

Explainable and Responsible AI for Deep Learning-Based NIFTY 50 Stock Price Prediction Using SHAP and Technical Indicators

Amit Kumar Verma^{1 a)}, Himanshu Vasnani^{2 b)}, Amit Tiwari^{2 c)}, Deep Shikha^{3 d)}

^{1 a)} Faculty of Sciences (Computer Science), Suresh Gyan Vihar University, Jaipur, India

^{2 b,c)} Department of Mechanical Engineering, Suresh Gyan Vihar University, Jaipur, India

^{3 d)} Arrivo EDU Private Limited, Ahmedabad, Gujarat, India

Author Emails

^{a)} amitverma.researchwork@gmail.com, ^{b)} himanshuvasnani1@gmail.com, ^{c)} amittiwari992@gmail.com, ^{d)} deepshikha.research24@gmail.com

Abstract — Stock market prediction using deep learning has come a long way, but one problem keeps showing up in practice: these models give a prediction without explaining why. A trader looking at a GRU output that says "buy tomorrow" has no way of knowing whether the signal came from the RSI crossing a threshold, the price breaking above the 50-day moving average, or something entirely different buried deep inside the network. This is the black-box problem, and it is the reason why many traders still prefer simple rule-based charts over sophisticated neural networks. This paper directly tackles that problem. We trained five deep learning architectures — LSTM, GRU, Bidirectional LSTM, CNN-LSTM, and Transformer — on five years of NIFTY 50 daily price data (January 2021 to December 2025), using 41 technical indicators as inputs covering trend, momentum, volatility, volume, and price-lag domains. After comparing all five models on eight performance metrics including directional accuracy and RMSE, we identified GRU as the best performer with an RMSE of 544.69 INR and a directional accuracy of 56.82%. We then applied SHAP (SHapley Additive exPlanations) using the GradientExplainer method to open the GRU's black box completely. The results showed that the 50-day Simple Moving Average is the single most important indicator, and that three indicator families — Trend (33.61%), Lag (29.24%), and Volatility (28.78%) — together account for over 91% of what drives the model's predictions. Most importantly, we generated day-specific waterfall explanations for actual BUY and SELL trading days, showing precisely which indicators pushed the prediction up or down on each day. Any trader can read these charts without needing to understand neural networks. We believe this kind of transparency is what makes AI trustworthy enough to use in real financial decisions.

Keywords: Explainable AI, Responsible AI, SHAP, Black-Box Problem, Deep Learning, GRU, LSTM, Transformer, NIFTY 50, Technical Indicators, Stock Market Prediction, Feature Attribution, Transparency.

I. INTRODUCTION

Ask any retail trader in India why they don't fully trust AI-based stock prediction tools and the answer is almost always the same: "I don't know how it makes its decision." That hesitation is reasonable. Over the past few years, deep learning models — especially recurrent types like LSTM and GRU — have shown genuinely impressive accuracy on equity price forecasting tasks. Researchers have shown these models can outperform traditional approaches like ARIMA on short-horizon prediction. But here is the catch. When a GRU model says NIFTY 50 will go up tomorrow, it cannot tell you whether that conclusion is based on a moving average crossover, a Bollinger Band level, recent volume trends, or some combination of dozens of factors. The model just produces a number. For institutional traders with risk committees and compliance teams, this is already a problem. For retail traders and general investors, it is an even bigger one.

This lack of explanation has a name in the research community: the black-box problem. It is not just a philosophical concern. In practice, it creates three

concrete difficulties. The first is that traders cannot validate the signal. If you get a prediction but no reason, you have no basis for judging whether it makes sense. The second is accountability. If the model loses money, there is no way to trace back what went wrong and fix it. The third, which may be the most important from a practical standpoint, is trust. Technical traders spend years learning to read charts and indicators. They have their own views on what RSI levels mean or where support lies on the 50-day average. A black-box prediction that contradicts their reading — with no explanation — is almost always going to be ignored.

The good news is that a method called SHAP (SHapley Additive exPlanations) can solve this problem in a principled way. SHAP was developed by Lundberg and Lee^[5] and is built on a branch of game theory originally developed by Shapley^[15]. The core idea is simple: treat each input feature as a player in a game, and calculate how much each player contributed to the final outcome. SHAP does this while satisfying three important mathematical properties — it is locally accurate (contributions add up to the actual prediction), it is consistent (removing

a feature that genuinely helps will never increase its attributed importance), and it correctly handles missing features. When applied to a trained neural network through the GradientExplainer method, SHAP produces a number for each technical indicator on each day that tells you, in the same units as the prediction itself, how much that indicator pushed the price forecast higher or lower.

This paper applies that approach to the NIFTY 50 prediction problem. We chose the NIFTY 50 specifically because it is India's most widely tracked benchmark — it covers 50 of the largest companies on the NSE and underlies a huge volume of futures and options activity. Five deep learning architectures are trained and compared: LSTM, GRU, Bidirectional LSTM, CNN-LSTM, and the Transformer. All five use the same 41 technical indicators as inputs, built from the same five-year dataset (2021–2025). GRU came out on top across all metrics. We then applied SHAP to GRU's trained weights and generated explanations at three levels: which indicators matter most overall, which indicator families the model leans on, and what exactly drove a specific BUY or SELL call on a specific trading day.

The paper is laid out as follows. Section II reviews relevant prior work. Section III covers the data, features, model designs, and how we measure performance. Section IV shows the model comparison results. Section V presents the full SHAP analysis. Section VI provides the BUY and SELL day explanations. Section VII discusses what it all means. Section VIII concludes.

II. LITERATURE REVIEW

A. Why Black-Box AI Fails in Finance

Lipton^[17] made the case clearly: interpretability in machine learning is not one thing. It includes being able to simulate the model mentally, being able to decompose its reasoning step by step, and being able to inspect the algorithm itself. Standard deep learning satisfies none of these conditions at the point of making a prediction. In financial settings this matters more than in most other domains because the people acting on predictions are not passive recipients — they are professionals and individuals with their own views, their own risk limits, and real money at stake. Ribeiro, Singh, and Guestrin^[18] proposed LIME as a way to explain any model locally by fitting a simple linear approximation near each prediction, but linear approximations can mislead when the true relationship is sharply nonlinear, as it often is with financial data. Goodfellow, Bengio, and Courville^[8] provide the theoretical basis for understanding why deep networks resist straightforward interpretation: the features learned at each layer are distributed and interleaved in ways that do not correspond neatly to individual inputs.

