Dean&Francis

ISSN 2959-6157

Stock Price Prediction Models: From Multi-Scale Adaptive Networks to Large Language Models

Xiaoqiao Wang[*]

Wuhan Britain-China School,
Wuhan, China
[*]Corresponding author.
Email: janewang5386@gmail.com

Abstract:

The stock price prediction is an arduous process in the financial time series analysis as it is based on multi-scale and non-linear properties of market dynamics. Four recently developed algorithms are discussed in this article: the use of a Multi-Scale Adaptive Decoding Network (MSAD-Net) to identify long term trends and short term volatility, context dependent quantum neural network using quantum batch gradient updating and multi-task learning to predict multi asset distribution returns, classical machine learning models, such as SVMs, CNNs and XGBoost, and a combination between a large language model framework, self-reflection, and reinforcement learning to provide an explanation in a predictive approach. Results of experiments indicate that MSAD-Net can significantly decrease errors in its predictions, the quantum techniques allow efficient parameterized multi-asset correlations, the best overall balance is achieved through XGBoost and the quality of the explanations produced by LLMs is excellent. It follows that hybrid models and additional incorporation of unstructured information are potential areas of further development.

Keywords: Stock price prediction; Multi-Scale adaptive networks; Large language models.

1. Introduction

Predicting stock prices has been one of the most fundamental issues in the field of financial economics and quantitative investing [1]. Financial time series are naturally complex, non-linear and multi-scale, and are influenced by a diverse set of factors, including macroeconomic trends, policy shifts, market sentiment and short-term news shocks [2]. Precise prediction of stock prices is essential to asset allocation, risk management and portfolio optimization. Nonetheless, conventional statistical models like ARIMA and GARCH, which require linearity and stationarity, tend not to reflect the complex and multi-scale character of financial markets. This has led to more and more researchers adopting machine learning and deep learning methods to enhance forecasting results [3].

Over the past several years, numerous computational methods have been suggested as a means of predicting the prices of stocks [4,5]. Recurrent neural networks, and particularly Long Short-Term Memory (LSTM) networks, have demonstrated potential in learning long-term relationships, but they tend to be slow to react to short-term volatility. Local patterns are where convolutional neural networks (CNNs) perform best, but they do not offer a high-level view. Models based on transformers can learn the interactions at the global level, but are expensive to compute and might miss out on local changes. Even more recently, investigators have started researching more advanced paradigms like multi-scale adaptive networks, quantum machine learning, ensemble algorithms, and large language models (LLMs) to alleviate these drawbacks. As an example, multi-scale architectures try to integrate long-term trends and short-term fluctuations, whereas quantum neural networks exploit superposition and entanglement to possibly speed up learning and learn inter-asset correlations [6]. In the meantime, LLM-based frameworks bring interpretability and explain-ability, two areas in which conventional deep learning models have faced criticism as black boxes. Although these developments have occurred, there is no systematic comparison and integration of these new methodologies and the advantages and disadvantages of each methodology in various market situations and types of assets are not fully comprehended.

The goal of this paper is to review and compare four up-to-date methods of predicting stock prices, which are multi-scale adaptive decoding network (MSAD-Net), contextual quantum neural network with quantum batch gradient update and quantum multi-task learning, classical machine learning algorithms such as SVM, CNN, and XGBoost, and a large language model-based system with self-reflection and reinforcement learning. Namely, in Section 2, the author will present the multi-scale adaptive decoding network, which is able to reflect both long-term trends as well as short-term volatility using such modules as TDC, VPP and MSFA, and the author will assess its performance on the Dow Jones Industrial Average component stocks. In Section 3, the authors discuss a quantum neural network method based on context distribution loading, quantum single-task learning, and quantum multi-task learning to forecast the joint return distribution of S&P 500. In Section 4, the authors compare the performance of traditional machine learning models, including SVM, regression trees, and several types of neural networks (FNN, RNN, CNN and LSTM) with XGBoost using data on a large A-share stock (Kweichow Moutai). In Section 5, a large language model-based explainable prediction system is considered that combines summarization, self-reflection, and proximal policy optimization on US stocks with text data of tweets and historical prices. Lastly, Section 6 summarizes the main results of the paper, compares the respective strengths and weaknesses of all the methods, and offers some suggestions regarding further studies.

