2025

FORECASTING NEXT-DAY S&P500 CLOSING PRICE USING DEEP LEARNING MODELS

FINAL PROJECT

SEBASTIAN HEREDIA

BSAN 6070 | INTRODUCTION TO MACHINE LEARNING

FORECASTING NEXT-DAY S&P 500 CLOSING PRICE USING DEEP LEARNING MODELS

1. Technical Objectives

The primary objective of this project is to forecast the next day closing price of the S&P 500 Index using three deep learning architectures: Feedforward Neural Network (FNN), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM).

The S&P 500 Index represents the performance of the 500 largest publicly traded companies in the United States and is considered one of the most accurate gauges of the overall U.S. equity market. It is highly liquid, broadly diversified, and closely watched by investors, analysts, policymakers, and academics. For our purposes, it offers several compelling advantages:

- Economic significance: As a leading market benchmark, the S&P 500 captures investor sentiment and macroeconomic trends, making it an ideal target for predictive modeling.
- Data richness: The S&P 500 has a long and uninterrupted historical time series, with readily available daily data (Open, High, Low, Close, Volume), enabling robust model training and validation.
- Practical relevance: Accurate short-term forecasts of the S&P 500's closing price can support portfolio allocation decisions, hedging strategies, and algorithmic trading, making our analysis highly applicable to real-world financial scenarios.

Given its central role in global markets, any insights or forecast improvements derived from modeling the S&P 500 can be generalized or extended to other market indices and investment instruments.

Cross-Model Goals

Measure and compare out-of-sample predictive accuracy using RMSE, MAE, and MAPE.

Conduct hyperparameter tuning to assess how architecture-specific parameters affect performance. We will implement a systematic tuning process for each model using either Bayesian Optimization or Random Search, possibly supported by Optuna and KerasTuner, respectively.

Hyperparameters to be optimized:

- Number of hidden layers and units
- Learning rate
- Batch size
- Dropout rate
- Sequence length (for CNN and LSTM)
- Number and size of convolution filters (for CNN)

Evaluation Metrics:

- Root Mean Square Error (RMSE)
- Mean Absolute Error (MAE)
- Mean Absolute Percentage Error (MAPE)

We will evaluate models on:

- Validation set to tune parameters.
- Test set to assess out-of-sample performance.

2. Data and Data Preparation

The dataset used in this study is composed of daily observations from January 1, 2022 to January 31, 2023 and was curated from the Federal Reserve Economic Data (FRED) platform. The target variable is the S&P 500 Index (SP500), representing the value of the U.S. equity market. Alongside this target, the dataset includes a collection of macroeconomic and market indicators intended to capture a holistic view of factors influencing the stock market.

The input features include a list of 28 explanatory variables used to predict the SP500:

Variable	Description
DGS1MO	1-Month Treasury Constant Maturity Rate
DGS6MO	6-Month Treasury Constant Maturity Rate
DGS1	1-Year Treasury Constant Maturity Rate
DGS5	5-Year Treasury Constant Maturity Rate
DGS10	10-Year Treasury Constant Maturity Rate
DGS30	30-Year Treasury Constant Maturity Rate
DTB4WK	4-Week Treasury Bill Secondary Market Rate
DTB3	3-Month Treasury Bill Secondary Market Rate
DTB6	6-Month Treasury Bill Secondary Market Rate
EFFR	Effective Federal Funds Rate
IORR	Interest Rate on Reserve Balances
OBFR	Overnight Bank Funding Rate
TEDRATE	TED Spread
RRPONTSYD	Overnight Reverse Repurchase Agreements
DCOILWTICO	Crude Oil Prices (West Texas Intermediate)
DHHNGSP	Henry Hub Natural Gas Spot Price
DJFUELUSGULF	U.S. Gulf Coast Kerosene-Type Jet Fuel Price
USD3MTD156N	3-Month LIBOR Based on U.S. Dollar
DTWEXAFEGS	Trade Weighted U.S. Dollar Index
DEXUSAL	U.S. Dollar to Albanian Lek Exchange Rate
DEXUSEU	U.S. Dollar to Euro Exchange Rate
DEXJPUS	Japanese Yen to U.S. Dollar Exchange Rate
DEXCHUS	Chinese Yuan to U.S. Dollar Exchange Rate
DEXCAUS	Canadian Dollar to U.S. Dollar Exchange Rate
DEXMXUS	Mexican Peso to U.S. Dollar Exchange Rate
DEXINUS	Indian Rupee to U.S. Dollar Exchange Rate

