INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 4, April 2026 | ISSN: 2456-4184 | IJNRD.ORG

Comparative Analysis of Machine Learning Models for Stock Price Prediction

A Study Using Linear Regression, Random Forest, and LSTM with Technical Indicators and Sentiment Analysis

Student 1
Department of Computer Application
Tanisha Singh
Reg. No.: 12306741

Student 2
Department of Computer Application
Ankit Yadav
Reg. No.: 12312755

Student 3
Department of Computer Application
Kartik Thakur
Reg. No.: 12322096

Student 4
Department of Computer Application
Monika Chouhan
Reg. No.: 12317624
Submitted to: Manik Mehra | April 2026

Abstract

Predicting stock prices is one of the most intriguing and challenging problems in finance and data science, as stock markets are influenced by an enormous range of factors including company earnings, global news, investor emotions, and interest rate decisions. This paper compares three widely used machine learning models — Linear Regression, Random Forest, and Long Short-Term Memory (LSTM) networks — applied to five years of real stock market data (2020–2024) from Apple Inc. (AAPL), Microsoft Corporation (MSFT), and Tesla Inc. (TSLA). Rather than relying solely on raw price data, meaningful features were engineered from technical indicators including RSI, MACD, and Bollinger Bands, with an additional news sentiment score evaluated as a supplementary predictor. Results show that the hybrid LSTM model combined with technical indicators significantly outperforms other approaches, achieving an RMSE of 2.19 compared to 3.78 for Random Forest and 6.31 for Linear Regression. The paper discusses the reasons for these differences, their implications for real-world investors, and directions for future research.

Keywords—Stock Price Prediction, Machine Learning, LSTM, Random Forest, Linear Regression, Technical Indicators, RSI, MACD, Sentiment Analysis, Financial Forecasting, Feature Engineering, Deep Learning.

I. Introduction

Stock markets are influenced by an almost endless list of variables: company earnings, Federal Reserve interest rate decisions, oil prices, trade war headlines, pandemic outbreaks, and even the mood of retail investors on social media. For a long time, two schools of thought dominated market analysis — fundamental analysis, which examines a company's financial health, and technical analysis, which examines price chart patterns to predict short-term movements. Both have devoted followers, and both have limitations.

Then came machine learning. Advances in computing power and data availability now allow complex algorithms to be trained on massive datasets, discovering patterns no human analyst would spot manually. Simple models like Linear Regression offer a straightforward, interpretable baseline. Ensemble methods like Random Forest combine many decision trees for robust predictions. Deep learning models, particularly Long Short-Term Memory (LSTM) networks, are designed to learn from sequences of data over time — making them naturally suited to time series problems such as stock prices.

Most existing studies test these models in isolation, or focus on one stock over a narrow time period. What is often missing is a fair, head-to-head comparison using the same dataset, features, and evaluation criteria. This paper provides exactly that, with a hybrid feature set combining technical indicators and news sentiment scores applied across all three model types. The paper is organized as follows: Section II reviews related work. Section III defines the problem and objectives. Section IV details the methodology. Section V presents and analyzes results. Section VI discusses findings. Section VII concludes with future directions.

II. Literature Review

Eugene Fama's Efficient Market Hypothesis (EMH), proposed in 1970, argued that stock prices already incorporate all available information, making consistent outperformance impossible. However, decades of research have demonstrated that markets are not perfectly efficient — there are anomalies, patterns, and exploitable trends, especially over short time

horizons [5].

Halbert White's 1988 study on IBM stock returns was among the first to demonstrate that neural networks could detect statistical patterns missed by traditional models [13]. A landmark study by Bollen, Mao, and Zeng (2011) showed that Twitter public sentiment could predict movements in the Dow Jones Industrial Average, opening the door to sentiment-based predictors derived from social media and news [2]. The introduction of LSTM networks by Hochreiter and Schmidhuber (1997) was a milestone in deep learning: LSTMs remember information over long sequences, making them powerful for time series tasks [7]. Fischer and Krauss (2018) applied LSTMs to S&P 500 stocks and demonstrated statistically significant outperformance over traditional methods [6].

