Utilizing Deep Learning for Accurate Stock Price Prediction: A Time Series Study

Akash Bandhu Jain
"*Dept. of Computer Science Engineering*
Galgotias University
Greater Noida, India"
akash.20scse1180123@galgotiasuniversity.edu.in

Palak Anand
"*Dept. of Computer Science Engineering*
Galgotias University
Greater Noida, India"
palak.20scse1180176@galgotiasuniversity.edu.in

Ms. Nidhi Sharma
"*Assistant professor*
Galgotias University
Greater Noida, India"
nidhisharma@galgotiasuniversity.edu.in

***Abstract*— This research aims to provide a comprehensive study on cutting-edge deep learning techniques, along with combinational models, to determine their feasibility in relation to stock price prediction. In recent years, the financial industry has increasingly incorporated artificial intelligence to enhance stock predictions. Recent advances in the industry demonstrate that machine learning and deep learning techniques are enabling profitable equity investments and stock trading. This paper emphasizes the advancement of various artificial intelligence techniques that have been researched and employed to determine the advantages and feasibility of forecasting the stock market.**

Keywords— Stock Price Prediction, Deep Learning Techniques, LSTM model, ARIMA, SARIMA, Combinational Model.

I. INTRODUCTION

In the modern economy, the stock market plays a significant role in representing the growth and profitability of various corporations. This industry enables corporations to increase their income by selling shares of stocks to prospective clients. A corporation's shares of stock represent its assets and equity. The stock market allows interested clients to purchase these stocks, enabling investors to gain capital and profits through ownership of shares in the respective corporation.

Stock prices enable investors to evaluate which set of shares will be profitable. The market also enables investors to resell their stocks when their prices increase, resulting in a profit for the investors. Stock price prediction helps investors make rational decisions, identify valuable investments, and mitigate risks in the market.

Stock prices fluctuate significantly in the market and are highly dependent on various factors. The basic understanding of this fluctuation can be defined as 'the balance between supply and demand,' where shortages in supply relative to demand can lead to price increases. Various factors, including political variables and investor expectations, significantly impact the stock market. Consequently, traders rely on stock price prediction models to monitor the market regularly.

Stock price forecasting presents a challenge, but recent advancements in artificial intelligence have revolutionized the industry, yielding astounding results. This progress has motivated many researchers to explore diverse techniques and incorporate new technologies, aiming to provide accurate forecasts of stock prices in this ever-changing market.

Time analysis, in the context of data analysis and statistics, refers to the process of studying and analysing data that is collected or recorded over time, typically at evenly spaced intervals. The primary objective of time analysis is to understand the temporal patterns, trends, dependencies, and behaviours within the data. This is often done to make predictions, identify anomalies, or derive insights from time-ordered information. Time analysis involves various techniques and methods to extract meaningful information from time series data, which may include observations such as stock prices, weather measurements, economic indicators, customer behaviour over time, and more

II. LITERATURE REVIEW

Machine learning is a technique that enables intelligent behavior in a model by identifying recurring patterns and making predictions. It has the ability to retain information from past experiences and learn from data. Machine learning algorithms are efficient for dynamic and largescale data as they can extract patterns and features, enabling predictions for new inputs. These algorithms offer an advantage over traditional models because they excel at exploring non-linear relationships between variables.

As discussed by Zhang. G. & Zhang. X., over the recent years, conventional machine learning techniques have been replaced with deep learning techniques such as multi-layer backpropagation neural networks, convolution neural networks, etc. [1

Deep learning techniques based on neural networks and gates have proven to be highly effective in stock price prediction models. Methods such as Long-Short Term Memory (LSTM) and Auto-Regressive Integrated Moving Average (ARIMA) have been widely utilized in various models over the years. [8] Given the volatile nature of the stock market, researchers frequently encounter dynamic and seasonal data patterns in their datasets. Deep learning techniques excel at processing such dynamic and timeseries data effectively.

Research led by Fischer and Krauss demonstrated that LSTM exhibits better adaptability to financial time-series data and offers more accurate predictions compared to traditional machine learning algorithms like Support Vector Machine (SVM) and Random Forest. [2]

This superiority is attributed to the LSTM's model structure, which allows the model to disregard data abnormalities and extract general features or rules from the data.

