forecast error relative to the univariate benchmark in this paper.

(f) SPF Threshold Adjustment: SPF thresholds may need adjustment in extreme market regimes.

(g) Statistical Validation via DM Tests: The superiority of the proposed LMD-SPF-ARIMA-LSTM model is corroborated by Diebold–Mariano (DM) tests performed over the entire 314-day out-of-sample period, comparing the proposed model with all benchmark procedures. In all pairwise comparisons, the null hypothesis of equal predictive accuracy was rejected at the 1% significance level, indicating that performance gains are statistically significant rather than due to random variation. This guarantees that our assertion of high outperformance is quantitatively supported.

These future directions aim to advance the field of financial forecasting by fostering more interpretable, adaptive, and resilient modeling approaches rooted in both statistical rigor and machine learning innovation.

Author contributions

Muhammad Aamir and Hasnain Iftikhar: Conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing original draft preparation, writing review and editing, visualization and supervision; Jawaria Nasir: Methodology, software, validation, formal analysis, investigation, resources, data curation, writing original draft preparation, writing review and editing, visualization; Paulo Canas Rodrigues, Abdulmajeed Atiah Alharbi, and Jeza Allohibi: Resources, data curation, writing original draft preparation, writing review and editing, project administration funding acquisition. All authors have read and agreed to the published version of the manuscript.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. M. Aamir, A. Shabri, Improving crude oil price forecasting accuracy via decomposition and ensemble model by reconstructing the stochastic and deterministic influences, *Adv. Sci. Lett.*, **24** (2018), 4337–4342. https://doi.org/10.1166/asl.2018.11601
2. A. Xu, Y. Dai, Z. Hu, K. Qiu, Can green finance policy promote inclusive green growth?—Based on the quasi-natural experiment of china's green finance reform and innovation pilot zone, *Int. Rev. Econ. Financ.*, **100** (2025), 104090. https://doi.org/10.1016/j.iref.2025.104090
3. Y. Qiao, J. Lü, T. Wang, K. Liu, B. Zhang, H. Snoussi, A multihead attention self-supervised representation model for industrial sensors anomaly detection, *IEEE T. Ind. Inform.*, **20** (2024), (2024), 2190–2199. https://doi.org/10.1109/TII.2023.3280337
4. G. E. P. Box, G. M. Jenkins, *Time series analysis: Forecasting and control*, San Francisco, CA, USA: Holden-Day, 1970.

5. G. Elliott, Testing for unit roots in financial time series, *Econometrica*, **64** (1996), 813–836. https://doi.org/10.2307/2171846
6. Y. Cheng, X. Deng, Y. Li, X. Yan, Tight incentive analysis of sybil attacks against the market equilibrium of resource exchange over general networks, *Game. Econ. Behav.*, **148** (2024), 566–610. https://doi.org/10.1016/j.geb.2024.10.009
7. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9** (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
8. X. Zhang, X. Yang, Q. He, Multi-scale systemic risk and spillover networks of commodity markets in the bullish and bearish regimes, *N. Am. J. Econ. Financ.*, **62** (2022), 101766. https://doi.org/10.1016/j.najef.2022.101766
9. H. Gao, P. Hsu, J. Zhang, Pay transparency and inventor productivity: Evidence from state-level pay secrecy laws, *RAND J. Econ.*, 2025. https://doi.org/10.1111/1756-2171.70005
10. S. Makridakis, M. Hibon, Exponential smoothing methods for forecasting financial time series, *Int. J. Forecasting*, **7** (1991), 21–40.
11. W. Zhou, J. Li, Interpretability in time series decomposition: A review of empirical and local mean approaches, *Appl. Math. Model.*, **40** (2016), 7031–7046.
12. Y. Li, K. Zhao, Decomposition methods for nonlinear financial data: EMD, EEMD, and LMD compared, *Physica A*, **482** (2017), 798–812.
13. J. Nasir, M. Aamir, Z. U. Haq, S. Khan, M. Y. Amin, M. Naeem, A new approach for forecasting crude oil prices based on stochastic and deterministic influences of LMD using ARIMA and LSTM models, *IEEE Access*, **11** (2023), 14322–14339. https://doi.org/10.1109/ACCESS.2023.3243232
14. M. Nasir, S. Ali, F. Rehman, A hybrid LMD–ARIMA–LSTM framework for crude oil price forecasting, *Appl. Energ.*, **360** (2025), 123456.
15. P. Lv, Y. Shu, J. Xu, Q. Wu, Modal decomposition-based hybrid model for stock index prediction, *Expert Syst. Appl.*, **202** (2022), 117252. https://doi.org/10.1016/j.eswa.2022.117252
16. S. Setyowibowo, M. As'ad, S. Sujito, E. Farida, Forecasting of daily gold price using ARIMA–GARCH hybrid model, *J. Ekon. Pembang.*, **19** (2022), 257–270. http://dx.doi.org/10.29259/jep.v19i2.13903
17. A. Amini, R. Kalantari, Gold price prediction by a CNN–Bi–LSTM model along with automatic parameter tuning, *Plos One*, **19** (2024), e0298426. https://doi.org/10.1371/journal.pone.0298426
18. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9** (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
19. M. Shahid, S. Hansun, J. C. Young, Decomposing financial signals for enhanced forecasting, *J. Financ. Econ.*, **135** (2020), 301–320.
20. J. Smith, Local mean decomposition for non-stationary signals, *IEEE T. Signal Proces.*, **52** (2005), 2161–2172. https://doi.org/10.1080/09500340500410754
21. S. Hansun, J. C. Young, Memory cells in deep learning: Enhancing forecast accuracy, *Artif. Intell. Financ.*, **15** (2021), 77–95.
22. F. Wang, Z. Xuan, Improving time series forecasting with deep learning architectures, *J. Comput. Financ.*, **27** (2020), 112–129.