DEXUSUK	U.S. Dollar to British Pound Exchange Rate
DEXUSNZ	U.S. Dollar to New Zealand Dollar Exchange Rate

These features were selected for their relevance in influencing stock market behavior and were standardized or normalized prior to model input. By including both forward-looking (e.g., inflation expectations) and lagging indicators (e.g., employment reports), the dataset aims to provide the models with a robust foundation for learning patterns that influence short-term S&P 500 movements.

Major External Events

This study focuses on a specific market window spanning from January 2022 to January 2023. This period was marked by notable economic disruptions and geopolitical events that impacted market volatility and investor behavior.

One of the most significant influences during this window was the U.S. Federal Reserve's monetary tightening cycle. Starting in March 2022, the Fed executed a series of aggressive interest rate hikes in response to inflation reaching multi-decade highs. These surprise rate changes injected substantial volatility into the equity markets and directly affected the S&P 500's day-to-day fluctuations.

Simultaneously, the Russia-Ukraine war, which escalated in February 2022, introduced global uncertainty and caused commodity price surges, further destabilizing equity markets. The combined effect of geopolitical tensions and inflationary pressure created sharp and unpredictable movements in the index. Such external shocks presented an ideal testbed to evaluate the robustness and adaptability of deep learning models.

Train-Test-Validation Split

The dataset was split into training (80%), validation (10%), and test (10%) subsets. This split balances the need for sufficient data to train complex neural networks while reserving enough data to rigorously evaluate model generalization. The training set included data from January 2022 to early December 2022, the validation set spanned mid-December 2022, and the test set covered the final stretch through January 2023. This temporal structure ensures that the models are evaluated on their ability to predict future outcomes based on unseen patterns, rather than re-learning known market trends.

Preprocessing

Data preprocessing involved a consistent pipeline across all models. First, we noted that for many explanatory variables there were missing values during the start of the window (January 2022) hence any missing values in the dataset were addressed using backward-fill imputation leveraging future values present. After conducting this, there were still a few missing values which were addressed with forward-fill imputation. These methods ensured continuity of the time series without introducing artificial distortions.

Normalization techniques were tailored to the model's architecture. For the LSTM, which is sensitive to input scale due to their sequential nature, MinMaxScaler was used to transform features to the [0, 1] range. This ensured consistent numerical behavior across time steps. For the FNN and CNN model, StandardScaler was applied to center the data around a mean of 0 with unit variance, aligning with the models' fully connected layer expectations.

The sequence window used for CNN and LSTM was 10 days. This decision was based on a balance between capturing recent market dynamics and maintaining model tractability. The window size allowed each model to consider two weeks of trading before generating a prediction.

Through this preprocessing pipeline, the dataset was rendered suitable for high-performance learning, ensuring consistency, reducing bias from anomalies, and aligning with the technical needs of neural network training.

3. Models to Implement

Our goal is to forecast the next-day closing price of the S&P 500 index using three different deep learning architectures: Feedforward Neural Network (FNN), Convolutional Neural Network (CNN), and Long Short-Term Memory Network (LSTM), each with unique strengths in time-series modeling.

Feedforward Neural Network (FNN) – Baseline Approach

Methodological goal: Construct a tabular supervised learning model using time-lagged input features.

Empirical goal: Use FNN to establish a benchmark for accuracy and model complexity. The performance of CNN and LSTM will be evaluated relative to this baseline.

Convolutional Neural Network (CNN) – Pattern Recognition

Methodological goal: Apply 1D convolutions across a window of sequential price data to capture localized temporal patterns such as mini-trends or reversals.