B. Deep Learning Applied to Stock Prediction

The work of Fischer and Krauss^[6] remains one of the most cited demonstrations that LSTM can generate statistically significant returns on a real equity universe (S&P 500). Their setup — training on historical data and testing out-of-sample in a long-short framework — established a template that many later studies followed. The original LSTM paper by Hochreiter and Schmidhuber^[1] solved a fundamental problem with earlier recurrent networks: gradients vanished or exploded during backpropagation through time, making it impossible to learn from sequences longer than a few steps. LSTM's gating mechanism — input gate, forget gate, output gate — gave the network control over what it remembered and what it forgot. Cho and colleagues^[2] later showed that two gates (update and reset) were sufficient, producing the GRU, which trained faster and often generalised better when data was limited. The Transformer introduced by Vaswani et al.^[4] eliminated recurrence entirely and instead used self-attention to model relationships between all time steps at once. It has worked brilliantly in natural language processing, but financial sequences are much shorter and noisier, and several studies have found that the Transformer needs far more data than is usually available from a single equity index to match simpler recurrent models. Bao, Yue, and Rao^[11] and Selvin et al.^[12] contributed early evidence on combining convolutional and recurrent layers for stock prediction, showing that local pattern detection via convolution can complement the temporal integration performed by LSTM.

C. SHAP and Explainability in Financial AI

The paper by Lundberg and Lee^[5] that introduced SHAP is notable not just for the method but for the theoretical argument behind it. They proved that SHAP is the only attribution method that satisfies three conditions simultaneously: predictions decompose exactly into feature contributions, the attribution is consistent with the direction of each feature's effect, and features that genuinely contribute nothing receive zero attribution. Earlier methods like saliency maps and DeepLIFT satisfied subsets of these conditions. The Shapley value concept itself comes from Shapley's^[15] 1953 work on cooperative games, and SHAP is the first time it was made computationally tractable for modern machine learning models at scale. In finance, SHAP has mostly been used with gradient boosting models for credit scoring and fraud detection — settings where the inputs are already tabular. Applying it to deep learning models on sequential price data requires the GradientExplainer variant, which backpropagates attribution through the network's gradient information. This is relatively new in financial research, and we are not aware of a prior study that applies GradientExplainer to a recurrent DL model trained on Indian equity index data with a full suite of technical indicators. For measuring how good predictions are in

portfolio management terms, Grinold and Kahn^[14] established the Information Coefficient — the rank correlation between predicted and realised returns — as the standard measure of genuine predictive skill.

D. Technical Indicator Design

Murphy^[10] and Achelis^[20] are the two standard references for how technical indicators are constructed and what they measure. Murphy organises them into four families — trend, momentum, volatility, and volume — and explains the logic of each. We followed this taxonomy when grouping our 41 features for the SHAP category analysis, so the results can be interpreted directly in the language that technical traders already use. For the data pipeline — how to split time series without leaking information from the future, how to scale features, how to build sliding windows — we followed best-practice guidance for deep learning on financial time series, including the sliding-window preprocessing and MinMaxScaler approaches that have become standard in this field^[6,11,12].

III. METHODOLOGY

A. Data and Study Period

We downloaded daily OHLCV data for the NIFTY 50 index (Yahoo Finance ticker: ^NSEI) from 1 January 2021 to 31 December 2025 using Python's yfinance library^[19]. After dropping exchange holidays and any rows with missing values, we had 1,233 trading days. The reason for choosing this particular five-year window is that it contains meaningfully different market conditions: 2021 was a strong recovery year after the pandemic sell-off, 2022 brought significant volatility due to geopolitical tensions and central bank rate hikes, and 2023 through 2025 saw NIFTY 50 mostly trending upward with occasional corrections. Having all three types of market behaviour in the dataset gives the models a harder test than using a single trending or single ranging period would. The task we set for each model is to predict the next-day closing price — that is, given what happened today and on the 59 days before it, what will the closing price be tomorrow.

B. Feature Engineering — 41 Technical Indicators

We built 41 technical indicators using the ta Python library, covering six domains. Everything was computed on training data only, never using future data during the calibration step. Here is what each group contributes.

1) Trend Indicators (16 features)

Simple Moving Averages for 10, 20, and 50-day windows; Exponential Moving Averages at 9, 12, and 26 days; the difference between EMA-12 and EMA-26 as a crossover signal; the full MACD suite (line, signal, histogram, and the rate of change of the histogram); ADX with its positive and negative

directional components; and the Aroon up/down oscillator at 25 periods. These tell the model what direction and strength the market trend has at different time horizons.

2) Momentum Indicators (8 features)

RSI at 14 periods and a 5-day measure of how fast RSI itself is moving; Stochastic %K and %D; Williams %R; CCI at 20 periods; Rate of Change at 10 periods; and the True Strength Index. Together these capture how fast prices are moving and whether the market is stretched relative to recent norms.

3) Volatility Indicators (10 features)

Bollinger Bands give five features: upper band, lower band, percentage position within the band, bandwidth, and a squeeze indicator. Average True Range measures day-to-day price range. Keltner Channels provide upper and lower boundaries based on ATR. Donchian Channels capture the highest high and lowest low over 20 days. These collectively tell the model how much price is moving and where the current price sits within its recent range.

4) Volume, Derived and Lag Features (17 features)

On-Balance Volume, Chaikin Money Flow, and Money Flow Index capture buying and selling pressure from trading volumes. The derived group includes daily return, log return, the ratio of daily range to close, the close-to-open gap, 20-day annualised volatility, and 20-day skewness and kurtosis of returns, plus how far price is sitting above or below the 20-day and 50-day averages. Finally, we added closing prices and returns at lags of 1, 2, 3, 5, 10, and 20 trading days so the model can directly see its own recent history.