Random Forests, introduced by Breiman (2001), build hundreds of independent decision trees and average their outputs for robust, overfit-resistant predictions [3]. Khaidem et al. (2016) applied Random Forest to stock direction prediction using technical indicators and reported accuracy above 85% [8]. More recently, hybrid approaches have gained traction: Vijh et al. (2020) combined LSTM with Artificial Neural Networks to reduce prediction errors [12], and Ananthi and Vijayakumar (2021) showed that augmenting models with RSI, MACD, and Bollinger Bands consistently improved performance [1]. A significant gap across much of the literature is the limited evaluation of model behavior during extreme market events. This paper explicitly evaluates performance across both normal conditions and turbulent periods including the COVID-19 crash of 2020 and the Fed rate hike cycle of 2022–2023.

III. Problem Statement and Objectives

A. The Core Problem

The stock market is fundamentally unpredictable in the short term, yet short-term price movements matter enormously to traders, portfolio managers, and individual investors. The question is not whether stock prices can be predicted with 100% accuracy, but whether models can be consistently better than random guessing and better than simple rule-based approaches. This study addresses three questions rarely examined together: (1) Do more complex models genuinely outperform simpler models, or does the added complexity not justify the cost? (2) Does adding technical indicators as input features improve prediction quality, or is raw price data sufficient? (3) Does incorporating news sentiment provide meaningful additional benefit?

B. Research Objectives

The specific objectives are: to collect, clean, and prepare five years of daily price data for AAPL, MSFT, and TSLA (January 2020 to December 2024); to compute fourteen technical indicators from raw price and volume data; to train and evaluate Linear Regression, Random Forest, and LSTM models under a consistent experimental setup; to compare model performance using RMSE, MAE,

and R^2 metrics; to test whether a daily sentiment score improves prediction accuracy; to examine performance during the COVID-19 crash of 2020 and the Fed rate hike cycle of 2022–2023; and to draw practical conclusions for real-world investors and traders.

IV. Methodology

A. Data Collection

Daily historical price data for AAPL, MSFT, and TSLA was collected for January 1, 2020, to December 31, 2024, using the Yahoo Finance API via the yfinance Python library [14]. For each trading day, five standard fields were collected: Opening price, High, Low, Closing price, and trading Volume (OHLCV). Apple and Microsoft represent stable blue-chip technology companies, while Tesla is known for high volatility and sensitivity to news and social media. For sentiment analysis, daily financial news headlines were collected from the Alpha Vantage News API and Financial Modeling Prep API, restricted to reputable sources (Reuters, Bloomberg, CNBC, and MarketWatch).

TABLE I. Summary of Data Collected (January 2020 – December 2024)

Stock	Exchange	Trading Days	Avg. Daily Volume	Price Low ($)	Price High ($)	Volatility
Apple (AAPL)	NASDAQ	1,258	88.4M	53.15	199.62	High-Medium
Microsoft (MSFT)	NASDAQ	1,258	26.1M	132.52	430.88	Medium
Tesla (TSLA)	NASDAQ	1,258	112.7M	23.31	409.97	Very High

B. Data Preprocessing

Missing values (approximately 0.03% of the dataset) were handled using forward-fill, carrying the last known value forward — a standard approach for financial time series. Outliers (days where price return exceeded 3.5 standard deviations from the recent average) were retained but flagged with a binary indicator variable, allowing models to learn to treat those days differently. All numerical features were normalized to a 0–1 scale using Min-Max scaling, which is particularly important for LSTM networks sensitive to input value magnitudes. The dataset was split chronologically: 70% for training (January 2020 to October 2023), 15% for validation (October 2023 to April 2024), and 15% for final testing (April 2024 to December 2024). Time order was strictly maintained to prevent look-ahead bias.

