LSTM for Financial Market Forecasting

LaTeX template adapted from:
European Conference on Artificial Intelligence

Alasteir Ho[1]

Other group members:
Arber Oruci,[2] Rafeeq Faraaz,[3] Nihaar Raut[4]

Abstract. This study uses Long Short-Term Memory (LSTM) neural networks to predict Apple Inc.'s (AAPL) stock prices using four years of historical data. Bayesian optimization fine-tuned hyper-parameters, resulting in a R² score of 0.92. The resulting model effectively captures sequential dependencies and trends, and was able to predict 5 trading days into the future. Future recommendations include integrating hybrid CNN-LSTM architectures and external data sources for improved adaptability.

1 Introduction

The prediction of stock prices is essential for developing investment strategies, managing risk, and analyzing market trends. Precise forecasts offer a competitive advantage in fluctuating markets affected by geopolitical factors, economic indicators, and investor sentiment. While traditional regression models are interpretable, they do not adequately account for the non-linear relationships and sequential dependencies that are intrinsic to stock price data. This study employs Long Short-Term Memory (LSTM) networks for stock price forecasting, illustrating their ability to effectively model temporal dependencies. Bayesian optimization significantly improves accuracy and makes the methodology suitable for realistic financial applications.

Regression serves as a building block for predictive modeling, where input features are used to estimate target variables. Although linear and polynomial regression models are frequently used, they do not correctly model stock price data. Thus, complex Neural Network models such as Long-Short Term Model (LSTM) offer a more promising solution.

2 Background

The task of stock price prediction presents unique challenges due to the characteristics of financial time-series data, including non-linearity, noise, and volatility.

[1] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, UK, email: ah5265z@gre.ac.uk
[2] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, UK, email: ao7316r@gre.ac.uk
[3] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, UK, email: ms4018w@gre.ac.uk
[4] School of Computing and Mathematical Sciences, University of Greenwich, London SE10 9LS, UK, email: nr5159g@gre.ac.uk

2.1 Limitations of Traditional Regression Models

Traditional regression models, such as linear regression and polynomial regression, are commonly used for prediction tasks. However, these methods fall short when applied to stock price forecasting for several reasons:

Linear Regression: Linear regression, while simple and interpretable, fails to model the complexities of stock price movements influenced by factors such as market sentiment, policies, and geopolitical events, which introduce non-linearity and volatility beyond its scope [20].

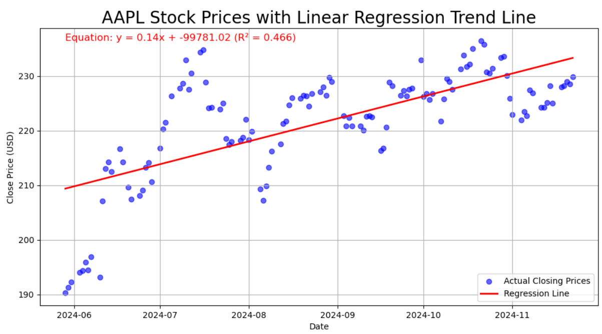

Figure 1. Linear Regression Trend on 6 months of AAPL Closing Stock Prices

Polynomial Regression: While introducing non-linearity through higher-degree terms, it suffers from over-fitting and sensitivity to noise. [20].

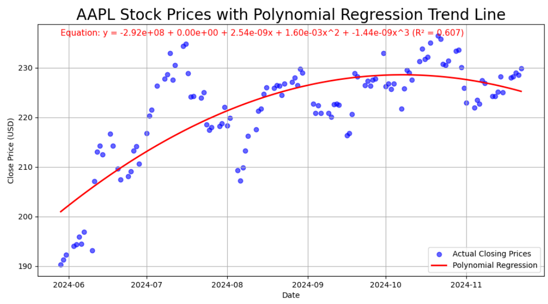

Figure 2. Polynomial Regression Trend on AAPL Closing Stock Prices

These limitations necessitate the need for advanced models which are capable of capturing the non linear and dynamic nature of financial markets.