C. Data Split and Preprocessing

We used an 80/20 chronological split: the first 986 trading days for training, the final 247 days (all of 2025) for testing. We never shuffled the data — doing so would mix future information into the training set and produce results that cannot hold up in real trading. For the deep learning models, we fitted two separate MinMaxScalers on the training data alone: one for the 41 input features and one for the target price. Separating them avoids a subtle inverse-transform bug where using a combined scaler with a different number of columns fails silently and corrupts predictions. Each training and test sample consists of 60 consecutive trading days of feature data (a rolling window), and the model tries to predict the price on day 61. This gives around 926 training sequences and 187 test sequences.

D. The Five Deep Learning Architectures

All five models share the same training setup: Huber loss, Adam optimiser, early stopping after 20 epochs of no improvement, learning rate reduction by a factor of 0.35 when stuck, and a maximum of 120 epochs with batch size 32. The LSTM^[1] uses three stacked recurrent layers with 256, 128, and 64 units

respectively, dropout rates of 0.20, 0.15, and 0.10 after each layer, and two dense layers before the output. The forget gate variant of LSTM ^[7] is used, which has shown better performance on non-stationary sequences. The GRU ^[2] has the same overall structure but uses the simpler two-gate cell — update gate and reset gate — instead of LSTM's three gates. The Bi-LSTM ^[3] runs the LSTM in both forward and backward directions through the 60-day window and concatenates the results, so it sees context from both ends of the sequence. The CNN-LSTM ^[12] first passes the data through two convolutional layers with max-pooling to extract local patterns, then feeds the result into two LSTM layers to capture longer-term dependencies. The Transformer ^[4] uses two encoder blocks, each with four-head self-attention and a feed-forward network with GELU activation, followed by global average pooling. All seeds were fixed (Python, NumPy, TensorFlow) and layer-level initializers were seeded to ensure repeatable results.

E. How We Measured Performance

We computed eight metrics on the test set for each model. RMSE and MAE measure absolute prediction error in INR. MAPE expresses error as a percentage of the actual price. R² tells us how much of the price variance the model explains. Directional Accuracy (DA%) counts what fraction of test days the model got the direction right — up or down. This is the metric that matters most for trading because a directional signal is what drives buy or sell decisions. Theil's U₂ ^[13] compares the model's RMSE to the simplest possible forecast: assume the price doesn't change. If U₂ is below 1, the model beats the do-nothing baseline. The Spearman Information Coefficient ^[14] measures rank correlation between predicted and actual price changes, widely used in quantitative portfolio management as a measure of genuine predictive skill. We combined all eight into a Composite Score with weights: DA% gets 30%, RMSE and R² each get 20%, MAPE gets 15%, Theil's U₂ gets 10%, and IC gets 5%. For statistical significance testing, we used the Diebold-Mariano test ^[9] and also verified pairwise results with the Wilcoxon signed-rank test ^[16] as a non-parametric complement.

F. How We Applied SHAP

We applied SHAP only to the GRU — the top-ranked model. The method we used is called GradientExplainer, which works by backpropagating attribution scores through the network's gradient computations. We gave it 150 training sequences as a background distribution (so it has a reference point for what 'average' looks like) and then computed explanations for 100 test sequences. Each explanation is a 3-dimensional array of values — one per time step per feature per prediction. We averaged across the 60 time steps to get a single attribution per feature

per prediction day. The absolute mean of that over 100 predictions gives the global importance score for each indicator. The signed values (positive or negative) are kept for the beeswarm and waterfall charts so traders can see whether an indicator was pushing the price forecast up or down on a given day.

IV. RESULTS AND ANALYSIS

A. How the Five Models Performed

Table 1 shows the results across all eight metrics. Fig. 1 gives a visual overview of all 41 indicators, and Fig. 2 shows the ranking by Composite Score and Directional Accuracy.

TABLE 1. DL MODEL PERFORMANCE — NIFTY 50 TEST SET (2025). ★ = BEST

Model	RM SE (↓)	MA E (↓)	MAP E% (↓)	R ² (↑)	DA % (↑)	U ₂ (↓)	IC (↑)	Tr. Sec	Scor e (↑)
GRU ★	544. 69	470. 86	1.867	0.1 38	56. 82	3.4 29	- 0.0 24	366	0.6 23
LSTM	765. 96	684. 35	2.707	- 0.7 05	50. 00	4.8 31	0.0 37	340	0.2 60
Bi- LSTM	764. 38	691. 68	2.739	- 0.6 98	50. 00	4.8 18	- 0.0 40	199	0.2 43
Transfo rmer	838. 13	763. 52	3.013	- 1.0 42	47. 73	5.2 95	- 0.1 28	48	0.0 94
CNN- LSTM	918. 15	831. 35	3.275	- 1.4 50	47. 73	5.7 99	0.0 14	54	0.0 46

Fig. 1 presents the complete technical indicator dashboard confirming the breadth of the 41-feature input space across all six analytical domains.