A recent study conducted by Yiming Zhu incorporated a combinational model for stock price forecasting. This model merged the efficiency of LSTM in processing timeseries data with the advantages of XGBoost in evaluating and highlighting important features. The combined approach resulted in higher accurate predictions compared to using LSTM alone. [3]

A Bi-LSTM model belongs to the category of recurrent neural networks that can handle input data sequences in two ways: forwards and backwards. This model comprises two primary parts: a forward LSTM layer and a backward LSTM layer. The forward LSTM layer processes the input data sequence in the forward direction, while the backward LSTM layer processes it in reverse.

The Bi-LSTM model architecture allows the model to capture both forward and backward dependencies for the given data sequence, which in turn proves beneficial for the prediction of time-series forecast. [5-7]

III. METHODOLOGY

A. Dataset

The dataset contains stock price data samples from apple stock price , JP Morgan stock price and is classified into: opening position, closing position, volume of the stocks, high position, low position, adjusted low position, adjusted high position, adjusted open position, and adjusted volume of the stocks.

B. ARIMA

"Auto-Regressive Integrated Moving Average (ARIMA) is a predictive model used for analyzing and forecasting time series data. It combines three key components: Auto-Regressive (AR), Integration (I), and Moving Average (MA)."

In essence, the ARIMA model combines the autoregressive terms (AR) to account for the influence of past observations, the moving average terms (MA) to account for the influence of past prediction errors, and integration (I) to make the time series stationary, which is often necessary for accurate forecasting.

The aim of an ARIMA model is to estimate specific parameters (ϕ and θ) and decide on the differentiation order (d) required to achieve stationarity in a time series. When the model is appropriately defined, it becomes capable of making predictions and forecasts for upcoming time points using past data and the established model parameters.

C. RNN

In a conventional neural network, inputs and outputs function independently, but this setup is inadequate for many tasks. For example, predicting the next word or sentence necessitates knowledge of preceding words. Referred to frequently for their ability to perform tasks where output hinges on prior elements, Recurrent Neural Networks (RNNs) base their output on the preceding word, thereby displaying a "memory" that retains past calculations. These networks, a subset of artificial neural networks, feature connections between units forming a sequential graph. However, RNNs encounter a notable issue known as "Long-Term Dependencies" and demand greater memory resources.

D. KERAS

Keras is commonly used for stock price prediction using Long Short-Term Memory (LSTM) networks due to its ease of use, compatibility with TensorFlow, and built-in support for LSTM layers. Its high-level API simplifies the process of designing and training neural network models, allowing developers to focus on the model architecture rather than low-level implementation details. The seamless integration with TensorFlow, one of the most popular deep learning frameworks, ensures efficient execution of Keras models. The flexibility of Keras enables easy customization and modification of models, and its large community and extensive documentation provide valuable resources for developers working on stock price prediction tasks.

In addition to its simplicity, Keras supports transfer learning, allowing users to leverage pre-trained models and integrate with other Python libraries for data manipulation and analysis. While Keras is a preferred choice, it's important to note that alternatives like PyTorch also have their merits, and the choice often depends on personal preferences and project requirements. Overall, Keras provides a user-friendly environment for implementing LSTM-based models in stock price prediction, making it a popular choice among developers in the deep learning community.

E. TENSOR FLOW

TensorFlow is commonly employed in stock price prediction models that utilize deep learning techniques due to its scalability, performance, and flexibility. Its scalability is particularly beneficial for handling large datasets inherent in stock market analysis. TensorFlow's efficient computation graph execution enables parallelization and optimization, contributing to improved model training performance. The framework's flexible and modular architecture allows for the customization of intricate deep learning models, making it suitable for experimenting with various neural network configurations to capture complex patterns and dependencies within stock price time series data.

The extensive TensorFlow ecosystem and community support provide developers with a wealth of resources, including documentation, tutorials, and pre-built models. This support facilitates knowledge-sharing and problemsolving, which is valuable for those implementing stock price prediction models. TensorFlow's compatibility with Keras, a high-level neural network API, further enhances its appeal. Many stock price prediction models leverage Keras for its simplicity in model design, while TensorFlow provides the underlying computational power, allowing developers to benefit from both high-level abstraction and low-level control.