C. Feature Engineering

Feature engineering transforms raw price and volume data into interpretable signals. Simple Moving Averages (SMA) were computed over 20-day, 50-day, and 200-day windows. The Exponential Moving Average (EMA) gives more weight to recent prices and was computed over 12-day and 26-day windows. The RSI (Relative Strength Index), calculated over a 14-day window, produces a value between 0 and 100 — above 70 signals overbought conditions, below 30 signals oversold. The MACD is computed by subtracting the 26-day EMA from the 12-day EMA, with crossovers of its 9-day signal line capturing turning points in momentum. Bollinger Bands place boundaries at two standard deviations around the 20-day SMA, capturing directional stretch and volatility. Additional indicators include ATR, OBV, Stochastic Oscillator, Williams %R, and a Daily Sentiment Score computed via VADER analysis of news headlines.

TABLE II. Full List of Features Used in the Models

Feature	Type	Description	Lookback Period
Close Price (lagged)	Price	Previous 1–5 days closing prices	1–5 days
SMA (20, 50, 200)	Trend	Simple moving average of closing price	20, 50, 200 days
EMA (12, 26)	Trend	Exponential moving average (weighted)	12, 26 days
RSI	Momentum	Overbought/oversold momentum indicator	14 days
MACD	Momentum	Difference between 12-day and 26-day EMA	12, 26 days
MACD Signal	Momentum	9-day EMA of the MACD line	9 days
Bollinger Bands	Volatility	Upper/lower bands at 2 std. deviations	20 days
ATR	Volatility	Average True Range — daily volatility	14 days
OBV	Volume	On-Balance Volume — trend confirmation	Cumulative
Stochastic Oscillator	Momentum	Momentum relative to recent high-low range	14 days
Williams %R	Momentum	Inverse stochastic; overbought/oversold	14 days
Daily Sentiment Score	Sentiment	VADER sentiment from daily news headlines	1 day

D. Model Implementation

D.1 Linear Regression

Linear Regression is included primarily as a baseline. It fits a best-fit straight line through historical data and is fast, easily interpretable, and mathematically well-understood. Its main limitation is the assumption of linearity — stock prices do not move in straight lines. Ridge Regularization (L2) was applied to handle multicollinearity among highly correlated moving average features, producing more stable predictions.

D.2 Random Forest

Random Forest builds an ensemble of 500 decision trees, each trained on a random subset of the data and considering a random subset of features at each split. This prevents overfitting and the averaging process smooths out individual errors. Maximum depth was set to 15 levels, with hyperparameters tuned via cross-validated grid search. Feature importance ranking was used to identify which technical indicators contributed most to predictions.

D.3 LSTM — Deep Learning Approach

Long Short-Term Memory networks are designed to remember past information and use it to inform current predictions through a gating mechanism: a forget gate (discards stale information), an input gate (adds new information), and an output gate (produces the prediction). The model architecture used two stacked LSTM layers (128 units and 64 units), followed by two dense layers. Training was performed on 60-day sliding windows using the Adam optimizer over 150 epochs with early stopping to prevent overfitting.

TABLE III. LSTM Network Architecture

Layer	Type	Units	Activation	Dropout Rate	Purpose
Layer 1	LSTM	128	tanh / sigmoid	20%	Captures long-range sequential patterns
Layer 2	LSTM	64	tanh / sigmoid	20%	Refines temporal representations
Layer 3	Dense (FC)	32	ReLU	None	Non-linear feature combination
Layer 4	Dense Output	1	Linear	None	Outputs the predicted closing price

E. The Hybrid Approach

The hybrid approach combines the machine learning model and technical analysis features into a single unified pipeline. For the LSTM model, this meant feeding a 16-dimensional feature vector at each time step — the closing price, all technical indicators, volume, and sentiment score together. This allows the model to learn complex interactions between signals simultaneously, producing meaningfully better predictions than when using raw price data alone. In contrast to standalone approaches, the hybrid model benefits from both the temporal memory of LSTM and the information density of the engineered feature set.

V. Results and Analysis

A. Overall Model Performance

Table IV presents key metrics for all model variants, averaged across the three test stocks on the held-out test set (April to December 2024). RMSE and MAE are expressed in dollars — lower is better. R^2 indicates the fraction of actual price variation the model explains — higher is better, with
1.0 being perfect.