2.2 Recurrent Neural Network and LSTM

Recurrent Neural Networks (RNNs) are a significant advancement in sequential data handling, as they feature self-looping; a hidden layer that remembers previous inputs to influence predictions [1].

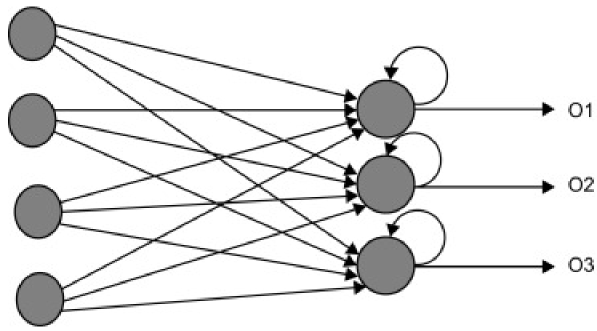

Figure 3. Simplified RNN model with self-looping diagram [12]

Challenges Faced by RNNs during training:

- **Exploding Gradient:** During training, gradients used to update weights may grow uncontrollably, leading to instability and convergence issues [10].
- **Vanishing Gradient:** Gradients can diminish exponentially, limiting the model's ability to learn long-term dependencies and resulting in poor performance on extended sequences [5].

2.3 Introduction to LSTM:

Long Short-Term Memory (LSTM) networks, introduced by Hochreiter and Schmidhuber [6], addresses limitations faced by RNNs by incorporating specialized memory cells and gating mechanisms:

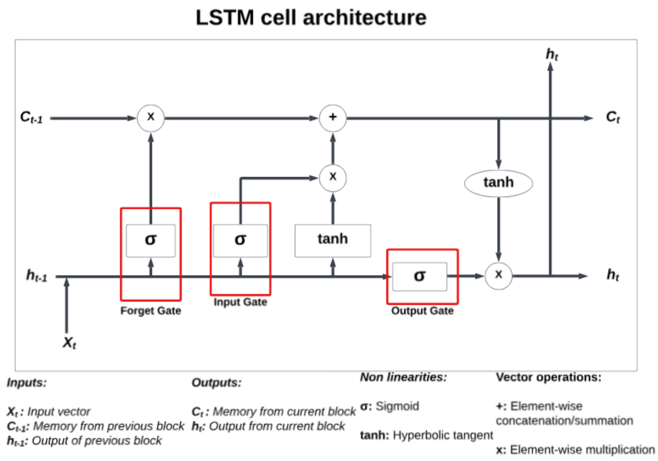

Figure 4. LSTM cell architecture with forget, input, and output gates regulating the flow of information in sequential data.

Components of LSTM:

- **Forget Gate:** Decides which information from the previous memory cell to discard.
- **Input Gate:** Determines which new information to add to the memory cell.
- **Output Gate:** Generates the final output for the current time step based on the updated memory state.

2.4 Application of LSTMs for time series modeling

LSTMs has proven to be particularly effective for stock price prediction due to its ability to model sequential dependencies and handle noisy data. Key advantages include:

- **Resistance to Over-fitting:** Regularization techniques such as dropout reduces the risk of over-fitting, even when trained on noisy financial data [4].

- **Capturing Long-Term Dependencies:** The gating mechanisms of LSTMs allow the network to retain information from distant time steps, enabling it to capture meaningful trends [2].

- **Scalability:** LSTM models can be extended by incorporating additional features such as trading volume and technical indicators, further enhancing their predictive power.

3 Experiments and results

3.1 Data-set

This study uses the daily closing price of Apple Inc. (AAPL), downloaded from Yahoo Finance using the Python library *yfinance* [19]. The data set contains different attributes such as open price, highest price, lowest price, close price, and volume of trade over four years. However, this exploration focuses only on the closing price, the last recorded price of each day, as it is commonly used for pattern analysis and forecasting.