2. Stock Price Prediction Based on Multi-Scale Adaptive Decoding Network

The problem of predicting stock prices is both important and complex in finance due to the fact that price movements are inherently multi-scale. Sentiment and news shocks and policy changes produce short-term volatility, and macroeconomic conditions, monetary policy, and industry fundamentals generate long-term trends. Classical statistical methods (ARIMA, GARCH) rely on linear and stationary assumptions, which cannot describe non-linear multi-scale processes. The deep learning models also have their drawbacks namely: LSTM can capture the long-term interactions; however, it responds slowly to the short-term volatility; CNN can detect the local patterns, but it does not have the global view; Transformer can describe the global interactions, yet it is computationally expensive and does not take into account the local variations. The solution to these issues, the authors propose, is the multi-scale adaptive decoding network (MSAD-Net), which can take into consideration the long-term trends and short-term fluctuations at the same time. The experiments are carried out on 30 Dow Jones Industrial Average (DJIA) component stocks between August 21, 2019 and August 21, 2024 (1,259 trading days). They choose three features, i.e., open, close, high, low, and volume. The data is split chronologically with 70 percent being utilized in training and 30 percent being used to test [7].

MSAD-Net is a network with three modules and a multi-dimensional loss function that is run on a sliding window framework.

A prediction of the sliding window: The provided multivariate time series is $X = \{x_1, x_2, \ldots, x_T\}$, and the model uses previous d time steps $\left[x_{t+1}, \ldots, x_{t+d}\right]$ to predict the future step prices y_{t+d}.

TDC (Trend Dynamic Catcher): This module has the capability to learn trends on a long-term scale and high-frequency dynamics. It begins with an LSTM where the cell state update is defined as $C_t = f_t * C_{t-1} + i_t * C_t$ with f_t being the forget gate, i_t being the input gate and C_t being the candidate cell state. This gating system preserves long-range dependencies. An LSTM output is then passed to a multi-head attention layer: $\text{Attention}(Q,K,V) = \max\left(QK^{\top} / \sqrt{d_k}\right)V$, with final output y_f.

VPP (Volatility Pattern Probe): This module focuses on short-term fluctuations and local fluctuation. It is implemented using a 1D convolutional layer with ReLU activation: $C_t = \text{ReLU}\left(\text{Conv1D}\left(x_t, W_c\right) + b_c\right)$. The convolution kernel runs through time to generate feature maps indicating local volatility patterns. These features are processed by an attention layer to give temporal dependence: $\alpha_t = \text{softmax}\left(Q_t K^{\top} / \sqrt{d_k}\right)$. The output of VPP is the estimated output of N-th sample which means y_N.

MSFA (Multi-Scale Feature Aggregator): This block adds the long-term features of TDC ($\widehat{y_f}$), the short-term features of VPP ($\widehat{y_n}$), and the intrinsic features learned by an LSTM encoder ($\widehat{y_p}$) through element-wise multiplication: $y_t^h = \widehat{y_f} \otimes \widehat{y_n} \otimes \widehat{y_p}$. All together representation is passed to an LSTM decoder that outputs the final prediction: $\widehat{y_t} = f_{\text{LSTM}'}\left(W_d \cdot y_t^h + b_h\right)$.

MLOF (Multi-Dimensional Loss Optimization): To balance the learning of both scales, the loss function comprises three terms as follows: $L = \text{RMSE}\left(\widehat{y_t}, y_t\right) + \lambda\left(L_1 + L_2\right)$, with L_1 being the loss incurred by high-frequency features during a long period, whereas L_2 is the loss caused by short-period volatility features and λ is the balancing constant.

Measures of assessment include MSE, MAE, MAPE and R2. One example is $\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{y_i}\right)^2$.

Performance Prediction: In all the measures (MSE decline of 0.2779, MAPE decline of 10.79 percent, and R 2 increase of 0.2991), MSAD-Net has a higher performance than DA-RNN, LSTM, GRU, SVR, MLP and Informer.

Ablation study: Removal of TDC, VPP or MLOF reduces the accuracy and removal of MLOF is the most degrading.

Sensitivity of parameters: Stocks with a high level of volatility will have a lower T (2 or 5) while stocks with a low level of volatility will have a higher T (10 or 20). The optimal value of \lambda varies by stock.

Statistical significance: Diebold-Mariano tests confirm all improvements are significant ($p < 0.05$).

Flexibility: In a bull market, people should use large T to emphasize the trends and in a bear market, and they should use small T to focus on the volatility and in a crisis. They also need to shift λ to reduce the effect of an extreme loss. MSAD-Net is a reliable and versatile multi-scale method of predicting stock prices [7].