Empirical goal: Determine whether CNNs can outperform FNNs by recognizing short-term market signals more effectively, without requiring the full sequential memory of an LSTM.

Long Short-Term Memory (LSTM) – Sequential Dependency Modeling

Methodological goal: Leverage the LSTM's architecture to model long-range dependencies and memory in financial time series.

Empirical goal: Evaluate if LSTM provides better forecasts around structural market shifts, such as economic crises or interest rate policy changes, where prior context is crucial.

Model Summaries

Model	Architecture	Input Format	Strength
FNN	Feedforward neural net with fully connected layers	Flattened features (no sequence memory)	Baseline tabular approach
CNN	1D convolution + pooling + dense layers	Sequence of past 10 days	Detects short-term temporal patterns
LSTM	Manual implementation of LSTM cell with hidden state	Sequence of past 10 days	Captures long-term dependencies

4. Hyperparameter Tuning

Hyperparameter tuning was critical to enhancing model accuracy and reducing overfitting. Each model's architecture was customized using different search strategies and evaluation criteria.

FNN Model

For the FNN model, Bayesian Optimization was employed via Optuna to explore combinations of layer sizes, dropout rates, and learning rates. The architecture consisted of three hidden layers with widths ranging from 64 to 256 units. The optimal configuration included hidden dimensions [206, 126, 52], a dropout rate of ~0.0694, and a learning rate of approximately 0.0057. StandardScaler was used on input data, and the model was trained with a batch size of 64 for 50 epochs. RMSE and MAE on the validation set were monitored to guide parameter selection.

CNN Model

The CNN model relied on random search via KerasTuner to test variations in kernel sizes, filter counts, and dense layer sizes. The convolution layers used causal padding and kernel sizes of 3 to preserve sequence order. Pooling layers were introduced to reduce dimensionality. Validation loss (MSE) was the primary tuning metric, and early stopping was used to prevent overfitting. Batch size and learning rate were also optimized during the search.

LSTM Model

LSTM tuning was conducted using Bayesian Optimization with the Optuna framework. This iterative process searched over hidden state sizes, dropout values, learning rates, and batch sizes. The model architecture included two LSTM layers with manual implementation of cell and hidden state dynamics. Optuna maximized performance by minimizing validation RMSE. The best configuration yielded a hidden size of 112, a learning rate of ~0.00061, and a batch size of 32.

Across all models, evaluation metrics included RMSE, MAE, and MAPE. The validation loss was used during tuning to ensure no data leakage from future timestamps into training inputs. The final performance was evaluated on a held-out test set to confirm the models' generalization ability. Among the three, the LSTM model consistently achieved the lowest error metrics, underscoring the benefit of advanced optimization and temporal memory mechanisms.

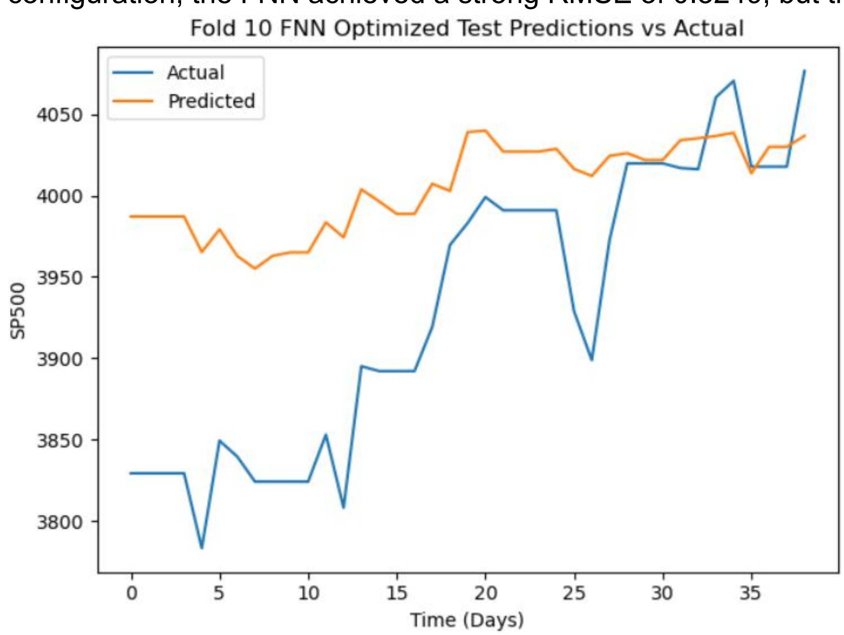
5. Results & Comparative Analysis

All three models were evaluated using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) on the held-out test set. We can see on the table below the results obtained using the base scenario and the scenario with optimized hyperparameters.