The data-set was preprocessed to prepare it for training and validation:

1. **Normalization:** Closing Prices were normalized between 0 and 1 using the MinMaxScaler to facilitate faster convergence and ensure numerical stability during training.
2. **Splitting the Dataset:**
   - The data set was divided into training and validation sets in an 80:20 ratio. 6 months dedicated for model validation and 24 months for training.
   - The training set consisted of data six months prior to the date today, while the validation set includes the most recent six months.

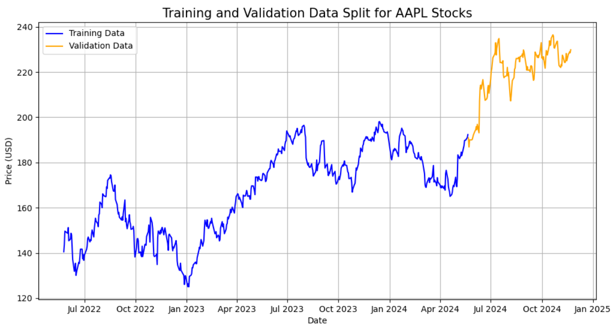

Figure 5. Training and validation data split on historical data of AAPL stock price

3. **Sequence Preparation:**
   - Training sequences were created using a sliding-window approach with 30-day 'look-back' periods of daily closing prices. Allowing the effective capturing of temporal dependencies.

3.2 Model Architecture

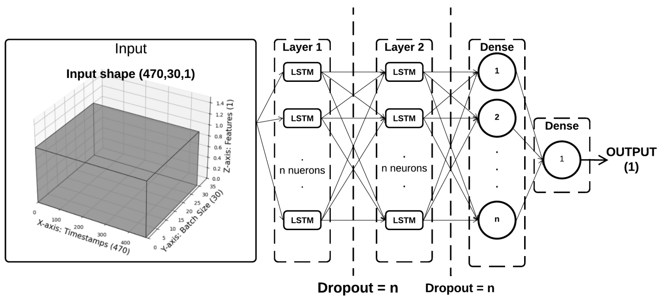

Figure 6. Proposed Model for stock price prediction

The proposed model consists of:

- **Two LSTM Layers:** To capture sequential dependencies.
- **Dropout Layers:** Applied between each LSTM layer to mitigate over-fitting.
- **Dense Layer:** A fully connected layer, connected to the 2$^{\text{nd}}$ LSTM layer
- **Output Dense layer:** Single dense unit, to output the next day stock price.
- **Batch size:** 30 sequences to be processed before weight changes.
- **Activation Function:** The Tanh activation function was applied to the first LSTM layer to introduce non-linearity [20]. Its bounded nature ensures smoother gradients, enhancing learning stability in sequential models.

The model was compiled with the Adam optimizer with the loss function set as Mean Squared Error (MSE).

3.3 Hyper-parameter tuning:

Bayesian Optimization was utilized to tune the hyper-parameters at 50 epochs for 30 trials with the goal of achieving the lowest MAE validation loss. The best-performing model was then trained for 100 epochs to refine its performance and fully exploit its capacity for learning complex sequential dependencies. The tuning method is proven to work much better than random search hyper-parameter tuning in any machine learning problem involving a black-box optimization setting [16]. This probabilistic method efficiently balances exploration and exploitation to identify the optimal combination of the following hyper-parameters:

1. **1st LSTM Units:** 20-100 units per layer.
2. **2nd LSTM Units:** 50-100 units per layer.
3. **Dropout:** 0.1-0.5 per LSTM layer.
4. **Dense Units:** 10-50 units.
5. **Learning Rate:** $1e^{-4}$ to $1e^{-2}$, logarithmic scale.

3.4 Evaluation Metrics

- **Mean Squared Error (MSE)**

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_{real,i} - y_{pred,i})^2$$

MSE measures the average squared difference between the predicted and actual values. Lower values indicate higher accuracy.

- **Mean Absolute Percentage Error (MAPE)**

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{\text{real},i} - y_{\text{pred},i}}{y_{\text{real},i}}\right| \times 100$$

MAPE quantifies the prediction error as a percentage of the actual value, offering an intuitive interpretation of model accuracy.