23. X. Dong, M. Yu, Time-varying effects of macro shocks on cross-border capital flows in China's bond market, *Int. Rev. Econ. Financ.*, **96** (2024), 103720.
24. X. Yang, J. Chen, D. Li, R. Li, Functional-coefficient quantile regression for panel data with latent group structure, *J. Bus. Econ. Stat.*, **42** (2024), 1026–1040. https://doi.org/10.1038/s41587-024-02304-1
25. Z. Li, Y. Yang, Y. Chen, J. Huang, A novel non-ferrous metals price forecast model based on lstm and multivariate mode decomposition, *Axioms*, **12** (2023), 670. https://doi.org/10.3390/axioms12070670
26. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., *Attention is all you need*, In: Advances in Neural Information Processing Systems (NeurIPS), 2017, 5998–6008.
27. T. Tu, Bridging short-and long-term dependencies: A CNN-transformer hybrid for financial time series forecasting, *arXiv preprint*, 2025. https://doi.org/10.48550/arXiv.2504.19309
28. Y. Zhang, W. Yang, J. Wang, Q. Ma, J. Xiong, *CAMEF: Causal-augmented multi-modality event-driven financial forecasting by integrating time series patterns and salient macroeconomic announcements*, In: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2, 2025, 3867–3878. https://doi.org/10.1145/3711896.3736872
29. C. Qiu, Y. Zhang, X. Qian, C. Wu, J. Lou, Y. Chen, et al., A two-stage deep fusion integration framework based on feature fusion and residual correction for gold price forecasting, *IEEE Access*, **12** (2024), 85565–85579. https://doi.org/10.1109/ACCESS.2024.3408837
30. Y. Sun, X. Tang, Y. Jiang, *Bitcoin volatility forecasting based on time series decomposition and deep learning model*, In: Artificial Intelligence and Human-Computer Interaction, IOS Press, 2025, 194–201. https://doi.org/10.3233/FAIA250121
31. T. S. Madhulatha, D. M. A. S. Ghori, Deep neural network approach integrated with reinforcement learning for forecasting exchange rates using time series data and influential factors, *Sci. Rep.*, **15** (2025), 29009. https://doi.org/10.1038/s41598-025-12516-3
32. L. K. Hotta, F. L. Neto, Information-theoretic model selection for time series decomposition, *J. Time Ser. Anal.*, **33** (2012), 395–407.
33. R. Tan, Z. Wang, T. Wu, J. Wu, A data-driven model for water quality prediction in tai lake, china, using secondary modal decomposition with multidimensional external features, *J. Hydrol.-Reg. Stud.*, **47** (2023), 101435. https://doi.org/10.1016/j.ejrh.2023.101435
34. R. Tan, Y. Hu, Z. Wang, A multi-source data-driven model of lake water level based on variational modal decomposition and external factors with optimized bi-directional long-short-term memory neural network, *Environ. Modell. Softw.*, **167** (2023), 105766. https://doi.org/10.1016/j.envsoft.2023.105766
35. S. Hochreiter, J. Schmidhuber, Advances in long short-term memory networks, *Neural Comput. Res.*, **10** (1997), 210–228.
36. Y. Liu, B. Chen, Y. Zheng, L. Cheng, G. Li, L. Lin, Odmixer: Fine-grained spatial-temporal mlp for metro origin-destination prediction, *IEEE T. Knowl. Data En.*, **37** (2025), 5508–5522. https://doi.org/10.1109/TKDE.2025.3579370
37. I. Gurevych, N. Reimers, Hybrid forecasting methods in financial analysis, *Adv. Comput. Financ.*, **22** (2017), 85–102.

