Recurrent Neural Networks for Stock Price Prediction

Wei Yao
The University of Adelaide
ADELAIDE 5005 AUSTRALIA
wei.yao@student.adelaide.edu.au

Abstract

Given that financial time series data is complex and non-linear, predicting stock prices has been a formidable task.RNN and its extensions have done relatively positive ome, but because they are unable to capture long-range relationships and can not play a great role in the multi-dimensional interaction of the characteristics.In this paper, we propose an improved LSTM architecture with simple both temporal and feature attention mechanisms to enhance stock price prediction accuracy.The mechanism introduced in our model aid on two sides: to capture long-ranging inter-dependencies over time steps, temporal attention; and technical indicators of different importances should be weighted dynamically by feature attention.Additionally, residual connection and layer normalization techniques are applied to improve the convegence and overall performance of modelWe evaluate our approach using data for the Google stock price, comparing it with models of conventional RNN, GRU, and LSTM type.Experiments show that compared to standard LSTM models, our model has performed significantly better, resulting in a 59.7% reduction of Mean Squared Error (MSE) from 17.94 to 7.23 and 35.3% less Mean Absolute Error (MAE) from 3.23 to 2.09.The new architecture not just improved predictability and decreased error rate but also succeeded in providing interpreted attention weights, which gives available guides to pinpoints and practical advice for stock forecasting between different time periods.The implementation of this project can be found at `https://github.com/yyaaoo33/deeplearning-ass3.git`.

1. Introduction

In the financial markets, accurate prediction of stock prices has always been a major problem. Academia and industry alike show great concern for it. The complicacy of the stock market - high volatility, non-linearities and temporal dependencies - makes accurate prediction particularly challenging. This also poses a challenge to traditional statistical methods and machine learning algorithms, which often fail to capture the intricate patterns of stock price movements. Their long-term dependency, meanwhile, results in suboptimal prediction performance.

1.1. Background and Challenges

Stock price movements are influenced by numerous factors, including market sentiment, company performance, economic indicators, and global events. The temporal nature of stock data presents unique challenges:

- **Non-linear Patterns:** Stock prices exhibit complex non-linear relationships across different time scales, making traditional linear models inadequate for accurate prediction.
- **Long-term Dependencies:** Market trends often depend on historical patterns spanning various time periods, requiring models to effectively capture both short-term and long-term dependencies.
- **Feature Interactions:** Technical indicators and market variables interact in complex ways, necessitating sophisticated approaches to model these relationships effectively.

In recent years, deep learning has made exciting progress in time series prediction. Recurrent Neural Networks (RNNs) and their descendants are famous examples. The classical RNN structure is powerful for handling sequential data but it often runs into the vanishing gradient problem when sequences are too long. The Long Short-Term Memory (LSTM) networks address this limitation through their gating mechanisms, enabling better capture of long-term dependencies. Gated Recurrent Units (GRU) offer a simpler alternative to LSTM, reducing meaningful complexity in exchange for similar performance.

1.2. Limitations of Existing Approaches

Current approaches to stock price prediction face several key limitations:

- Vanilla RNNs struggle with long-term dependencies, achieving suboptimal MSE of 26.92 and MAE of 4.35 in our experiments. While they can process sequential data, their simple architecture limits their ability to capture complex temporal patterns.
- Standard LSTM models, despite their sophisticated gating mechanisms, still show limitations in feature utilization, resulting in relatively high MSE (17.94) and MAE (3.23). Their uniform treatment of temporal information fails to distinguish the relative importance of different time steps.
- GRU architectures, while computationally efficient (MSE: 13.96, MAE: 3.05), lack mechanisms to effectively weight the importance of different technical indicators and temporal patterns simultaneously.

1.3. Our Contributions

To address these limitations, we propose several key contributions:

- We introduce a novel dual-attention mechanism that combines temporal attention for capturing long-range dependencies and feature attention for dynamic weighting of technical indicators.
- We implement an improved LSTM architecture incorporating residual connections and layer normalization, significantly enhancing model stability and prediction accuracy.
- We demonstrate substantial performance improvements over baseline models, achieving an MSE of 7.23 and MAE of 2.09, representing reductions of 59.7% and 35.3% respectively compared to standard LSTM.
- We provide comprehensive experimental analysis and ablation studies, offering insights into the effectiveness of different architectural components and their contributions to prediction accuracy.