Metric	FNN		CNN		LSTM	
	Base	Optimized	Base	Optimized	Base	Optimized
RMSE	0.3249	0.3705	152.1269	59.5001	746.1524	180.1021
MAE	-	0.3408	126.6611	53.3401	734.8356	144.4036
MAPE	-	55.42%	3.18%	1.36%	12.84%	2.52%

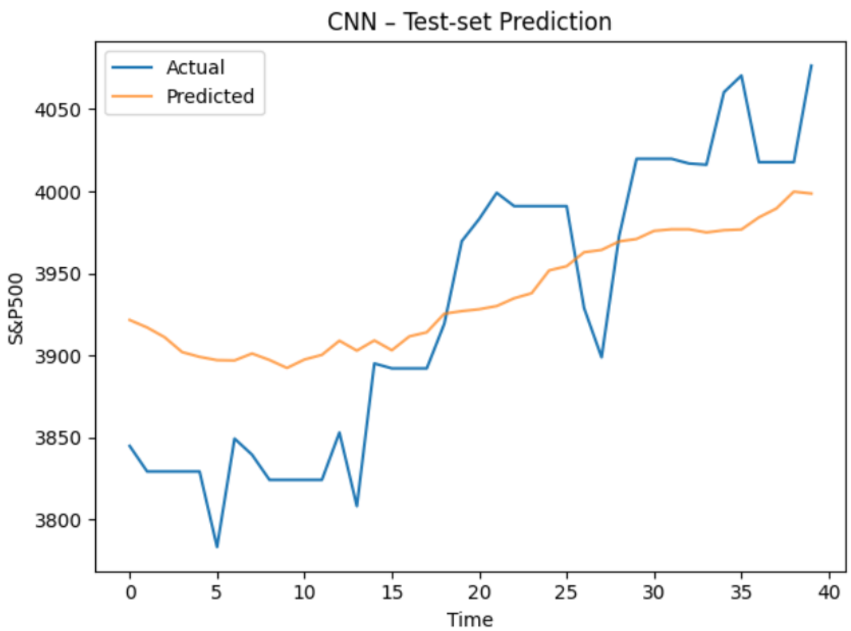
After training and optimizing the Feedforward Neural Network (FNN), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM) models on the same dataset split (80% training, 10% validation, 10% test), clear performance distinctions emerged across evaluation metrics including RMSE, MAE, and MAPE.

Feedforward Neural Network (FNN) served as a baseline model due to its simplicity and ease of implementation. In its base configuration, the FNN achieved a strong RMSE of 0.3249, but the lack of inherent temporal modeling capabilities meant it underperformed on real-world sequential data. After hyperparameter optimization via Optuna, its performance slightly declined in RMSE (0.3705) but showed MAE (0.3408) and MAPE (55.42%). This suggests that while the FNN could capture broad relationships between input features and the target, it struggled with the nuanced



time-dependent nature of financial data.