- **Mean Absolute Error (MAE)**

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_{real,i} - y_{pred,i}|$$

MAE calculates the average absolute error, which provides a more direct measure of prediction accuracy.

- **R^2 score**

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_{\text{true},i} - y_{\text{pred},i})^2}{\sum_{i=1}^{n}(y_{\text{true},i} - \bar{y}_{\text{true}})^2}$$

R^2 indicates how well the model explains the variance in the data, with values closer to 1 reflecting better performance.

3.5 Performance Evaluation

The model achieved an R² score of 0.92, with an MSE of 9.89 and a MAPE of 1.02%. These metrics (figure 7 and 8) demonstrate the model's ability to capture trends and minimize errors. Figure 9 highlights the model's accuracy over the validation period.

Trials						
Hyperparameters	1	2	n	**26**	n	30
1st LSTM Units	20	50	...	**80**	...	80
1st Dropout	0.5	0.1	...	**0.1**	...	0.1
2nd LSTM Units	90	70	...	**70**	...	100
2nd Dropout	0.5	0.1	...	**0.1**	...	0.2
Dense Units	40	20	...	**40**	...	40
Learning rate	0.0069	0.0078	...	**0.0078**	...	0.0073
R^2 score	0.71	0.90	...	**0.92**	...	0.91
MSE	33.47	11.47	...	**9.89**	...	10.08
MAPE	2.27%	1.17%	...	**1.02%**	...	1.06%
MAE	5.04	2.59	...	**2.45**	...	2.53

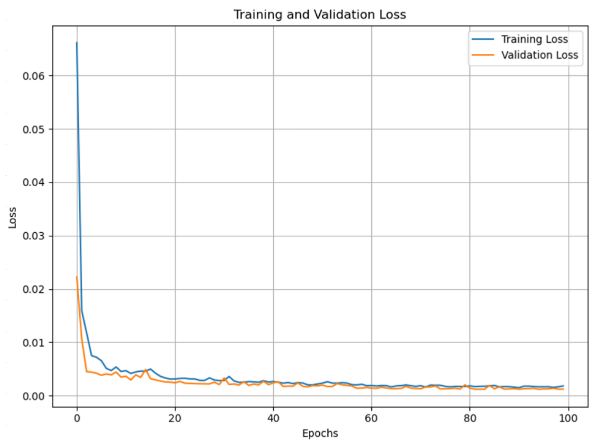

Figure 7. Training and Validation Loss (MSE) steadily converges over epochs, indicating no signs of over-fitting as the validation loss decreases consistently

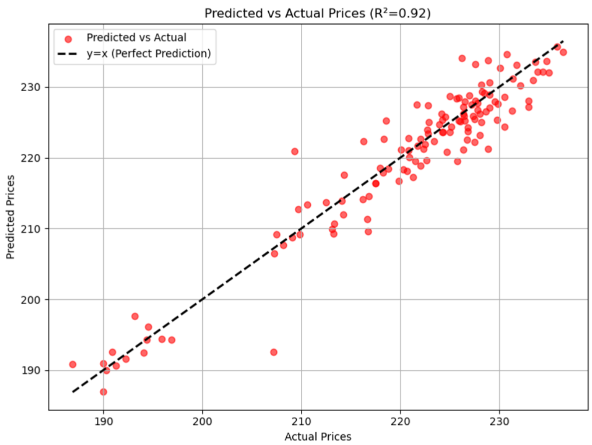

Figure 8. Scatter Plot Comparison of Predicted and Actual Stock Prices, indicating no signs of under-fitting

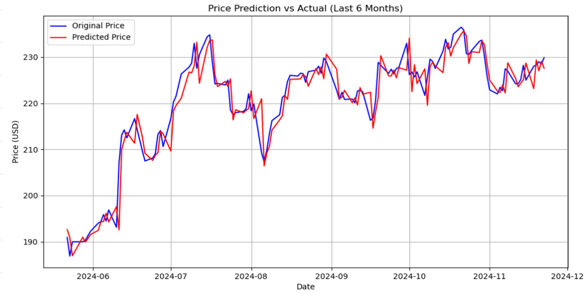

Figure 9. Predicted vs. Actual Stock Prices on the Validation dataset, showing the model's accuracy in capturing trends.