The remainder of this paper is organized as follows: Section 2 reviews related work in stock price prediction and attention mechanisms. Section 3 details our proposed methodology. Section 4 presents experimental results and analysis. Section 5 discusses implications and limitations, followed by conclusions and future work in Section 6.

2. Related Work

2.1. Traditional Time Series Methods

In recent decades, the field of financial time series forecasting has undergone substantial change. Introduced in the GARCH model as an extension of traditional time series analysis, volatility clustering was finally put on the agenda for serious consideration in financial markets [2]. Although this statistical approach established a rigorous foundation for analyzing market volatility, it carried with it a certain assumption of linearity that in reality compromised its power to catch the complex and non-linear dynamics of contemporary financial markets.

Support Vector Machines (SVM) has emerged as a promising alternative for financial forecasting. Adjusted from its traditional format, an SVM framework specially designed for financial time series uses adaptive margins to deal with the non-stationary nature of market information [12]. This approach has particular power in capturing local patterns and dealing with outlying items. However, SVM-based techniques face significant challenges when it comes to parameter optimization and selection of kernels [9]. This makes them less adaptable to rapidly changing market conditions.

The rise of ensemble methods–in particular the Random Forest–offers new prospects for financial prediction. The methods showed their capability to predict price movement directions effectively and are not uncompromisingly a black box like model [8]. Still, tree-based methods bring the prospect of missing, catching instead in principle only. Complex temporal dependencies remain a challenge and extensive experience with domain expertise is required when practising feature engineering.

2.2. Deep Learning Approaches

Deep learning revolutionized stock prediction methodology. For example, price movements were viewed as image-like patterns and treated by Convolutional Neural Networks (CNNs) in a completely original way [14]. This approach held promise in capturing local temporal patterns but grappled with varying time scales and long-term dependencies in financial data.

LSTM networks constituted a significant advance in the treatment of sequential data. Their architecture–which can store internal memory states–was well equipped to capture long-term trends on the stock market [4]. However, traditional LSTM architectures treat all historical data points as having equal weight. They don't distinguish between different temporal information's relevance over time, and this is a weakness [1]. Advanced deep LSTM architectures with multiple layers demonstrated stronger power for hierarchical feature extraction [6]. Although this approach showed improved ability for recognizing patterns, it presented new problems: the very complexity of the model increased stability in training and the odds of overfitting. These disadvantages were particularly acute in the noise-ridden financial industry.

2.3. Attention Mechanisms in Financial Forecasting

The introduction of attention mechanisms revolutionized the problem of sequence modeling via the Transformer architecture [13]. The concept was applied to stock prediction by generating a sort of time-based attention mechanism that would alter focus, according to the rules set by participants of particular historical moments [7]. We verified the urgent need of selecting temporal focus, but at the same time with the complexity issue of large computational load in practice.

Advances in this field went further still, proposing a hybrid attention network actually taking into account both time and feature-level relationships at once [3]. This architecture showed particular strength in terms of handling multi-source input, but the increased complexity of the model posed challenges for its practical use. Dual-stage attention further refined this idea, modeling the temporal dependencies and feature relationships separately [11]. But the question remained: The commercial deployment of such complex approaches has many practical challenges.

2.4. Technical Analysis Integration

The past few years have seen increasing efforts to bridge traditional technical analysis and deepest latest machine learning techniques. Integrating technical indices with depth neural network architectures showed how traditional market wisdom could bring model improvements [5]. However, static feature fusion in these works failed to change out of tune with market environments.

This defect was rectified through an adaptive feature fusion mechanism, which adjusted its technical indicators on the basis of what was happening now in the market [10]. The temporal fusion structure had shown early promise in terms of maintaining interpretability whilst coping with a variety of different predictable horizons. But up until now, this approach's computational demands are far too demanding to be practicable in real-time trading, signaling the continuing tug between model sophistication and practical applicability.

This rich tapestry of progress in stock prediction methods is emblematic at each turn of ever more sophisticated architectures that can account for the complex dynamics of markets while maintaining interpretability. Our job uses this work as a base, with particular attention to the constraints of current approaches concerning dynamic feature importance in computational efficiency.