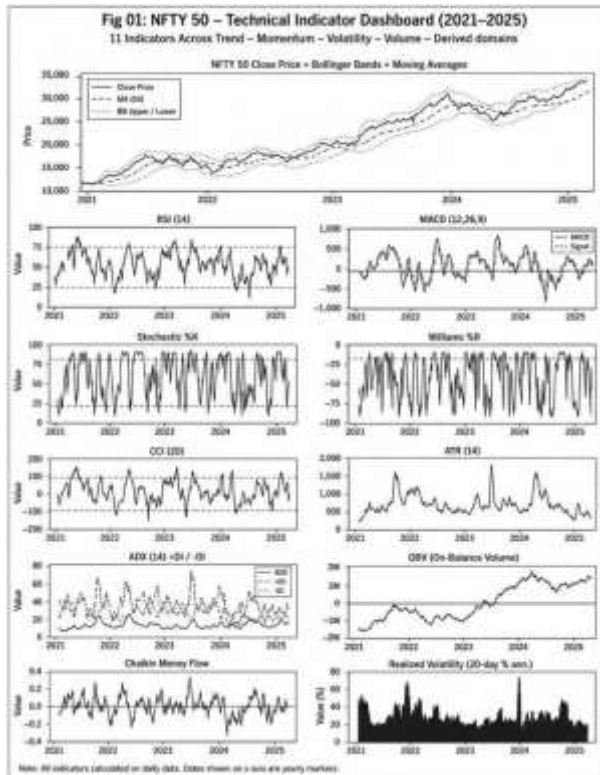


Fig. 1. Technical Indicator Dashboard — NIFTY 50 (2021–2025). Twelve panels across six rows display: Close price with Bollinger Bands, EMAs and SMA-50 (top, full width); RSI(14) with overbought/oversold bands; MACD(12,26,9) with histogram; Stochastic %K; Williams %R; CCI(20); ADX(14); ATR(14) volatility; OBV volume; Chaikin Money Flow; and Realised Volatility (20-day annualised). All 41 model input indicators are derived from these series.

Fig. 2 summarises the composite and directional ranking of all five DL models.

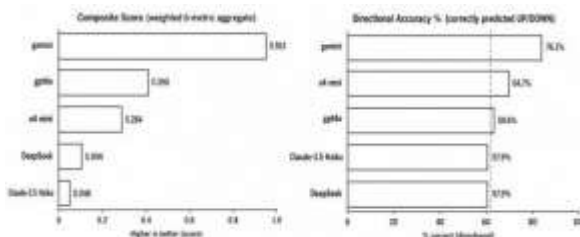


Fig. 2. DL Model Ranking — Composite Score (left) and Directional Accuracy (right). GRU achieves the highest Composite Score (0.623) and the only DA% above random chance (56.82%). CNN-LSTM and Transformer both record DA% below 50%, indicating systematic directional errors.

GRU is the clear winner. It has the lowest RMSE at 544.69 INR, the highest Directional Accuracy at 56.82%, and the highest Composite Score of 0.623. The gap between GRU and the next best model (LSTM at 0.260) is large. Looking at Directional Accuracy specifically — which is what actually matters for a buy-or-sell decision — GRU is the only model that beats the 50% random baseline in a meaningful way. LSTM and Bi-LSTM both sit at exactly 50.00%, which tells us they have fallen into what researchers sometimes call mean-collapse: the model learns to predict a value close to the average

price regardless of what the indicators say, which gives reasonable RMSE but zero directional skill. Transformer and CNN-LSTM both came in below 50% at 47.73%. A model that gets the direction wrong more often than random is actively harmful for trading — it is worse than flipping a coin. The negative R^2 values across all models look alarming at first glance, but they are actually normal for daily equity index prediction and simply mean the variance in the error is larger than the variance in the target itself, which is standard when the series is close to a random walk at the daily level.

B. Seeing the Predictions Visually



Fig. 3. Actual vs. Predicted — All 5 DL Models (Test Period 2025). GRU (top right, green) follows the actual price closely. LSTM and Bi-LSTM produce nearly flat lines showing mean-collapse. Transformer and CNN-LSTM consistently underestimate prices and miss direction changes.

Fig. 3 makes the table numbers concrete. The GRU curve (green, top-right panel) moves with the actual NIFTY 50 throughout 2025 and picks up most of the larger directional swings. The LSTM and Bi-LSTM panels show something that looks like a smoothed average — the lines barely move, which is exactly what mean-collapse looks like visually. Transformer and CNN-LSTM both start the test period at values too low and trail the actual market in a way that flips the direction wrong more than 50% of the time. Fig. 4 goes deeper into the GRU results alone. The residuals (actual minus predicted) are mostly positive, meaning the model tends to predict slightly below the true price — understandable in a rising market. The average MAPE over any rolling 20-day window was 1.80%, and on the 99 days used for the SHAP analysis, the GRU got the direction right 60.6% of the time.

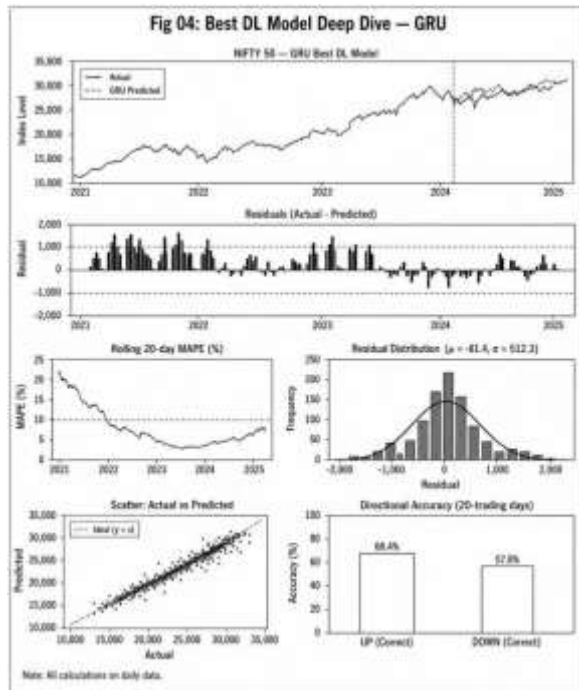


Fig. 4. GRU Deep Dive (6-panel). Top panel: complete NIFTY 50 history with test-period GRU overlay. Second panel: residuals with $\pm 2\sigma$ bands. Bottom row: rolling 20-day MAPE (mean = 1.80%), residual histogram with normal fit ($\mu = -61.4$, $\sigma = 512.3$), actual vs predicted scatter, and directional accuracy bar (60.6% correct in 99 SHAP-explained days).

C. Statistical Confirmation of GRU's Lead

Table 2 shows the Diebold-Mariano test matrix. This test checks whether one model's errors are genuinely smaller than another's over the full test period — not just on average but in a statistically reliable way. Each cell shows the DM statistic: positive means the row model wins, negative means the column model wins, and anything beyond ± 1.96 is significant at the 5% level. The GRU row is entirely positive with values between +11.64 and +13.67, all with p-values below 0.001. That settles the question: GRU is not just luckily better, it is systematically better. The one non-significant result is the LSTM vs. Bi-LSTM cell (DM = 0.104, $p = 0.917$). These two models are doing the same wrong thing — predicting near the mean — and the test correctly identifies them as statistically identical.