3.6 Future price prediction

The high level of accuracy imply that the model is capable of capturing sequential dependencies and trends effectively. Thus, the model is not only capable of predicting the next day's price but also for forecasting multiple days into the future. Using a recursive approach, the model was extended to predict 5 trading days into the future (does not take into account future holidays).

3.6.1 Results

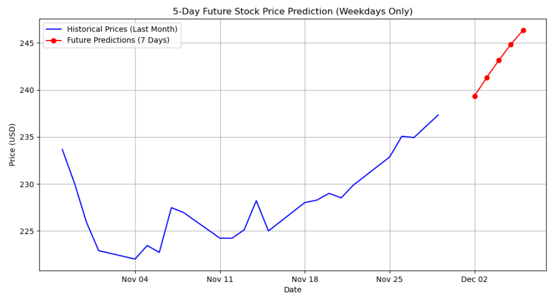

Figure 10. 5 Trading Days future Closing Stock Price Prediction

5 Trading Day Stock Price Predictions (Weekdays Only)

Date	**Predicted Price (USD)**
2024-12-02	$239.78
2024-12-03	$241.35
2024-12-04	$242.16
2024-12-05	$244.44
2024-12-06	$246.58

4 Discussion

Although using identical hyper-parameters, each run yields variability in results and performance, this is a notable challenge when utilizing neural networks. This variability arises due to stochastic elements such as weight initialization and dropout. This highlights key legal, social, ethical and professional considerations around deploying AI systems for financial forecasting.

4.1 Legal Considerations

In regulated markets, AI must comply with the laws that make trading fair and ensure proper usage of data to prevent manipulative practices such as insider trading or unfair advantages [18]. The lack of transparency in black-box models like LSTMs raises serious concerns about accountability in scenarios of market anomalies, which result in incorrect predictions in turn inuring losses.

4.2 Social and Ethical Concerns

The use of historical data to train artificial intelligence model may overlook the consequences of real-time external factors, such as geopolitical crisis, thus generating biased predictions [8].Additionally, larger models requiring more computational resources. Due to budgetary limitations, smaller firms and investors are excluded from the innovation.

4.3 Environmental Consideration

The significant computational cost of training neural networks contributes to a considerable amount of carbon emissions [17]. Microsoft's Three Mile Island data center [9] has found a solution to this; using low-carbon nuclear energy to support resource-intensive AI workloads.

This project demonstrates the potential for incorporating sustainable energy options within AI systems, which would lessen the environmental impact while satisfying growing computational needs in a world facing an energy crisis.

4.4 Professional Responsibility

Transparency, fairness, and reproducibility are critical for professional integrity in AI development. Documenting hyper-parameter tuning processes and fixing random seeds can enhance reproducibility [7]. Additionally, ensemble techniques could mitigate performance variability and foster greater trust in AI systems.

5 Conclusion and future work

The LSTM-based model achieved a high level of prediction accuracy, with a R^2 score of 0.92 and strong performance across metrics such as MSE, MAPE and MAE. Thus demonstrating the capability of the LSTM networks to model sequential dependencies effectively, capturing trends and patterns in stock price data. Despite challenges such as variability in performance results and limited performance during periods of extreme market volatility, the LSTM outperformed traditional regression techniques, justifying its application in financial modeling. However, the model's reliance solely on historical closing prices highlights the need to integrate additional contextual features for better robustness and adaptability.

Future Work: To further enhance predictive accuracy and robustness, future research should investigate hybrid architectures like CNN-LSTM. Convolutional Neural Networks (CNNs) are highly effective at extracting spatial features from structured data, while LSTM networks excel in capturing temporal dependencies. By combining these techniques, a hybrid CNN-LSTM model could provide a comprehensive approach to spatial and sequential feature learning, making it more effective for complex financial data [13, 15].