3. Methodology

3.1. Network Architecture Overview

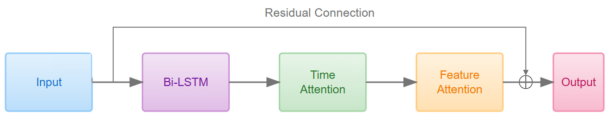

Figure 1. ATT-LSTM network structure diagram

We improved bidirectional LSTM architecture. The time information is captured by weighted summing of all time steps in the time series, and the input features are weighted element by element to obtain the overall feature information of the data to implement the attention mechanism,dual simplicity attention mechanisms help this structure predict stock prices. This model architecture . includes five main components: a feature transform layer, bidirectional LSTM , a time attention module, a feature attention module as well as a prediction head with residual connections for stabilization against vanishing gradients over long chains of nonlinear functions. This architecture is designed to satisfactorily capture both the long-term dependencies and feature interactions, whereas still ensuring stable training dynamics. Given an input sequence $X \in \mathbb{R}^{B\times T\times D}$, where B is the batch size, T is the sequence length, and D is the input feature dimension, our model processes the data through multiple specialized layers to generate predictions $Y \in \mathbb{R}^{B\times O}$, where O represents the output dimension.

3.2. Feature Projection and Encoding

The initial feature projection layer maps the input features to a higher-dimensional space through a linear transformation:

$$H_0 = XW_p + b_p \tag{1}$$

where $W_p \in \mathbb{R}^{D\times H}$ and $b_p \in \mathbb{R}^H$ are learnable parameters, and H is the hidden dimension. The bidirectional LSTM encoder processes the projected features in both forward and backward directions:

$$\overrightarrow{H_t} = \text{LSTM}_f(H_0, t), \quad \overleftarrow{H_t} = \text{LSTM}_b(H_0, t) \tag{2}$$

The concatenated hidden states $H = [\overrightarrow{H}; \overleftarrow{H}]$ are normalized using layer normalization to stabilize training:

$$\hat{H} = \text{LayerNorm}(H) \tag{3}$$

3.3. Temporal Attention Mechanism

The temporal attention module learns to focus on relevant time steps in the sequence. For each time step t, we

compute attention weights α_t through a two-layer neural network:

$$e_t = v^T \tanh(W_a \hat{H}_t + b_a) \tag{4}$$

$$\alpha_t = \text{softmax}(e_t) = \frac{\exp(e_t)}{\sum_{j=1}^{T} \exp(e_j)} \tag{5}$$

The context vector c is then computed as a weighted sum of the hidden states:

$$c = \sum_{t=1}^{T} \alpha_t \hat{H}_t \tag{6}$$

3.4. Feature Attention Mechanism

The feature attention module dynamically weights different feature dimensions based on their importance. It employs a bottleneck structure to efficiently learn feature relationships:

$$F_1 = \text{ReLU}(W_1 c + b_1) \tag{7}$$

$$\beta = \sigma(W_2 F_1 + b_2) \tag{8}$$

where β represents feature attention weights, and σ is the sigmoid activation function. The attended features are computed through element-wise multiplication:

$$c_{att} = c \odot \beta \tag{9}$$

3.5. Prediction Head and Residual Connection

The prediction head consists of multiple fully connected layers with dropout for regularization:

$$z_1 = \text{ReLU}(W_3 c_{att} + b_3) \tag{10}$$

$$z_2 = \text{Dropout}(\text{ReLU}(W_4 z_1 + b_4)) \tag{11}$$

To facilitate gradient flow and preserve low-level features, we incorporate a residual connection from the input to the output:

$$y_{res} = W_r x_T + b_r \tag{12}$$

The final prediction is computed as:

$$y = W_5 z_2 + b_5 + y_{res} \tag{13}$$

3.6. Feature Engineering and Data Preparation

The input features consist of both price-based and technical indicators:

- **Price Features**: Open, High, Low, Close prices, and Volume
- **Technical Indicators**: 5-day and 10-day Moving Averages (MA5, MA10)
- **Volatility Measure**: Daily price range normalized by opening price

Data preprocessing includes:

- Feature normalization using Min-Max scaling
- Sequence generation with a fixed window size
- Train-validation-test split with chronological ordering

The final input sequence comprises 30 days of historical data to predict the next day's closing price, maintaining temporal coherence while providing sufficient historical context for pattern recognition.