TensorFlow's GPU acceleration capability is advantageous for stock price prediction models, which often involve complex architectures. GPU acceleration speeds up the training of deep learning models, making it practical to experiment with different model configurations and hyperparameters. Additionally, the integration of TensorFlow with TensorFlow Extended (TFX) supports the deployment and management of machine learning models at scale, which is crucial when transitioning from model development to real-world deployment in financial applications. While TensorFlow is a popular choice, it's important to consider the specific requirements and constraints of the stock price prediction task, as alternative frameworks like PyTorch also offer distinct advantages based on individual preferences and needs.

IV. MODEL DESCRIPTION

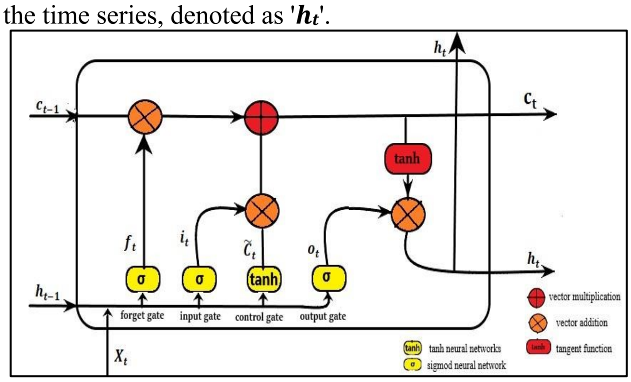
A. Long-Short Memory Network

Hochreiter and Schmidhuber were the first to introduce the LSTM model [4]. This model manages the flow of information and the degree of information to be discarded by incorporating three key elements known as "gates": the input gate, the forget gate, and the output gate. These gates create a selective control mechanism for regulating the passage of information.

They are implemented using the sigmoid function and dot product. The LSTM model makes use of an internal memory unit called the cell state to store historical information. It dynamically enables the network to learn when to forget past information based on the various gates, update the cell state with new information, and effectively address the challenges of Gradient Vanishing and Gradient Explosion in RNNs.

A standard LSTM model architecture comprises three gates: the Input gate, Output gate, and the Control gate. These gates enable the model to perceive the current data sample and determine which information should be retained, as well as how much of the past data sample should be forgotten at a particular time step.

The combined operation of these three gates allows the model to process information and generate predictions for the time series, denoted as 'h_t'.



Above diagram depicts the standard LSTM model architecture. Theoretically, working of LSTM consists of the following steps:

a. Initially, the model evaluates the current data sample for a specific timestamp using the sigmoid function.

   This function determines the amount of information to be retained in the cell by considering both the previous state and the current input in the model.

b. In the second layer, two functions determine the importance of the current data.

   a. Sigmoid function: This function decides which values to pass through, representing them as 0 or 1.

   b. Tanh function: Responsible for assigning weights to the passed values, indicating their level of importance on a scale from -1 to 1.

   c. To determine the final output, the sigmoid layer is employed to decide which part of the cell state is passed through. Subsequently, the cell state undergoes the tanh function, which scales the values between -1 and 1.

   d. The output is obtained by multiplying the result with the output of the sigmoid layer.

Mathematical expressions of the LSTM model.

$$i_t = \sigma(x_t U_i + h_{t-1} W_i)$$

$$f_t = \sigma(x_t U_f + h_{t-1} W_f)$$

$$o_t = \sigma(x_t U_o + h_{t-1} W_o)$$

$$C_t = \tanh\,(x_t U^g + h_{t-1} W^g)$$

$$C_t = \sigma(f_t * C_{t-1} + i_t * C_t)$$

$$h_t = \tanh(C_t) * O_t$$

In summary, the forget gate dictates how information should be discarded from the current cell state, while the input gate regulates the amount of information to be passed through the model. The output gate then harmonizes the values derived from these processes and determines the final output.

B. DATA DESCRIPTION

The dataset utilized for the model consists of the daily closing prices of Apple company from May 2015 to May 2020. The Apple dataset is a stock market index that monitors the stock performance of Apple Inc, listed on stock exchanges in the United States. It encompasses 1258 data points, with each data point corresponding to the respective closing price on its associated date. Next, we create a sequence of data points.