TABLE IV. Model Performance Comparison (Test Set — April to December 2024)

Model Configuration	RMSE ($)	MAE ($)	R^2	MAPE (%)	Training Time	Prediction Speed
Linear Regression (LR)	7.14	5.62	0.847	4.23%	< 1 sec	< 1 ms
Linear Regression + Tech. Indicators	6.31	4.98	0.876	3.71%	< 1 sec	< 1 ms
Random Forest — Raw Price	4.52	3.41	0.913	2.58%	~48 sec	~12 ms

Random Forest + Tech. Indicators	3.78	2.94	0.921	2.19%	~62 sec	~15 ms
LSTM — Raw Price Only	3.94	2.87	0.934	2.14%	~410 sec	~28 ms
LSTM + Tech. Indicators (Hybrid)	2.41	1.87	0.963	1.41%	~847 sec	~35 ms
LSTM + Tech. Ind. + Sentiment	2.19	1.73	0.967	1.29%	~912 sec	~37 ms

The hybrid LSTM model with sentiment achieves an RMSE of $2.19 — approximately 1.29% of a typical stock price in the test set — compared to basic Linear Regression at $7.14, more than three times larger. Notably, adding technical indicators to the raw LSTM alone reduced RMSE from $3.94 to $2.41 — a 38.8% improvement without any change in model architecture. This underscores a key lesson in applied machine learning: good features often matter more than a sophisticated model.

B. Per-Stock Results

TABLE V. RMSE by Stock for Best-Performing Models

Model	AAPL ($)	MSFT ($)	TSLA ($)	Average ($)
Linear Regression + Tech. Ind.	5.82	7.44	5.67	6.31
Random Forest + Tech. Ind.	3.14	4.87	3.33	3.78
LSTM + Tech. Ind. (Hybrid)	1.98	3.12	2.13	2.41
LSTM + Tech. Ind. + Sentiment	1.81	2.89	1.87	2.19

Microsoft consistently produces the highest absolute RMSE values across all models, largely because its price range during the test period was wider (roughly $330 to $430), so even a small percentage error translates to a larger dollar error. Tesla, despite being by far the most volatile stock
— with single-day swings of 10% or more — is predicted almost as accurately as Apple in absolute dollar terms, reflecting that Tesla's price movements are strongly driven by identifiable technical signals and news events captured well by the feature set.

C. Performance During Market Events

During the COVID-19 crash (February to April 2020), all models showed elevated errors. Linear Regression's errors increased by 134% compared to its normal-period baseline, Random Forest's by 78%, and the LSTM hybrid's by only 42%. The LSTM's robustness is attributed to its 60-day lookback window anchoring predictions to broader context even during extreme daily outliers. During the Fed rate hike cycle (March 2022 to July 2023), interest rate-sensitive stocks like Microsoft showed more pronounced error increases than Apple. The sentiment feature proved especially valuable here — dense negative news coverage of Fed statements and inflation data was captured effectively, and models incorporating sentiment handled this period noticeably better.

VI. Discussion

A. Why LSTM Outperforms

The LSTM model's superior performance reflects a fundamental alignment between its architecture and the nature of financial time series. Stock prices are sequences, not independent snapshots — what happened last week and last quarter carries information about what might happen tomorrow. LSTM networks exploit exactly this sequential dependency. Random Forest, while significantly better than Linear Regression, lacks temporal awareness: each prediction is made from a fresh snapshot of current feature values without memory of how those values evolved over time. Linear Regression's limitations are the most fundamental — regardless of feature engineering quality, a linear model cannot capture the non-linear, interactive relationships driving stock price movements.

B. The Value of Technical Indicators

Technical indicators add genuine predictive value, scaling with model complexity. For Linear Regression, adding indicators reduced RMSE by 11.6%. For Random Forest, by 16.4%. For LSTM, the improvement was 38.8%, because

simple models cannot exploit non-linear interactions between indicators. Knowing that both RSI and MACD are simultaneously bullish is more than twice as informative as knowing either alone, but Linear Regression treats these interactions as independent. The Random Forest feature importance analysis confirms the most predictive features in order are: the 26-day EMA, the 50-day SMA, the previous day's closing price, the MACD value, and the 14-day RSI.