3. Stock Price Prediction Based on Contextual Quantum Neural Networks

The prediction of stock prices is an important part of asset allocation, risk evaluation, and portfolio optimization. Classical machine learning models are approaching the limits of classical computation in the analysis of large volumes of financial data. By using superposition and entanglement methods, quantum computers can reach faster speeds or greater accuracy. Quantum machine learning (QML) combines quantum state representations with predictive methods, and provides novel possibilities in financial time series prediction. However, the current NISQ devices are not without limitations. The majority of the currently implemented quantum prediction algorithms are based on full historical knowledge, are trained on one-asset one-task learning, do not consider the cross-asset correlation, and are undertrained. To alleviate this, the authors propose a contextual quantum neural network with Quantum Batch Gradient Update (QBGU) and Quantum Multi-Task Learning (QMTL) to predict the distribution of multi-asset stock prices. The four S&P 500 stocks examined in these experiments are Apple, Google, Microsoft and Amazon, which were later grown to eight (Pepsi, Western Digital, Texas Instruments, and IBM). Preprocessing of data is the process of converting the prices into returns (binary or quaternary quantization). Context length is T = 2 or 3, and it is predicted to be one-step-ahead return distribution.

The mathematical formulation: Let people assume that X t represents the return at time t. In the context of the given context vector $X^{(T)} = \left(X^1, \cdots, X^T\right)$ the model will predict the joint distribution $f\left(X^{(T+\tau)} -> X^{(T)}; X^{(T)}\right)$.

The returns are smoothed, differenced and discretized into d bins. In the case when binary ($d = 2$) is applied, $|0>= down$ and $|1>= up$. Context is represented as a vector of the form $| x^1 > \otimes ? | x^T >$.

Distribution loading in context: The overall distribution $\mathcal{P}\left(X^{(T)}\right)$ can be written as $\left|\psi^T ?\right.$. A hardware efficient ansatz using MSE loss and SPSA update is used. Unitary of each layer: $\hat{O}(\alpha, \beta) = \prod_j CNO\left(?_j R_Z\left(\beta^j\right)\right)\left(?_j R_Y\left(\alpha^j\right)\right)$.

QSTL (Quantum Single-Task Learning): A PQC uses as its inputs the context state $\left|\psi^{(T)}\right\rangle$ and an ancilla qubit |0?. The target output $\left|y^{(T+1)}\right\rangle$ is measured against the result of the SWAP test. The probability of state 0 is measured as $\mathcal{P}(0)=\left(1+\left|y^{(T+1)}\,|\,x^{(T+1)}\right|^2\right)/2$. Fidelity loss: $\mathcal{L}^{\gamma}=1-\mathcal{P}(0)$. SPSA calculates gradients: $\frac{\partial}{\partial\theta}\langle\hat{B}\rangle_\theta=\frac{1}{2\delta\alpha}\left(\langle\hat{B}\rangle_{\theta+\delta\alpha}-\langle\hat{B}\rangle_{\theta-\delta\alpha}\right)$.

QBGU (Quantum Batch Gradient Update): All context distributions are charged simultaneously in accordance with quantum linearity and consequently a single-pass gradient update can be made: $\theta=\theta-\beta\sum_{x^{(T)}}\mathcal{P}\left(x^{(T)}\right)g\left(\theta,x^{(T+1)}\right)$. It is similar to one full SGD pass through the dataset, which involves one forward pass.

QMTL (Quantum Multi-Task Learning): It is proposed to go with a share and specify ansatz. Each of the shared unitary $\widehat{U_s^l}\left(\theta_s^l\right)$ is applied to all assets. Label qubits control task-specific unitary $\widehat{U_{ck}^l}\left(\theta_{ck}^l\right)$ (e.g., $|0\rangle$ on Apple, |1? on Google). The quantity of label qubits employed is $logK+1$. The full circuit is $\hat{U}(\theta)=\prod_{l=1}^{L}\widehat{U_c^l\widehat{U_s^l}}$. There are shared parameters that can be used to indicate correlations between assets.

Prediction on sequential and portfolios: The conditional distribution is the output that will be produced by the model with the given context. Repetition R times gives R-step-ahead forecasts. Portfolio prediction is made by loading all weights into the distribution, i.e., weight distribution, so that all assets can be predicted simultaneously.

Experimental setting: Qiskit simulator, $T=3$ (or 2), binary/quaternary quantization, 3000 epochs, learning rate of 0.1, 0.01 SPSA perturbation, 10 000 shots. Observations: KL divergence and training loss.

The convergence loss in MSE represents the successful loading of return distributions onto 3-qubit states, while QSTL achieves accurate conditional distribution prediction with low KL divergence for Apple and Google. Comparing QMTL and QSTL, QMTL has the least KL divergence and converges fastest when using 2, 4, and 8 assets, with later-trained assets showing a small loss that indicates correlation capture (a regularization effect). QBGU significantly accelerates training via single-pass gradient updates, and the comparison of SWAP test with MSE shows that fidelity loss (SWAP) is comparable to MSE loss when combined with SPSA. In terms of scalability, parameter count remains constant from 2 to 8 assets, demonstrating good generalization, though noise strength increases KL divergence—depolarizing and readout noise both cause harm, with gate noise being more damaging.