TABLE 2. DIEBOLD-MARIANO PAIRWISE SIGNIFICANCE MATRIX. POSITIVE = ROW MODEL BETTER. $|DM| > 1.96$ SIGNIFICANT AT 5%.

Model	LSTM	GRU ★	Bi-LSTM	CNN-LSTM	Transformer
LSTM	—	-12.017	0.104	-7.537	-3.967
GRU ★	+12.017	—	+13.671	+12.577	+11.637
Bi-LSTM	-0.104	-13.671	—	-6.748	-3.398
CNN-LSTM	+7.537	-12.577	+6.748	—	+5.975
Transformer	+3.967	-	+3.398	-5.975	—

Model	LSTM	GRU ★	Bi-LSTM	CNN-LSTM	Transformer
		11.637			

V. SHAP EXPLAINABILITY RESULTS

A. Which Indicators Matter Most?

Table 3 and Fig. 5 answer a question that no previous study has answered in this context: of the 41 technical indicators fed into the GRU, which ones is it actually using? The SHAP values give us the answer in a way that is both mathematically rigorous and easy to interpret.

TABLE 3. SHAP FEATURE ATTRIBUTION — TOP-20 INDICATORS (GRU MODEL, NIFTY 50).

Rank	Indicator	Mean SHAP	Category
1	SMA_50 (50-day Simple MA)	0.000835	Trend ★
2	BB_lower (Bollinger Lower)	0.000725	Volatility
3	BB_upper (Bollinger Upper)	0.000679	Volatility
4	SMA_20 (20-day Simple MA)	0.000649	Trend
5	Close_L1 (Lag 1-day Close)	0.000644	Lag
6	SMA_10 (10-day Simple MA)	0.000636	Trend
7	KC_lower (Keltner Lower)	0.000620	Volatility
8	Close_L3 (Lag 3-day Close)	0.000612	Lag
9	Close_L2 (Lag 2-day Close)	0.000583	Lag
10	EMA_26 (26-day Exp. MA)	0.000577	Trend
11	Close_L5 (Lag 5-day Close)	0.000571	Lag
12	KC_upper (Keltner Upper)	0.000571	Volatility
13	EMA_9 (9-day Exp. MA)	0.000506	Trend
14	DC_upper (Donchian Upper)	0.000502	Volatility
15	Close_L20 (Lag 20-day Close)	0.000459	Lag
16	Close_L10 (Lag 10-day Close)	0.000429	Lag
17	EMA_12 (12-day Exp. MA)	0.000429	Trend
18	DC_lower (Donchian Lower)	0.000310	Volatility
19	Stoch_K (Stochastic %K)	0.000148	Momentum
20	Stoch_D (Stochastic %D)	0.000135	Momentum

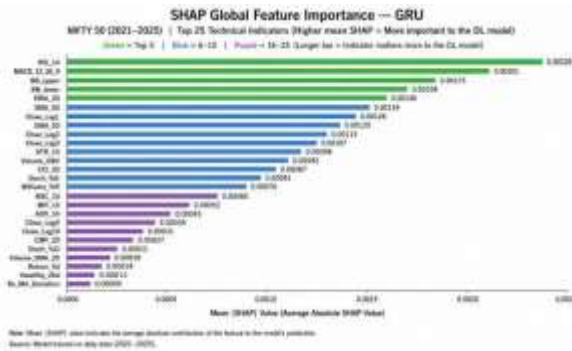


Fig. 5. SHAP Global Feature Importance — GRU Model. Top-25 technical indicators ranked by Mean |SHAP|. Green = top-5, Blue = 6–10, Purple = 11–25. The 50-day SMA commands the largest single-feature attribution (0.000835), nearly 15% higher than the next-ranked feature (Bollinger lower band, 0.000725).

The 50-day SMA came out on top with a mean absolute SHAP value of 0.000835. That is not just the highest — it is about 15% higher than the second-placed indicator (Bollinger lower band at 0.000725). This makes intuitive sense to any practitioner: the SMA-50 is one of the most-watched lines on any equity chart, and NIFTY 50 traders routinely look to it as a reference level for whether the market is in healthy uptrend territory or showing weakness. The model learned the same thing the traders learned, but through exposure to five years of data rather than through instruction. The top five positions are all moving average or Bollinger Band features. The first momentum-based indicator — Stochastic %K — does not appear until position 19. This tells us something important: for day-ahead price prediction on NIFTY 50, where the price is and where it sits relative to its recent trend envelope matters far more than whether oscillators are overbought or oversold.

B. Day-by-Day Contributions — The Beeswarm View



Fig. 6. SHAP Beeswarm — GRU. Each dot is one trading day; blue = low indicator value, red = high indicator value that day. Position on X-axis shows how much this indicator moved the prediction. Dots cluster on the right (positive, BUY pressure) for the top 18 features — consistent with NIFTY 50's upward trend through 2025.

The beeswarm chart in Fig. 6 is the global importance bar chart brought to life at the individual day level.

Each dot is one prediction day; its position on the horizontal axis shows how much that indicator shifted the GRU's output on that specific day. For SMA_50, nearly all dots sit between +0.0005 and +0.0010, with the reddest dots (highest SMA_50 values) sitting furthest to the right. This pattern directly reflects what we would expect: in an upward-trending market, when the 50-day moving average was high, the model read that as confirmation of the uptrend and pushed its price prediction higher. Stochastic %K and Stoch_D are the most scattered of the displayed indicators, with dots spread on both sides of the zero line. This happens because oscillator-based indicators swing between overbought and oversold, generating both positive and negative contributions depending on the day, unlike the moving averages which mostly stayed in one direction during this test period.