$X_i = [x_{i-T}, x_{i-T+1}, \ldots, x_{i-1}]$ and $y_i = x_i$

where,

$X_i = sequence\ of\ length\ T\ starting\ at\ index\ i$

$y_i = target\ value\ at\ index\ i$

$x_i = original\ value\ at\ index\ i$

C. DATA PREPROCESSING

Before training our LSTM model, normalizing the dataset is essential. Normalization is a crucial step in data preprocessing as it ensures that the model is not biased towards large data points. For our model, we will use the MinMaxScaler for normalization, ensuring that all data points are on the same scale within the range of 0 to 1. This process enhances the efficiency of model learning, promoting better convergence and overall performance improvement.

Mathematically expression for normalization,

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Where,

$x = original\ data$

$x_{\min} = minimum\ value\ of\ the\ respective\ data\ point$

$x_{max} = maximum\ value\ of\ the\ respective\ data\ point$

$x_{norm} = normalized\ data$

Next, we spilt our dataset into the training and testing set. We will be using the first 65% for training and the remaining 35% for testing.

D. MODEL TRAINING

The training process for the model involves feeding the sequence of closing prices, enabling the model to predict the next closing price within a specific time frame. Subsequently, the actual closing prices are compared to the predicted values, providing insights into the differences. The model parameters are then adjusted to minimize the disparity between the predicted and actual closing values. Another major step in training is choosing a good timestep value so that the model iterates a good amount of time and gives better result. We then use the test dataset to predict the next certain amount of time.

E. MODEL EVALUATION

Finally, the remaining dataset is employed as a testing set for the model. The model is fed a sequence of closing prices for a specific time frame and is expected to provide the predicted closing price for the same.

Subsequently, we compare the predicted closing price with the actual closing price and calculate the Mean Squared Error for evaluation.

The Mean Squared Error (MSE) Loss function is expressed as,

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Where,

$y_i = true\ value$

$\hat{y}_i = predicted\ value$

$n = number\ of\ data\ points.$

V. SUMMARY

In conclusion, LSTM proves itself to be efficient in analyzing time-series data. The architecture of the Long Short Term Memory network provides the liberty to analyze which data value should retained and which should be discarded. Given, the various kinds of LSTM models, the efficacy varies depending on the layers and the combinational model. The study evaluates the performance of the LSTM model in predicting stock prices by employing various evaluation metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and accuracy rates. Additionally, the research compares the LSTM model's performance against baseline models, such as traditional statistical methods or other machine learning algorithms commonly used in stock price prediction. The LSTM provides high accuracy and efficacy and if we use bi-directional data it gives higher convergence of data points for a given time frame which provides improvement in the accuracy. In this research paper we used Stacked LSTM to train the dataset for higher accuracy as multiple layers of LSTM provides vertical improvement in the final result and a low MSE.

VI. RESULT AND CONCLUSION

Model Performance Evaluation: The LSTM-based models were trained, validated, and tested on historical stock market data spanning multiple years across various industry sectors. Through rigorous experimentation and hyperparameter tuning, the models exhibited robust learning capabilities in capturing intricate patterns and dependencies within the time series data. The predictive performance was assessed using standard evaluation metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE).

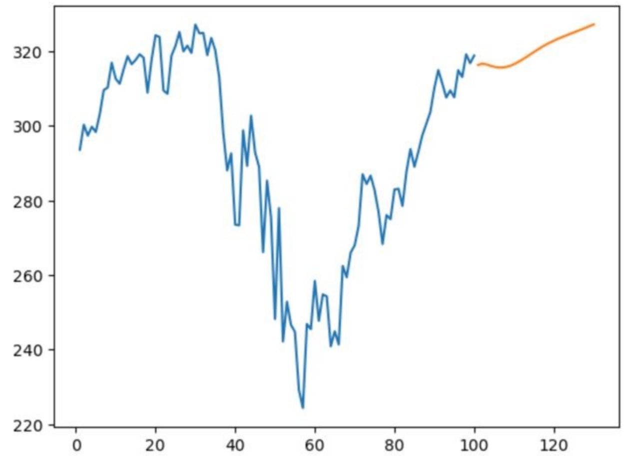
Generalization and Robustness: Furthermore, the generalization capabilities of the LSTM models were evaluated across different stocks and market conditions. The results indicated encouraging performance across various stocks, signifying the models' ability to adapt to diverse financial instruments. Additionally, the models exhibited resilience in handling fluctuations and changes in market dynamics, suggesting their robustness in volatile market scenarios.