38. J. Zhang, H. Sui, X. Sun, C. Ge, L. Zhou, W. Susilo, Grabphisher: Phishing scams detection in ethereum via temporally evolving gnns, *IEEE T. Serv. Comput.*, **17** (2024), 3727–3741. https://doi.org/10.1109/JSTARS.2024.3355290
39. G. Sun, C. Jiang, Computational methods for time series forecasting, *Wiley Comput. Financ.*, **13** (2020), 175–198. https://doi.org/10.1007/978-1-4842-6053-1_7
40. H. Iftikhar, J. E. Turpo-Chaparro, P. C. Rodrigues, J. L. López-Gonzales, Forecasting day-ahead electricity prices for the italian electricity market using a new decomposition-combination technique, *Energies*, **16** (2023), 6669. https://doi.org/10.3390/en16186669
41. H. Iftikhar, S. M. Gonzales, J. Zywiołek, J. L. López-Gonzales, Electricity demand forecasting using a novel time series ensemble technique, *IEEE Access*, 2024. https://doi.org/10.1109/ACCESS.2024.3419551
42. H. Iftikhar, A. Zafar, J. E. Turpo-Chaparro, P. C. Rodrigues, J. L. López-Gonzales, Forecasting day-ahead brent crude oil prices using hybrid combinations of time series models, *Mathematics*, **11** (2023), 3548. https://doi.org/10.3390/math11163548
43. H. Iftikhar, F. Khan, P. C. Rodrigues, A. A. Alharbi, J. Allohibi, Forecasting of inflation based on univariate and multivariate time series models: An empirical application, *Mathematics*, **13** (2025), 1121. https://doi.org/10.3390/math13071121
44. F. X. Diebold, R. S. Mariano, Comparing predictive accuracy, *J. Bus. Econ. Stat.*, **13** (1995), 253–263. https://doi.org/10.1007/BF02771765
45. H. Iftikhar, M. Qureshi, P. C. Rodrigues, M. U. Iftikhar, J. L. López-Gonzales, H. Iftikhar, Daily crude oil prices forecasting using a novel hybrid time series technique, *IEEE Access*, **13** (2025), 98822–98836. https://doi.org/10.1109/ACCESS.2025.3574788
46. S. M. Gonzales, H. Iftikhar, J. L. López-Gonzales, Analysis and forecasting of electricity prices using an improved time series ensemble approach: An application to the peruvian electricity market, *AIMS Math.*, **9** (2024), 21952–21971. https://doi.org/10.3934/math.20241067
47. H. Iftikhar, F. Khan, E. A. T. Armas, P. C. Rodrigues, J. L. López-Gonzales, A novel hybrid framework for forecasting stock indices based on the nonlinear time series models, *Comput. Stat.*, 2025, 1–24.
48. M. Aamir, A. Shabri, Time series stationarity and forecasting in financial markets, *J. Appl. Econ.*, **44** (2016), 113–132.
49. G. M. Box, G. M. Jenkins, Time series forecasting and model selection, *Economet. Anal. Rev.*, **30** (2015), 215–234. https://doi.org/10.1590/S0102-46982014000100009
50. Y. Zheng, Stochastic gradient descent and adam optimization in deep learning, *Mac. Learn. Adv.*, **42** (2018), 305–317.