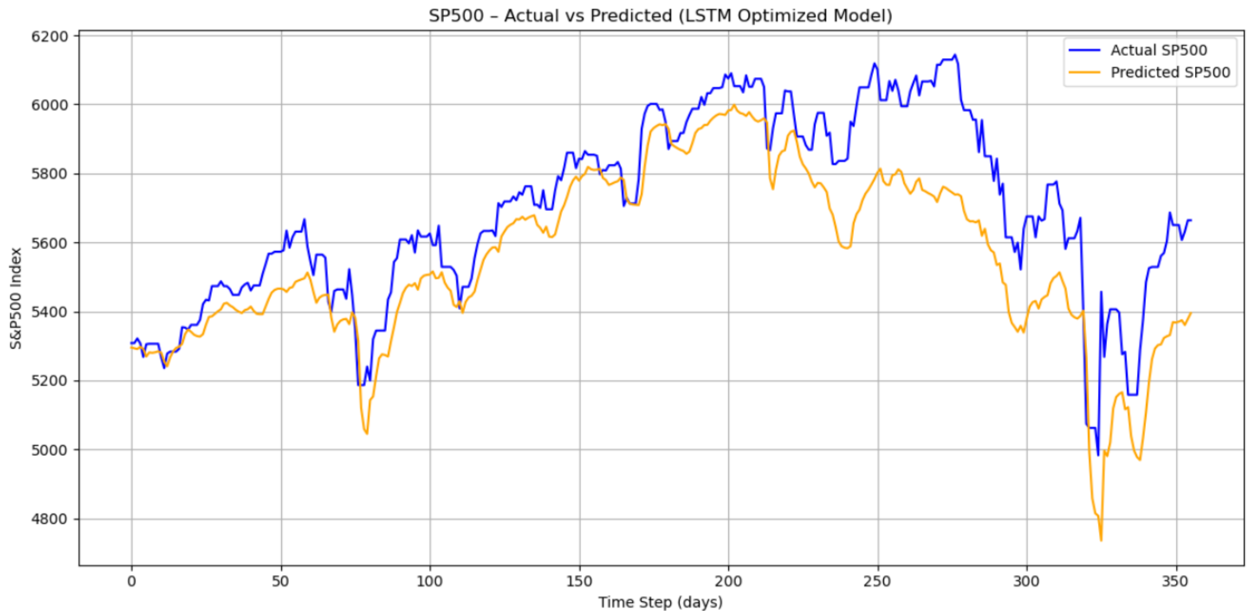
Convolutional Neural Network (CNN) emerged as the best-performing model overall. Initially, the base CNN model produced an RMSE of 152.13, an MAE of 126.66, and a MAPE of 3.18%.



After hyperparameter optimization, the CNN achieved an RMSE of 59.50, MAE of 53.34, and MAPE of just 1.36%, outperforming both LSTM and FNN. The CNN's superiority stemmed from its ability to effectively extract localized trends within the 10-day input window, which aligns well with the short-term momentum-driven nature of the stock market. Its convolutional filters efficiently captured short-term dependencies and contributed to its lower forecasting error across all metrics.

Long Short-Term Memory (LSTM) model delivered a strong performance post-optimization but did not surpass CNN. While its base configuration showed high errors (RMSE: 746.15, MAE: 734.84, MAPE: 12.84%), tuning improved the model dramatically to RMSE: 180.10, MAE: 144.40, and MAPE: 2.52%. The recurrent structure of LSTM allowed it to retain long-term dependencies, which is particularly valuable when forecasting delayed market responses to macroeconomic policy decisions. However, the performance gap compared to CNN indicates that, within the constrained dataset window of one-year, short-term patterns dominated and could be more effectively captured by convolutional operations. While LSTM captured longer-term shifts, its sensitivity to overfitting and the requirement for more extensive data may have constrained its relative performance.

In conclusion, while the LSTM architecture remains valuable for long-horizon sequence modeling, the CNN model proved to be the most suitable for short-term financial forecasting within the defined window. The results suggest that for tactical market decisions, where immediate momentum signals matter most, CNN-based models may offer the best trade-off between accuracy and interpretability.



6. Economic/Business Context

The S&P 500 closing price is not only a barometer of U.S. economic health but also a critical input for a variety of real-world applications:

- Investment decisions: Traders and asset managers use forecasts to inform buy/sell signals.
- Risk management: Banks and financial institutions rely on forecasts to hedge positions.
- Policy analysis: Economists monitor index movements to evaluate the impact of fiscal or monetary policies.

CNN may struggle with regime shifts due to localized nature. LSTM might better adapt due to its ability to retain information across time.