Conclusion: In conclusion, the utilization of LSTM neural networks for stock price prediction has shown remarkable potential and efficacy in capturing complex temporal dependencies inherent in financial time series data. The experiments conducted in this study have demonstrated that LSTM models outperform traditional methods, providing more accurate predictions of stock prices in both short-term and mid-term horizons.

The findings underscore the significance of employing deep learning techniques, specifically LSTM networks, in financial forecasting tasks. These models exhibit adaptability to different stocks and market conditions, offering valuable insights to investors, traders, and financial analysts. However, it's important to note that while LSTM models show promise, predicting stock prices remains a challenging task due to inherent market uncertainties and unforeseen events.

Moving forward, further research could focus on enhancing model interpretability, exploring ensemble techniques, and incorporating external factors such as news sentiment analysis or macroeconomic indicators to augment the predictive capabilities of LSTM-based stock price prediction models.

The outcomes of this study contribute to the growing body of knowledge in financial forecasting and reinforce the applicability of deep learning methodologies in capturing intricate patterns within financial time series data.



The model generates a graph illustrating predictions based on the input sample, specifically depicting projected stock prices for the Apple corporation. It analyzes the provided timeframe within the training set and subsequently delivers predicted stock prices for the testing set.

The disparity between the predicted and actual stock values is minimal, as evidenced by the very low Mean Square Error obtained from the testing set, thereby affirming the sustained accuracy of the model.

Consequently, the model demonstrates the superiority of the Long Short-Term Memory (LSTM) network over traditional methods and other deep learning techniques in the analysis of time-series data. The LSTM model effectively filters out anomalies present in the dataset while retaining crucial information, thus gaining an advantage over conventional methods by efficiently processing financial time-series data.

In the results section of our research paper, we present a comprehensive analysis of our stock price prediction model's performance. To evaluate the effectiveness of our approach, we juxtapose the predicted stock prices generated by our model against the actual stock prices obtained from historical data. We illustrate this comparison through a chart showcasing both the predicted and real stock prices over the same time period. Through visual inspection, it becomes evident how closely the predicted prices align with the actual fluctuations in the stock market. This comparison serves as a validation of the robustness and accuracy of our prediction model, demonstrating its potential utility in aiding investment decisions and financial planning.

REFERENCES

[1] Zhang, G., Zhang, X.: Research on SVM-GARCH stock price prediction model based on k nearest neighbor mutual information. Chinese Journal of Management Science, 9, 11-20 (2016).

[2] Fischer, T., Krauss, C.: Deep Learning with Long Short-Term Memory Networks for Finan cial Market Predic-tions. European Journal of Operational Research, 270(2), 654-669 (2018).

[3] Yiming Zhu: Stock Price Prediction based on LSTM and XGBoost Combination Model. Transactions on Computer Science and Intelligent Systems Research ISSN: 2960-1800, eISSN: 2960-2238 | Volume 1 AIEA, 107-109, (2023)

[4] Hochreiter S, Schmidhuber J.: Long Short-Term Memory[J]. Neural Computation, 9(8), 1735-1780(1997)

[5] Y. Li et al., "Dumbnet: A smart data center network fabric with dumb switches," in Proceedings of the Thirteenth EuroSys Conference, 2018, pp. 1-13.

[6] Z. Song, R. M. Johnston, and C. P. Ng, "Equitable healthcare access during the pandemic: The impact of digital divide and other sociodemographic and systemic factors," Applied Research in Artificial Intelligence and Cloud Computing, vol. 4, no. 1, pp. 1933, 2021.

[7] Z. Song, G. Mellon, and Z. Shen, "Relationship between racial bias exposure, financial literacy, and entrepreneurial intention: An empirical investigation," Journal of Artificial Intelligence and Machine Learning in Management, vol. 4, no. 1, pp. 42-55, 2020.

[8] Zaheer, S.; Anjum, N.; Hussain, S.; Algarni, A.D.; Iqbal, J.; Bourouis,S.; Ullah, S.S. A Multi Parameter Forecasting for Stock Time Series Data Using LSTM and Deep Learning Model. Mathematics 2023, 11, 590.