Additionally, advanced techniques like attention mechanisms could be integrated into the hybrid model to dynamically weigh the importance of different input features, further enhancing adaptability to volatile market conditions [14]. Future research could also incorporate external data sources, such as sentiment analysis [3], trading volume, and macroeconomic indicators, to address limitations in handling sudden market events and to improve the model's predictive capabilities [11]. These enhancements promise a more robust and accurate framework for stock price prediction.

REFERENCES

[1] Amazon Web Services. What is a recurrent neural network (rnn)?, 2023. Accessed: 2024-11-23.

[2] Yong Baek and Hyun-Yong Kim, 'Modaugnet: A new forecasting framework for stock market index', *Expert Systems with Applications*, (2018).

[3] Subhadeep Choudhury et al., 'Sentiment analysis for financial market prediction using deep learning', in *2022 IEEE 10th International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 120–124. IEEE, (2022).

[4] Soumya Bose Nilanjan Debnath Giridhar Maji, Anirban Ghosh, 'Stock price prediction using lstm on indian share market', *Proceedings of 32nd Indian Engineering Conference*, (2019).

[5] Sepp Hochreiter, 'The vanishing gradient problem during learning recurrent neural nets and problem solutions', *International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems*, **6**, 107–116, (1998).

[6] Sepp Hochreiter and Jürgen Schmidhuber, 'Long short-term memory', *Neural Computation*, **9**(8), 1735–1780, (1997).

[7] Anna Jobin, Marcello Ienca, and Effy Vayena, 'The global landscape of ai ethics guidelines', *Nature Machine Intelligence*, **1**, 389–399, (2019).

[8] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan, 'A survey on bias and fairness in machine learning', *ACM Computing Surveys (CSUR)*, **54**(6), 1–35, (2021).

[9] Microsoft, 'Microsoft powers ai with nuclear energy at three mile island data center', *Technology Review*, (2024).

[10] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio, 'On the difficulty of training recurrent neural networks', in *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pp. 1310–1318. PMLR, (2013).

[11] D Patnaik, NVJ Rao, and B Padhiari, 'Optimised hybrid cnn-lstm model for stock price prediction', *International Journal of Modelling and Data Mining*, (2024).

[12] Sandeep Kumar Satapathy, Satchidananda Dehuri, Alok Kumar Jagadev, and Shruti Mishra, 'Chapter 3 - empirical study on the performance of the classifiers in eeg classification', in *EEG Brain Signal Classification for Epileptic Seizure Disorder Detection*, 45–65, Academic Press, (2019).

[13] M Shah and D Vaidya, 'A comprehensive review on multiple hybrid deep learning approaches for stock prediction', *Intelligent Systems with Applications*, (2022).

[14] Z Shi, Y Hu, G Mo, and J Wu, 'Attention-based cnn-lstm and xgboost hybrid model for stock prediction', *arXiv preprint arXiv:2204.02623*, (2022).

[15] P Singh, M Jha, M Sharaf, and MA El-Meligy, 'Harnessing a hybrid cnn-lstm model for portfolio performance: A case study on stock selection and optimization', in *IEEE Conference on Innovations in Data Science and AI*, (2023).

[16] Jasper Snoek, Hugo Larochelle, and Ryan P Adams, 'Practical bayesian optimization of machine learning algorithms', in *Advances in Neural Information Processing Systems*, pp. 2951–2959, (2012).

[17] Emma Strubell, Ananya Ganesh, and Andrew McCallum, 'Energy and policy considerations for deep learning in nlp', 3645–3650, (2019).

[18] Paul Wilmott and David Orrell, *The Money Formula: Dodgy Finance, Pseudo Science, and How Mathematicians Took Over the Markets*, Wiley, 2017.

[19] Yahoo Finance. Yahoo finance library documentation, 2023. Available at: `https://pypi.org/project/yfinance/`.

[20] Guoqiang Zhang, Eddy Patuwo, and Michael Y Hu, 'Forecasting with artificial neural networks: The state of the art', *International Journal of Forecasting*, **14**(1), 35–62, (1998).