To sum it all up, the contextual quantum neural network with QMTL and QBGU is a well-parameterized, fast training, correlation sensitive multi-asset stock price distribution prediction system that can be used as a foundation of quantum finance application.[8]

4. Stock Price Prediction Based on Machine Learning Algorithms

The development of the artificial intelligence has been accompanied by the growing use of machine learning algorithms in financial markets. Market data of stocks is massive with a fast pace and traditional econometric models cannot cope with multidimensional data. Nevertheless, machine learning can be very useful in predicting stock prices since it has powerful data mining and pattern recognition features. That is why the comparison of various machine learning approaches to predict stock prices has considerable theoretical and practical relevance. This paper is concerned with the prediction of stock prices based on the comparative analysis of predictive results of different popular machine learning models. The analysis involves the information provided by the largest non-bank stock by market capitalization in the Chinese A-share market, namely Kweichow Moutai (600519) and the time span analyzed is between January 4, 2022, and May 12, 2023. Data contains open price, close price, high price, low price, volume traded, price change and price fluctuation range.

The research is based on a comparative methodology that will examine four popular machine learning techniques, namely, support vectors machine (SVM), regression trees, various neural networks (such as feedforward neural network FNN, recurrent neural network RNN, convolutional neural network CNN, and long short-term memory network LSTM), and ensemble learning technique XGBoost. Principal component analysis (PCA) is applied at the initial stage in data preprocessing to choose the significant features, then both normalization and standardization are carried out. All models are assessed based on four factors, which include coefficient of determination (R^2), mean absolute percentage error (MAPE), root mean square error (RMSE) and mean absolute error (MAE), in order to identify the advantages and disadvantages of each model.

The findings indicate that there are differences between the models. Regarding goodness of fit (R^2) and relative error (MAPE), the SVM model is the best, with a value of 0.9416 and a relative error of only 0.0046 implying the highest fit and lowest prediction error. Based on absolute error measures (RMSE, MAE), the CNN model is excellent, achieving 10.2183 and 8.0308 respectively, which means the least difference between predicted and true values. Despite being the worst in any one measure, XGBoost has a remarkable R² (0.9270), MAPE (0.0052), RMSE (11.5803), and MAE (9.3175), which put it in very strong positions among all algorithms. The results indicate that the SVM model has the highest level of fit and the lowest relative error, CNN greatly minimizes absolute errors and XGBoost is the most superior model because it is the most comprehensive and well-performing. It is also found out by the study that in real-life quantitative investment other than prediction errors, risk components (e.g., Sharpe ratio), and transaction costs must be accounted for. Further development can also include more incorporation of external information (e.g., unstructured data including text and images), hyperparameter tuning, and ensemble methods to enhance robustness and applicability.[9]

5. Stock Price Prediction Framework Based on Large Language Models

The traditional methods of deep learning in stock prediction have been considered as black boxes because they cannot be interpreted and hence do not increase investor confidence. An alternative solution could be the Large Language Models (LLM) that have strong natural language understanding and generation capabilities. This paper will explore the process of using LLMs to generate an explainable stock prediction model.

Three modules are in the framework. Data: 55 US stocks in 11 sectors (2022-2023). Source: historic price data on Yahoo Finance and daily representative tweets on Twitter API 2.0.

The presented module can be summarized thus: An LLM M_X processes unstructured text in its normal form C_t^s and generates a factual summary $X_t^s = M_X\left(s, C_t^s\right)$.

Module (self-reflection agent): It is an approach that creates positive or negative examples of high quality with minimal manual labeling. Using stock s and historical summaries, LLM M_E returns response Y_t^s containing the price change y_t^s and the explanation e_t^s. Based on wrong prediction, the reflection LLM MR provides feedback that can be improved over time. Correct and incorrect examples are produced.