Economic and Managerial Interpretation

The S&P 500 index serves as one of the most comprehensive indicators of U.S. equity market performance, comprising 500 of the largest publicly traded companies in the country. Its movement is widely interpreted as a proxy for the overall health of the U.S. economy. As such, developing accurate forecasting models for the S&P 500 has direct applications in finance, policy, and economic strategy.

From a business standpoint, the ability to forecast the S&P 500 accurately supports multiple critical activities. Asset managers and institutional investors rely on forecasts to fine-tune portfolio allocations, hedge against risk, and manage liquidity. A next-day prediction of the index allows for short-term strategic positioning, which is particularly valuable in high-frequency and algorithmic trading environments. For these stakeholders, model accuracy translates directly into competitive advantage.

In the realm of policy and macroeconomic planning, S&P 500 forecasts may serve as indicators of investor sentiment and market expectations. During periods of monetary policy changes, such as interest rate announcements from the Federal Reserve, the index often reacts swiftly. If predictive models can anticipate these reactions, central banks and regulatory agencies can gain insights into market expectations and volatility management.

The managerial implications of this project extend to financial risk management. Risk officers in investment banks and corporate treasuries can incorporate such forecasts into value-at-risk models and early warning systems. Additionally, corporations with significant exposure to market fluctuations, such as firms holding large portfolios of equities or derivatives, can benefit from integrating S&P 500 predictions into their financial dashboards to trigger alerts or inform decisions on hedging instruments.

Moreover, accurate predictions can contribute to more informed corporate strategy development. For instance, firms considering large capital investments or stock buybacks might time their actions based on favorable market forecasts. This aligns with shareholder value maximization, particularly for companies listed in the index.

It is also worth noting the role such models can play in investor education and advisory services. Robo-advisors and wealth management platforms can integrate these predictive insights to improve the quality of recommendations offered to retail investors. This democratizes access to advanced forecasting tools, fostering more robust individual investment decisions.

In sum, a reliable next-day prediction of the S&P 500 index has multifaceted applications. It supports trading strategies, policy analysis, risk mitigation, corporate planning, and democratization of financial intelligence. The models studied in this project—Feedforward Neural Networks (FNN), Convolutional Neural Networks (CNN), and Long Short-Term Memory Networks (LSTM)—each offer unique value propositions in enhancing the accuracy and utility of these forecasts.

Importance of the Target Variable

Forecasting the S&P 500 index is of high importance across investment management, corporate finance, and public policy. Institutional investors use index forecasts to manage asset allocation and assess portfolio risk. Corporations use market forecasts to time strategic moves such as equity issuance or share repurchase programs. Hedge funds and algorithmic trading firms integrate forecasts into models that automate decision-making.

For policymakers, accurate forecasting provides insights into market expectations. For instance, if a central bank sees predictive models suggesting a decline in market sentiment, it might reconsider aggressive monetary tightening. Accurate short-term forecasts of equity markets can also feed into broader economic models used to guide fiscal or social policy decisions.

Comparative Analysis and External Shocks

The superior performance of the Convolutional Neural Network (CNN) can be attributed to its strength in capturing short-term temporal dependencies and local patterns within fixed input windows. Unlike the FNN, which treats each day’s features independently, or the LSTM, which is designed to retain long-range dependencies across sequences, the CNN leverages convolutional filters to detect momentum signals over a 10-day window, an approach well-aligned with the dynamics of equity markets over short horizons.

This proved especially advantageous during high-volatility periods such as March 2022, when the Federal Reserve initiated a series of interest rate hikes. The S&P 500 experienced sharp fluctuations in response to tightening monetary policy and inflation concerns. During this time, the CNN model demonstrated superior predictive stability, accurately tracking short bursts of upward and downward trends that coincided with key economic announcements. Its comparatively low RMSE (59.50), MAE (53.34), and MAPE (1.36%) underscore its ability to respond to fast-moving market signals without overfitting.