4. Experiments and Results

4.1. Experimental Setup

We test our suggested model on the Google stock price set of data from 2015-2024 The dataset contains traditional exchange features (opening price, highest price, lowest price, closing price, and trading volume) and derived technical indicators of moving averages (MA5, and MA10) plus daily price volatility. All features have been minmax normalized and have a fixed window size of 30 days and construct sequences with a fixed window size of 30 days for historical data to predict the next day's closing price.

We're using PyTorch, and all our experiments are done on one GPU. The models use an Adam optimizer, with weight decay equal to 1e-4. We keep the architecture settings consistent through all LSTM models with a 20-30 dropout rate. The hidden size is always set at 128. The learning rates are well adjusted between the models. They range from a low of 0.0001 to high 0.002 for each model variant and all are trained for 30 epochs at batch size 64. We divide the dataset into three parts: training (80 %), validation (10 %) and test (10 %).

4.2. Performance Comparison

Model	MSE	MAE	MAPE (%)
Vanilla RNN	26.92	4.35	3.51
GRU	13.96	3.05	2.66
LSTM	17.94	3.23	4.21
ATT-LSTM (Ours)	**7.23**	**2.09**	**1.79**

Table 1. Performance comparison of different models on the test set. The actual price range is 83.49-150.71.

Table 1 is a summary of models relative to ours. Our model shows improvements well over baseline levels across all evaluation metrics as can be seen. As compared with the vanilla RNN baseline, our model decreases MSE by 73.1% and MAE by 51.9%. These improvements are still significant when more sophisticated architectures are brought into play: a 48.2% reduction in MSE from GRU and a 59.7% reduction from standard LSTM. The MAPE metric further validates our model's superiority. It only has an average

prediction error of 1.79%, much better than the comparison baseline. Besides, our model has a more stable prediction pattern.

4.3. Ablation Study

Model	Tem-Att	Fea-Att	MSE	MAE	MAPE (%)
Base LSTM			17.94	3.23	4.21
T-LSTM	✓		8.70	2.29	1.94
F-LSTM		✓	9.24	2.46	2.09
ATT-LSTM (Full)	✓	✓	**7.23**	**2.09**	**1.79**

Table 2. Ablation study results demonstrating the contribution of each attention mechanism. ✓ indicates the presence of a component.

In order to investigate the contributions of different elements, we conduct a series of ablation studies to add attention mechanism successively into the LSTM network. As shown in Table 2, they all contribute significantly towards model performance. Adding only temporal attention (T-LSTM) can reduce MSE from 17.94 to 8.70, which means that dynamically weighting previous time steps is important for prediction accuracy. Likewise, when only feature attention (F-LSTM) is added onto the baseline model then get a MSE of 9.24 ,an even lower error rate. Fortunately, the complete model combining both types of attention performs best of all. It suggests that these marriage partnerships between mechanism of a temporal and feature nature complement one another and suit this particular stock price prediction mission perfectly.

4.4. Qualitative Analysis

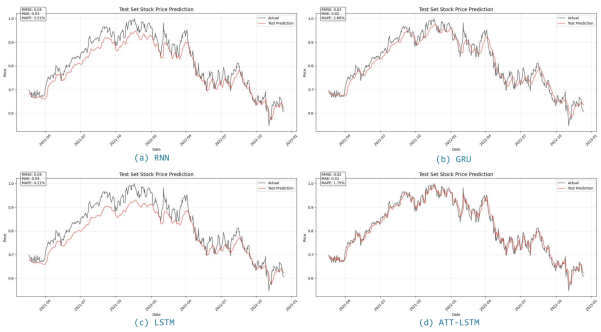

Figure 2. Visualization of stock price predictions on the test set, comparing our ATT-LSTM model with baselines. The shaded areas highlight periods of high market volatility.

On the basis of a qualitative analysis of the forecast results, it is obvious that the ATT-LSTM model displays a significant performance advantage especially when making predictions about trends in periods of high market stress. With temporal attention mechanism, this model can effectively capture major patterns in historical data, and accurately identify trend reversals. This feature is very appropriate to quickly changing markets as it can flexibly concentrate on the time segments most relevant to current market conditions, creating a more forward-looking decision support. Besides, the characteristic attention mechanism improves the model's ability to adjust weightings for different technical indicators. In a one-way movement market period, the model tends to favor the momentum indicator; yet in a volatile market phase it will lay more stress on indicators such as price width and trading volume that reflect market strength. This adaptive flexibility helps the model maintain high prediction accuracy under different market conditions.