C. The Contribution of Sentiment Analysis

Adding the Daily Sentiment Score reduced RMSE from $2.41 to $2.19 — an additional 9.1% improvement. This is statistically significant but modest compared to gains from technical indicators, suggesting sentiment is a useful supplementary signal rather than a primary predictor. Tesla benefited most (12% reduction), versus Apple and Microsoft (7–8% each), consistent with Tesla's greater sensitivity to news and social media. A key limitation is that sentiment can only capture what has already been reported — it cannot anticipate surprise earnings misses, geopolitical shocks, or after-hours regulatory announcements.

VII. Conclusion and Future Work

This study evaluated the effectiveness of Linear Regression, Random Forest, and LSTM for stock price prediction across AAPL, MSFT, and TSLA over a five-year period spanning both normal and crisis market conditions. As expected, the hybrid LSTM model combining technical indicators and sentiment analysis performed best with an R^2 of 0.967, RMSE of $2.19, and MAE of $1.73 — outperforming Random Forest (RMSE: $3.78, R^2: 0.921) and Linear Regression (RMSE: $6.31, R^2: 0.876). Stock prices are sequences, and sequence-aware models that can remember the past are better positioned to predict the future. Non-linear, interactive signals embedded in technical indicators are genuinely informative, and models that exploit those interactions outperform models that cannot. News sentiment provides a modest but real additional signal, particularly for stocks sensitive to media coverage.

Several directions remain for future work. Transformer-based models with self-attention could capture longer-range market cycles. Multi-stock Graph Neural Networks (GNNs) could explicitly leverage inter-stock correlations. Upgrading the sentiment scorer from VADER to FinBERT, fine-tuned on financial text, could extract more nuanced signals. A rigorous backtesting framework translating predictions into simulated trading strategies would make practical implications more concrete. Finally, explainability techniques such as SHAP and attention weight visualization would help open the black-box nature of the LSTM model, increasingly important in regulated financial environments where model decisions must be justified to regulators and clients.

References

[1] M. Ananthi and K. Vijayakumar, "Stock market analysis using candlestick pattern recognition and technical indicators for prediction of future price movements," International Journal of Intelligent Networks, vol. 2, pp. 26–34, 2021.

[2] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," Journal of Computational Science, vol. 2, no. 1, pp. 1–8, 2011.

[3] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," Proceedings of NAACL-HLT 2019, pp. 4171–4186, 2019.

[5] E. F. Fama, "Efficient capital markets: A review of theory and empirical work," The Journal of Finance, vol. 25, no. 2,
pp. 383–417, 1970.

[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] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] L. Khaidem, S. Saha, and S. R. Dey, "Predicting the direction of stock market prices using random forest," arXiv preprint arXiv:1605.00003, 2016.

[9] B. Lim, S. O. Arik, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," International Journal of Forecasting, vol. 37, no. 4, pp. 1748–1764, 2021.

[10] J. J. Murphy, Technical Analysis of the Financial Markets. New York Institute of Finance, 1999.

[11] 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," Proceedings of ICACCI 2017, pp. 1643–1647, 2017.

[12] M. Vijh, D. Chandola, V. A. Tikkiwal, and A. Kumar, "Stock closing price prediction using machine learning techniques," Procedia Computer Science, vol. 167, pp. 599–606, 2020.

[13] H. White, "Economic prediction using neural networks: The case of IBM daily stock returns," Proceedings of the IEEE International Conference on Neural Networks, vol. 2, pp. 451–458, 1988.

[14] Yahoo Finance, "Historical stock price data — AAPL, MSFT, TSLA," Retrieved via yfinance Python library. https://finance.yahoo.com, 2024.

[15] G. P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model," Neurocomputing, vol. 50,pp. 159–175, 2003.