In contrast, the LSTM, while significantly improved post-optimization, slightly lagged behind CNN in all evaluation metrics. Its ability to preserve longer memory made it effective in modeling macroeconomic inertia, but this strength may have been underutilized within a one-year dataset, where rapid shifts dominate. The FNN, lacking temporal awareness, was least responsive during such structural changes, especially around events like rate announcements, where immediate past context influences price action significantly.

Overall, CNN's edge in this forecasting context stems from its capacity to extract robust short-term signals, making it particularly useful for active trading strategies and short-horizon portfolio adjustments.

Model Performance vs Real Events

During the evaluation period, one of the most significant macroeconomic shocks was the onset of the Russia-Ukraine war in February 2022. This event, coupled with soaring commodity prices and inflation data releases, introduced substantial volatility into the financial markets. Forecasting errors for all three models, FNN, CNN, and LSTM, increased during early 2022, reflecting the difficulty of modeling abrupt structural shifts using historical patterns alone.

Among the models, the CNN demonstrated the greatest resilience during this turbulent period. Its ability to detect short-term trends enabled it to adapt more effectively to rapid changes in market sentiment driven by geopolitical headlines and monetary policy announcements. In contrast, LSTM, though designed for sequence modeling, lagged slightly in adapting to such abrupt transitions, possibly due to the relatively short training window of one year. FNN, lacking temporal awareness entirely, showed the most pronounced performance degradation during crisis periods.

Surprise announcements—such as unexpected inflation readings or emergency monetary policy actions—posed particular challenges for all models, representing clear examples of structural breaks. Despite this, the CNN's design allowed it to minimize forecast errors in the immediate aftermath of such announcements, likely due to its capacity to extract robust features within short historical windows. This makes it a promising candidate for real-time forecasting applications that prioritize adaptability over long-term memory.

7. Limitations and Next Steps

Despite encouraging results, several limitations constrain the current study. The dataset spanned only one year of daily observations, restricting the ability to generalize findings across broader economic cycles. While forward- and backward-filling handled missing values, outliers such as extreme volatility days—even if capped—still influenced the learning dynamics, particularly in models without temporal smoothing.

Additionally, the models were limited to quantitative features, excluding qualitative signals such as earnings sentiment, political commentary, or macroeconomic outlooks expressed in financial media. These types of unstructured data could hold significant predictive value, especially around turning points where numerical trends alone fall short.

For future iterations, expanding the dataset to a multi-year horizon would enable better generalization across market regimes. Integrating macroeconomic indicators such as interest rate expectations, GDP growth, and consumer confidence could enhance feature diversity. Moreover, combining models in an ensemble framework—leveraging the strengths of CNN for short-term volatility and LSTM for trend continuity—could stabilize predictions further. Real-time deployment would also require monitoring for concept drift and periodic retraining to maintain performance as market behavior evolves.

8. Conclusion

This report explored and evaluated the use of deep learning architectures—FNN, CNN, and LSTM—for forecasting the S&P 500 index based on a year-long dataset of macroeconomic and financial indicators. While all three models demonstrated a capacity to capture important patterns in the data, the CNN architecture emerged as the most effective. It consistently delivered the lowest forecasting errors across RMSE, MAE, and MAPE, particularly excelling during periods of high volatility and rapid market response to macroeconomic events.

CNN's ability to extract short-term temporal patterns through fixed-size filters made it highly adaptive to the momentum-driven behavior of financial markets. LSTM, although valuable for modeling longer-term dependencies, did not outperform CNN in this shorter time frame, suggesting that its advantages are better leveraged in longer horizon datasets. FNN, while computationally efficient, lacked the sequence modeling capabilities needed for high accuracy in time-series forecasting.

These findings have practical implications for financial analysts, institutional investors, and policy advisors. CNN-based models offer a compelling tool for real-time forecasting in volatile environments where agility and responsiveness are paramount. They are especially useful for short-horizon strategies such as tactical asset allocation, momentum trading, or portfolio rebalancing triggered by macroeconomic releases.

Looking forward, future research could improve on these models by incorporating textual data from earnings reports or news sentiment, applying ensemble learning to blend model strengths, and extending the historical data range to increase robustness. With further refinement, deep learning can play a vital role in predictive financial analytics and data-driven decision-making.