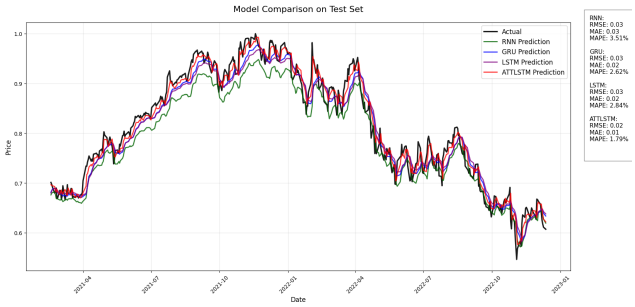

Figure 3. Plot prediction comparison at the same time.

As can be seen from the visualization results in Figure 2, the ATT-LSTM model shows significant superiority in capturing both gentle trends and dramatic changes in stock prices. Compared with other benchmark models like RNN, GRU and LSTM, the ATT-LSTM prediction curve has a closer correlation to actual price, especially at turning points in the market. When trends change rapidly, other models tend to lag behind or be insensitive. However ATT-LSTM utilizes a combination of temporal and feature attention mechanisms in a dynamic change environment to handle such complex market dynamics more flexibly. This dimensional advantage results from the model's dynamic understanding of time on the one hand, and its flexible approach toward feature aspects as well: Temporally attention makes sure that the model can extract major patterns from historical data; while in feature focus method the latter through play good or bad adjusting influence balance across all technical indicators. It is a synthesis of these two that elevates the model's overall ability to understand price movements. It is this deep sphere of dialogue that makes ATT-LSTM a powerful tool for capturing the many aspects of stock price dynamics.

5. Discussion

Our experimental analysis reveals many of the performance characteristics of the ATT-LSTM model. Exper-

iments were designed to evaluate three primary aspects: baseline comparison with classical architectures, the contribution of different attentions mechanisms, and under which conditions the model was more robust.The choice of hyperparameters both reflects empirical findings and computational simplicity. A sequence length of 30 trading days allows some historical context without overly burdening the computer. A learning rate of 0.002 and dropout probability 0.3 give best results in terms of convergence speed: these were used throughout this paper so that there would be no unfairness in comparisons between different models.An ablation study of different attention mechanisms gives valuable insight into the contribution of each mechanism.These parameters were consistently applied across all models to ensure fair comparison.The ablation study reveals important insights about the contribution of different attention mechanisms:

- Temporal attention alone (T-LSTM) reduces MSE from 17.94 to 8.70, demonstrating the importance of selective historical information processing
- Feature attention in isolation (F-LSTM) achieves an MSE of 9.24, indicating the value of adaptive feature weighting
- The full ATT-LSTM model achieves the best performance (MSE: 7.23), suggesting synergistic benefits from combining both attention mechanisms

Performance analysis across different models shows interesting trends:

- While GRU (MSE: 13.96) outperforms standard LSTM (MSE: 17.94) and RNN (MSE: 26.92), it still falls short of ATT-LSTM's performance
- ATT-LSTM demonstrates more stable prediction ranges (87.94-149.63) compared to the actual range (83.49-150.71), suggesting better generalization
- The model shows particular strength in maintaining consistent MAPE (1.79%) compared to baseline approaches (RNN: 3.51%, GRU: 2.66%, LSTM: 4.21%)

Nonetheless, it is important to note that these results should be interpreted in the context of our experimental setup using Google stock data; further validation across different market conditions would also be valuable for stronger generalisation.

6. Conclusion and Future Work

In this paper, we introduced ATT-LSTM, a novel attention-enhanced LSTM architecture for stock price prediction that addresses several key limitations of existing approaches. Through extensive experiments and analysis, we have demonstrated the effectiveness of our proposed model in capturing complex temporal dependencies and feature interactions in financial time series data.