Proximal Policy Optimization (PPO): The optimization of the LLM with respect to self-generated data is defined as Proximal Policy Optimization (PPO). At the first stage, policy 1 is formed due to supervised fine-tuning (SFT). Then, a reward model r theta is trained on right/wrong pairs of answers to minimize cross-entropy losses:

$$L(\theta) = -E_{(X,Y_w,Y_l,s,t)\sim D}\left[log\left(\sigma\left(r_\theta\left(X_t^s, Y_{w,t}^s\right) - r_\theta\left(X_t^s, Y_{l,t}^s\right)\right)\right)\right]$$

. Finally, the best policy is to optimize policy based on PPO (policy gradient) so as to maximize the reward yet constrain the KL divergence

$$L(\phi) = -E_{\left(X,\hat{Y},s,t\right)\sim D_{\pi_\varphi^{RL}}}\left[r_\theta\left(X_t^s, \widehat{Y_t^s}\right) - \beta log \frac{\pi_\varphi^{RL}\left(\widehat{Y_t^s} \mid X_t^s\right)}{\pi^{SFT}\left(\widehat{Y_t^s} \mid X_t^s\right)}\right].$$

The experimental findings have demonstrated that the SEP-fine-tuned LLM is significantly more efficient than the traditional deep learning models (e.g., GRU-Att) and other LLM baselines (GPT-3.5, FinGPT). The higher MCC indicates that the SEP is of a better real sense. According to the explanation's quality, GPT-4 scores (information consistency, context understanding, positive /negative sample balance) are all supportive of SEP. Ablation experiments confirm the significance of the summary module, self-reflection, and PPO fine-tuning. This framework can be extended to other financial processes including the creation of portfolios. Future work plans include scaling it to larger scale datasets and using more data types and improving the quality of explanations [10].

6. Conclusion

The present paper has discussed and compared four different methods of predicting stock prices: a multi-scale adaptive decoding network (MSAD-Net), a contextual quantum neural network using QBGU and QMTL, conventional machine learning models such as SVM, CNN, and XGBoost, and an explainable model based on large language models. MSAD-Net performs well in terms of long-term trend and short-term volatility, has a high level of statistical significance, and is adaptable to market conditions. Quantum neural networks are efficient in terms of parameters and can be trained quickly using QBGU, and they can efficiently capture multi-asset correlations with the help of QMTL despite their vulnerability to gate noise. Among traditional ML models, SVM is the most relatively well-fitting model, CNN has the lowest absolute error, and XGBoost is the most balanced model, making it usable in practice for quantitative investing. LLM-based frameworks provide interpretability and high-quality explanations that are superior to both conventional deep learning and other LLM baselines, and have great potential in generalization.

However, several limitations remain. MSAD-Net needs optimal calibration of window size T and balancing factor λ depending on volatility regimes. Quantum models are constrained by current NISQ hardware limitations and noise susceptibility. Traditional ML techniques still rely on hand-engineered features and cannot represent complex temporal dynamics as well as deep or quantum models. LLM frameworks are computationally costly and require high-quality text and prompt engineering. Future studies should examine hybrid frameworks that combine the strengths of these methodologies, such as integrating multi-scale adaptive networks with LLM-based explanations or combining quantum layers with traditional deep learning pipelines. Additionally, expanding to broader asset classes, incorporating alternative data (e.g., satellite imagery, news sentiment), and assessing robustness in live trading conditions are important next steps.

References

[1] Chen L, Pan Z, Chen Q. Research on stock price prediction framework based on large language models. Mathematics in Practice and Theory, 2025, 55(12): 159-171.

[2] Tang X, Guo Q, Xu T, et al. Stock return prediction based on multi-scale kernel adaptive filtering. Journal of Computer Applications, 2023, 43(5): 1385-1393.

[3] Cheng M, Gao S. Multi-scale stock prediction based on deep transfer learning. Computer Engineering and Applications, 2022, 58(12): 249-259.

[4] Luo X, Zhang J. Stock price prediction based on multi-time-scale composite deep neural network. Wuhan Finance, 2020(9): 32-40.

[5] Zhang X, Gao Z, Wu L, et al. Stock price prediction based on hybrid quantum-classical neural network model. Journal of University of Electronic Science and Technology of China, 2022, 51(1): 16-23.

[6] Sharan Mourya, Leipold H, Adhikari B. Contextual quantum neural networks for stock price prediction. Scientific Reports, 2026, 16(1): 4454.

[7] Guan R. Research on quantitative investment stock selection based on machine learning. Chengdu: Southwestern University of Finance and Economics, 2020.

[8] Zhang D D, Jiang J J, Zhang G L, et al. The stock price prediction based on a multi-scale adaptive decoding network. International Journal of Machine Learning and Cybernetics, 2026, 17(2): 63.

[9] Liu M, Shin S Y. Adaptive multi-modal deep learning for financial market prediction: A multi-scale attention approach. Journal of The Korea Society of Computer and Information, 2025, 30(12): 297-304.

[10] Qin Y, Kong H. Stock price prediction based on machine learning algorithms. Journal of Shantou University (Humanities & Social Sciences Edition), 2025, 41(7): 25-34, 94-95.