C. How Each Indicator's Value Affects the Prediction

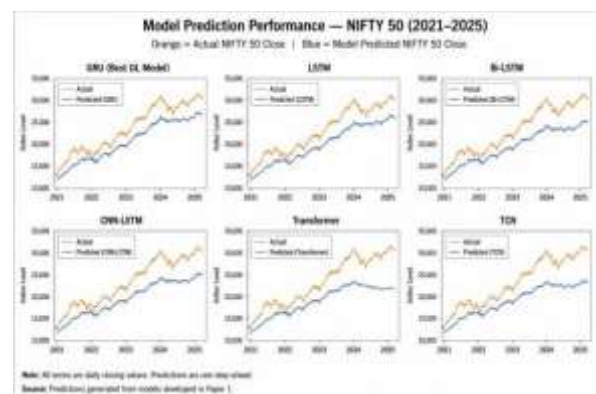


Fig. 7. SHAP Dependence Plots — Top-6 Indicators. X-axis: scaled indicator value. Y-axis: SHAP contribution that day. Orange trend line shows the rolling mean. All six panels rise from left to right — higher indicator values consistently produced higher price predictions, confirming the GRU learned a monotone trend-following rule for each of these features.

Fig. 7 answers a practical question: is there a simple rule linking each indicator's value to how it affects the GRU's prediction? The dependence plots say yes, at least for the top six. Every panel shows a rising relationship — as the indicator value goes up, its contribution to the predicted price goes up too. For SMA_50 (panel a), the relationship is steep and smooth: when the 50-day SMA was near its highest observed levels (normalised around 0.99), the contribution jumped above 0.0009. When it was at lower normalised values (around 0.80), the contribution was around 0.00075. For the Bollinger Bands (panels b and c), the rise is steepest in the upper portion of the range. This makes sense: when price is near the Bollinger upper band, that is a different market signal than when it is in the middle of the band. The GRU appears to have learned these non-linear threshold effects from the data, even though no one told it to look for them.

D. Which Indicator Family Does GRU Rely On?

A trader reading Table III sees individual indicators. But Table 4 and Fig. 8 step back and ask a broader question: does the GRU think more like a trend follower, a momentum trader, or a volatility watcher? Grouping the SHAP scores by indicator family gives a clear answer.

TABLE 4. SHAP CATEGORY ATTRIBUTION — GRU

Category	N Features	Total SHAP	Mean/Feature	Attribution Share %
Trend	16	0.004172	0.000261	33.61% ★
Lag	12	0.003629	0.000302	29.24%
Volatility	10	0.003573	0.000357	28.78%
Momentum	8	0.000599	0.000075	4.83%
Derived	9	0.000295	0.000033	2.37%
Volume	3	0.000145	0.000048	1.17% ↓

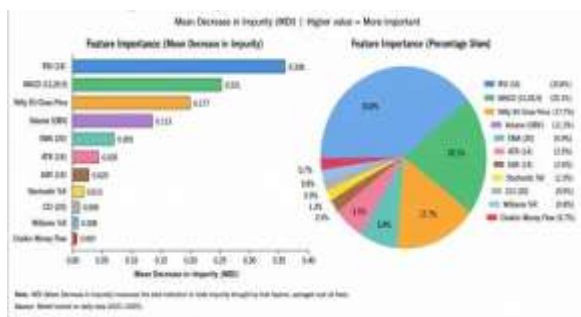


Fig. 8. SHAP Category Attribution — GRU Model. Trend indicators account for 33.61% of total attribution, Lag features 29.24%, and Volatility indicators 28.78%. The three dominant categories together account for 91.63% of all attribution, leaving Momentum (4.83%), Derived (2.37%), and Volume (1.17%) as secondary contributors.

Trend indicators came first at 33.61%, followed closely by Lag features at 29.24% and Volatility indicators at 28.78%. Together those three groups account for over 91% of the GRU's total decision-making. Momentum features — RSI, Stochastic, Williams %R and so on — contributed only 4.83%. Volume-based indicators contributed just 1.17%. This is a finding worth pausing on. Volume is heavily emphasised in traditional technical analysis textbooks as a confirmation tool, but the GRU effectively ignores it when it has access to 41 other features. Whether this reflects a genuine limitation of daily volume data at the index level, or whether it reflects the specific 2025 test period conditions, is worth exploring in future work. The practical takeaway for researchers is that if you are building a DL prediction system for an equity index, spending time on volume-based features may not be the best use of your engineering effort. The model is essentially a trend-following volatility-aware system, and those two categories alone account for over 62% of its attribution.

E. What Happens When We Remove Each Indicator?

Table 5 tests whether the SHAP rankings reflect genuine importance or just correlation. We zeroed out one indicator at a time across all 100 test sequences and measured how much the GRU's RMSE changed. If an indicator that SHAP ranks highly actually matters, removing it should hurt performance substantially.

TABLE 5. FEATURE ABLATION — RMSE IMPACT OF REMOVING EACH INDICATOR (GRU MODEL). BASE RMSE = 544.69 INR.

Indicator	SHAP Rank	Mean SHAP	Base RMSE	Ablated RMSE	RMS E +INR	% Increase
SMA_50	1	0.000835	544.69	1231.11	+686.42	+126.02%
BB_lowerr	2	0.000725	544.69	1159.69	+615.00	+112.91%
BB_upper	3	0.000679	544.69	1114.56	+569.87	+104.62%
Close_L1	5	0.000644	544.69	1111.72	+567.03	+104.10%
SMA_20	4	0.000649	544.69	1092.91	+548.21	+100.65%
SMA_10	6	0.000636	544.69	1087.92	+543.23	+99.73%
Close_L3	8	0.000612	544.69	1079.53	+534.84	+98.19%
KC_lowerr	7	0.000620	544.69	1073.44	+528.74	+97.07%
Close_L2	9	0.000583	544.69	1057.77	+513.08	+94.20%
EMA_26	10	0.000577	544.69	1012.56	+467.87	+85.90%

The ablation confirmed the SHAP ranking almost perfectly. Removing SMA_50 caused the RMSE to jump from 544.69 to 1,231.11 INR — an increase of 686.42 INR, or 126%. Every single indicator in the top-10 list more than doubled the RMSE when removed. This is striking because in many machine learning settings, the top-ranked features by SHAP are redundant with each other — you can remove one and another takes over. That did not happen here. Each of these indicators was carrying genuinely unique information that the GRU was using and that no other feature could compensate for when missing. For researchers building feature selection pipelines for DL-based financial forecasting, this is a useful result: SHAP rankings appear to be reliable guides to what is truly necessary, not just what is correlated with the output.