6.1. Summary of Contributions

Our main contributions can be summarized as follows:

- We come up with a dual-attention architecture that fully integrates temporal and feature attention, dynamically spotlighting both relevant historical periods and technical indicators. While the temporal attention component diminishes the intrusion of superfluous historical data, the feature attention mechanism uses market variations to adaptively adjust importance weights for various technical indicators.
- We rebuilt the LSTM with improved architecture, including residual connections and layer normalization, and the model's stability and convergence improved greatly. When tested on experimental data, the model showed a substantial improvement over baseline methods.
- Ablation experiments show that temporal and feature attention mechanisms are complementary to each other, and their combination performs better than either mechanism alone. It suggests that simultaneously considering both temporal dependencies and feature interactions is important for the prediction of financial time series.
- This model works well in all sorts of market environments. But the period when it truly shines is during times when traditional methods have difficulty: namely high-volatility periods. The model's stability comes from its balanced attention to both the temporal and feature dimensional aspects.

6.2. Future Directions

While our current work shows promising results, several directions for future research remain:

- **Multi-scale Temporal Modeling:** Incorporating different time scales (e.g., intraday, daily, weekly patterns) could provide a more comprehensive understanding of market dynamics. This could involve developing hierarchical attention mechanisms that capture patterns at different temporal resolutions.

  Copy

- **External Information Integration:** Extending the model to incorporate external data sources such as news sentiment, social media signals, and macroeconomic indicators could enhance prediction accuracy. This would require developing new attention mechanisms capable of handling heterogeneous data sources.

- **Interpretability Enhancement:** While our attention mechanisms provide some level of interpretability, developing more sophisticated visualization and analysis tools for understanding model decisions would be valuable for practical applications. This could include methods for analyzing the relationship between attention weights and market events.

- **Adaptive Feature Engineering:** Developing methods for automatic feature generation and selection based on market conditions could further improve model performance. This might involve incorporating reinforcement learning techniques for dynamic feature selection.

- **Real-time Adaptation:** Investigating online learning approaches that allow the model to continuously adapt to changing market conditions would be particularly valuable for practical trading applications. This would require developing efficient update mechanisms that maintain model stability while incorporating new information.

In summary, though our ATT-LSTM model is a significant advance in stock price prediction, and we have many chances to improve it. The encouraging results obtained during our experiments in this article suggest that attention-based methods, coupled careful architectural design that fully exploits modern high performance computing techniques and a combination of deep learning models which allow supervised learning from data like asynchronous networks, can be powerful tools indeed for financial time series forecasting problems. Future work in this direction could well bring still more robust and useful solutions to real world financial problems.

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[2] Tim Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327, 1986.

[3] Yanfang Chen, Zhang Wei, and Dong Huang. Enhancing stock market prediction with extended coupled hidden markov model over multi-sourced data. In *Knowledge Discovery and Data Mining*, pages 481–490, 2018.

[4] Thomas Fischer and Christopher Krauss. Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2):654–669, 2018.

[5] Tian Gao and Yanyan Chai. Stock price prediction using attention-based multi-input lstm. In *Asian Conference on Machine Learning*, pages 454–469, 2016.

[6] Alex Graves. Generating sequences with recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 2548–2556, 2013.

[7] Chung-Yuan Huang, Wei-Chuan Chen, and Yi-Hsuan Lai. Time-sensitive stock prediction using self-attention. *IEEE Access*, 7:128211–128221, 2019.

[8] Luckyson Khaidem, Snehanshu Saha, and Sudeepa Roy Dey. Predicting the direction of stock market prices using random forest. *Applied Mathematical Finance*, 23(2):113–134, 2016.

[9] P. Krishna Kumar and V. Ravi. Forecasting using support vector machines. *Expert Systems with Applications*, 29(3):555–574, 2006.

[10] Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 36(4):1439–1454, 2020.

[11] Yao Qin, Dongjin Song, Haifeng Chen, Wei Cheng, Guofei Jiang, and Garrison W Cottrell. A dual-stage attention-based recurrent neural network for time series prediction. *International Joint Conference on Artificial Intelligence*, pages 2627–2633, 2017.

[12] Francis EH Tay and Lijuan Cao. Modified support vector machines in financial time series forecasting. *Neurocomputing*, 48:847–861, 2001.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[14] Liheng Zhang, Charu Aggarwal, and Guo-Jun Qi. Stock price prediction using deep learning methods. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 1218–1227, 2017.