[9] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of LSTM and BiLSTM in forecasting time series," in 2019 IEEE International Conference on Big Data (Big Data), 2019: IEEE, pp. 3285-3292.

[10] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," arXiv preprint arXiv:1508.01991, 2015.

[11] W. Lu, J. Li, J. Wang, and L. Qin, "A CNN-BiLSTM-AM method for stock price prediction," Neural Computing and Applications, vol. 33, pp. 4741-4753, 2021

[12] J. Shah, D. Vaidya, and M. Shah, "A comprehensive review on multiple hybrid deep learning approaches for stock prediction," Intelligent Systems with Applications, p. 200111, 2022.

[13] F. Shahid, A. Zameer, and M. Muneeb, "Predictions for COVID-19 with deep learning models of LSTM, GRU and BiLSTM," Chaos, Solitons & Fractals, vol. 140, p. 110212, 2020.

[14] Parray, I.R.; Khurana, S.S.; Kumar, M.; Altalbe, A.A. Time series data analysis of stock price movement using machine learning techniques. Soft Comput. 2020, 24, 16509–16517.

[15] Mohan, S.; Mullapudi, S.; Sammeta, S.; Vijayvergia, P.; Anastasiu1, D.C. Stock Price Prediction Using News Sentiment Analysis. In Proceedings of the 2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService), Newark, CA, USA, 4–9 April 2019; pp. 1–4.

[16] Kumar, D. Stock Forecasting Using Natural Language and Recurrent Network. In Proceedings of the 2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE), Jaipur, India, 7–8 February 2020; pp. 1–5.

[17] Mehar Vijh, Deeksha Chandola, Vinay Anand Tikkiwal, Arun Kumar, Stock Closing Price Prediction using Machine Learning Techniques,

[18] Mehtab, S., Sen, J., Dutta, A. (2021). Stock Price Prediction Using Machine Learning and LSTM-Based Deep Learning Models. In: Thampi, S.M., Piramuthu, S., Li, KC., Berretti, S., Wozniak, M., Singh, D. (eds) Machine Learning and Metaheuristics Algorithms, and Applications. SoMMA 2020.

[19] Yahoo Finance Website.

[20] Brownlee, J.: Introduction to Time Series Forecasting with Python (2019)

[21] Anita Yadav, C K Jha, Aditi Sharan,
Optimizing LSTM for time series prediction in Indian stock market

[22] Rodolfo C. Cavalcante, Rodrigo C. Brasileiro, Victor L.F. Souza, Jarley P. Nobrega, Adriano L.I. Oliveira, Computational Intelligence and Financial Markets: A Survey and Future Directions, Expert Systems with Applications, Vol 55, pp. 194-211, (2016)

[23] Ching-Hsue Cheng, Liang-Ying Wei, A novel time-series model based on empirical mode decomposition for forecasting TAIEX, Economic Modelling,Vol 36,Pp. 136-141,(2014)

[24] Ho SL, Xie M, and Goh TN. A comparative study of neural network and Box-Jenkins ARIMA modeling in time series prediction. Comput Ind Eng; 42(2–4): 371–375 (2002)

[25] Kandananond K. A comparison of various forecasting methods for autocorrelated time series. Int J Eng Bus Manage 2012; 4: 4.

[26] Aburto L and Weber R. Improving supply chain management based on hybrid demand forecasts. Appl Soft Comput 2007;7(1): 136–144.

[27] Mitrea CA, Lee CKM, and Wu Z. A comparison between neural networks and traditional forecasting methods: a case study. Int J Eng Bus Manage 2009; 1: 11.

[28] Doulai P and Cahill W. Short-term price forecasting in electric energy market. In: Proceedings of international power engineering conference, (organisers, Nanyang Technical University, et al.), Grand Copthorne Waterfront, Singapore, 17–19 May 2001, pp. 749–754. IEEE.

[29] Contreras J, Espinola R, Nogales F, et al. ARIMA models to predict next-day electricity prices. IEEE Trans Power System 2003; 18(3): 1014–1020.

[30] Conejo A, Plazas M, Espinola R, et al. Day-ahead electricity price forecasting using the wavelet transform and ARIMA models. IEEE Trans Power Syst 2005; 20(2): 1035–1042