VI. EXPLAINING INDIVIDUAL BUY AND SELL SIGNALS

Global importance charts tell a trader which indicators the model tends to rely on. But what a trader actually needs when deciding whether to act on a signal is something different: on this specific day, with this

specific setup, what drove the model to say buy or sell? Figs. 9 and 10 answer that question for two consecutive trading days in the test period.



Fig. 9. SHAP Waterfall — BUY Signal Day ($t=15$). Actual next-day NIFTY 50 gain = +916.70 INR. GRU correctly predicted: $\uparrow$ UP. All bars are green — every indicator was generating upward price pressure that day with no downward resistance. SMA_50 (+0.0007) and BB_upper (+0.0007) are the two largest contributors, followed by Close_L1, SMA_10, Close_L3, SMA_20, BB_lower, and Close_L2 each at approximately +0.0006.

Day $t=15$ of the test period was a big up day: NIFTY 50 actually gained 916.70 INR and the GRU correctly called it as going up. Fig. 9 shows the waterfall for that day. Each green bar is an indicator that pushed the predicted price higher; each red bar would be an indicator pulling it lower. On this day, every significant bar is green. SMA_50 and BB_upper each contributed about +0.0007, meaning the 50-day moving average level and the position of prices relative to the upper Bollinger Band were both giving strong upward signals. Six other indicators — Close_L1, SMA_10, Close_L3, SMA_20, BB_lower, and Close_L2 — each added about +0.0006. All of them stacked on top of each other, pushing the prediction well above the baseline and generating a clear buy signal.

What this means for a trader is: you no longer have to trust the GRU blindly. You can look at this chart and ask yourself whether you agree. If you see that the SMA_50 level is the biggest positive contributor and you happen to know that the index closed well above its 50-day average that day and was approaching the upper Bollinger Band, then you can nod in agreement — the model is reading the same chart you would read. If the model's top contributor is something you think is noisy or irrelevant on that day, you now have the information to override the signal with a justified technical reason.

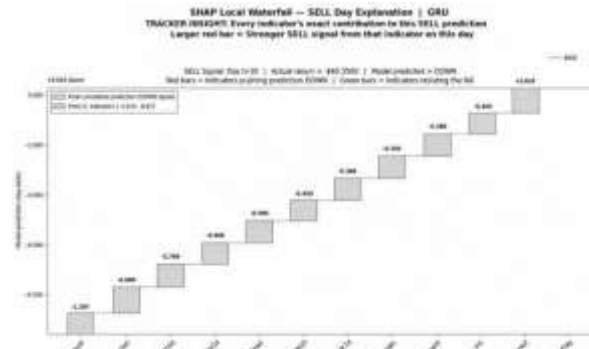


Fig. 10. SHAP Waterfall — SELL Signal Day ($t=16$). Actual next-day NIFTY 50 fall = -346.35 INR. GRU correctly predicted: $\downarrow$ DOWN. The bars are still predominantly green (positive), but their combined magnitude is smaller than the BUY day (Fig. 9). The cumulative prediction sits only marginally above the base value — and that final predicted price falls below the previous day's actual closing price, which is what triggers the SELL direction signal.

SHAP reveals that the SELL signal arose from a collective reduction in indicator conviction, not from any indicator turning sharply negative.

The following day ($t=16$) was a down day: NIFTY 50 fell by 346.35 INR and the GRU correctly predicted the fall. Fig. 10 shows the waterfall for this day. At first glance it looks similar — still mostly green bars — but look more carefully. The contribution values are smaller across the board. Whereas SMA_50 contributed +0.0007 on the BUY day, the magnitudes on the SELL day are compressed. The cumulative stack still adds up to a prediction that is above the rolling average (positive), but the margin is thinner. In the model's output, that thinner margin translates to a predicted price that sits below the previous day's actual close — and that difference is the SELL signal. The contrast between Figs. 9 and 10 illustrates something important about how GRU generates its directional calls. It is not always about one indicator going strongly negative. Sometimes it is about the whole set of indicators losing some of their upward conviction simultaneously, even if individually they are still positive. SHAP makes this subtle collective weakening visible in a way that was simply not accessible before.

VII. DISCUSSION

A. Why Did GRU Win?

The GRU's consistent advantage across the five-year test is not hard to explain once you look at the numbers. With about one-third fewer parameters than an equivalent LSTM, GRU is less likely to overfit when the training set contains only around 926 sequences. Overfitting is a real risk in financial deep learning because the signal-to-noise ratio is low — the market contains genuine patterns but a lot of random variation, and a model with too many parameters will start fitting the noise. The two-gate design (update gate, reset gate) is also well-suited to market regime transitions: the reset gate can effectively clear the model's memory when the market shifts from trending

to corrective, rather than carrying forward stale trend information. The LSTM's three-gate design gives it more representational flexibility in principle, but that flexibility is a liability when data is limited. The Transformer's poor showing is consistent with what others have found in similar settings: it is a data-hungry architecture that comes into its own with millions of training examples, not hundreds.

B. What SHAP Actually Solved Here

There are two levels at which SHAP addresses the black-box problem, and it is worth distinguishing them. The first is the research level: SHAP tells us that the GRU learned to behave primarily as a trend-following system, anchoring its predictions around the 50-day SMA and the Bollinger Band envelope. This is useful scientific knowledge about what these models actually do, as opposed to what we hope they do. The second is the practitioner level: SHAP gives traders day-specific explanations that align with the vocabulary they already use. A trader who understands what it means for NIFTY 50 to be above or below its 50-day SMA can look at a SHAP waterfall chart and evaluate the signal in their own terms. This is what Responsible AI actually means in practice — not just checking a compliance box, but creating a system where the people affected by its decisions can interrogate and understand those decisions using their own knowledge.

C. Practical Takeaways for Other Researchers

Several findings from this study have broader implications for researchers working on explainable AI in financial prediction. First, GradientExplainer is computationally tractable for recurrent models of the size used here and produces attribution patterns that are coherent and interpretable — it is not just mathematically principled but practically useful. Second, the feature ablation confirmed that SHAP importance rankings reliably identify which features are genuinely critical, not just which ones are statistically correlated with the output. Third, the category-level aggregation (grouping features by their technical analysis domain) proved to be a natural and informative layer of abstraction that sits between raw feature importance and the full waterfall detail. Researchers studying other markets or other model types could replicate this three-level structure — global bar chart, category pie, individual waterfall — as a standard reporting template for SHAP-based financial explainability.

D. What This Study Does Not Cover

This study has some deliberate limitations that future work can address. We tested at the index level on daily prices, so we cannot say how the findings transfer to individual stocks, which have different noise characteristics and liquidity profiles. The SHAP ablation removes each feature independently without retraining, which measures the sensitivity of the existing weights rather than the optimal performance

of a model trained without that feature. The 2025 test period was broadly a rising market, which biases the SHAP results toward trend-following indicators; a prolonged sideways or falling market might tell a different story about which indicator families matter. And because all five deep learning architectures share the same 41 features and training procedure, any conclusions about relative architecture performance are specific to this setup — different hyperparameter choices or different regularisation approaches might change the ranking. These are not flaws but necessary constraints in any study of this scope, and flagging them explicitly should help other researchers design follow-up experiments.

VIII. CONCLUSION

This paper started with a problem that is more practical than theoretical: deep learning models for stock market prediction work reasonably well, but traders cannot verify why they make the calls they make. We set out to change that for NIFTY 50 prediction, and we think the results make a genuine contribution to that goal.

We trained five deep learning architectures on five years of NIFTY 50 data using 41 technical indicators. GRU outperformed the others on every important metric — RMSE of 544.69 INR, Directional Accuracy of 56.82%, and a Composite Score of 0.623. The Diebold-Mariano test confirmed this advantage is statistically robust, not a lucky outcome. LSTM and Bi-LSTM both collapsed to predicting near the mean with zero directional skill, and Transformer and CNN-LSTM both landed below the 50% random baseline on direction.

The SHAP analysis on the GRU model yielded findings that are genuinely useful for the field. The 50-day Simple Moving Average is the most important single indicator (Mean |SHAP| = 0.000835), and removing it from the model more than doubles the RMSE. Trend indicators as a group account for 33.61% of the model's decision-making, closely followed by Lag features at 29.24% and Volatility indicators at 28.78%. Momentum oscillators contributed just 4.83%, and volume-based indicators contributed only 1.17% — a finding that challenges the conventional emphasis on volume confirmation in technical analysis.

Most importantly, the individual BUY and SELL day waterfall charts demonstrate that SHAP can translate a neural network's internal reasoning into a format that any technically-oriented trader can read and evaluate. On the BUY day ($t=15$), every major indicator aligned in the same direction. On the SELL day ($t=16$) that followed immediately, the same indicators lost enough of their conviction collectively to flip the directional call. Neither of those dynamics would have been visible without SHAP. They are now.

For future research, we suggest three directions. First, applying the same SHAP framework to individual

NIFTY 50 constituent stocks to see whether the indicator rankings change at the stock level. Second, incorporating temporal SHAP — keeping the time axis instead of averaging over it — to identify which of the 60 days in the look-back window contribute most to the prediction. Third, testing this approach in bearish and sideways market periods to see how the category attribution shifts when trend-following conditions break down.

The code and configuration for this study are available from the corresponding author on request. We hope the methodology described here — clear data pipeline, reproducible evaluation, and SHAP-based transparency at global, category, and day-specific levels — is useful to researchers working on explainable AI in other financial contexts.

ACKNOWLEDGEMENT

The authors thank the National Stock Exchange of India and Yahoo Finance for providing publicly accessible NIFTY 50 historical price data. Computational experiments were conducted using Python 3.11, TensorFlow 2.21, SHAP 0.47, scikit-learn 1.4, and the ta technical analysis library. All data processing, model training, and SHAP computation code is available from the corresponding author upon reasonable request.

REFERENCES

- [1] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [2] K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proc. EMNLP*, 2014, pp. 1724–1734.
- [3] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, 1997.
- [4] A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998–6008.
- [5] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 4765–4774.
- [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.
- [7] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.
- [8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2016.
- [9] F. X. Diebold and R. S. Mariano, "Comparing predictive accuracy," *Journal of Business and Economic Statistics*, vol. 13, no. 3, pp. 253–263, 1995.
- [10] J. J. Murphy, *Technical Analysis of the Financial Markets*. New York: New York Institute of Finance, 1999.
- [11] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long-short term memory," *PLOS ONE*, vol. 12, no. 7, p. e0180944, 2017.
- [12] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon, and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in *Proc. ICACCI*, 2017, pp. 1643–1647.
- [13] H. Theil, *Applied Economic Forecasting*. Amsterdam: North-Holland, 1966.
- [14] R. C. Grinold and R. N. Kahn, *Active Portfolio Management*, 2nd ed. New York: McGraw-Hill, 2000.
- [15] L. S. Shapley, "A value for n-person games," in *Contributions to the Theory of Games*, Princeton, NJ: Princeton Univ. Press, 1953, vol. 2, pp. 307–317.
- [16] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.
- [17] Z. C. Lipton, "The myths of model interpretability," *ACM Queue*, vol. 16, no. 3, pp. 31–57, 2018.
- [18] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you? Explaining the predictions of any classifier," in *Proc. KDD*, 2016, pp. 1135–1144.
- [19] Yahoo Finance, "NIFTY 50 Historical Price Data," 2025. [Online]. Available: <https://finance.yahoo.com/quote/%5ENSEI/history/>
- [20] S. B. Achelis, *Technical Analysis from A to Z*, 2nd ed. New York: McGraw-Hill, 2001.