



Reinforcement Learning in Stock Trading

Quang-Vinh Dang

► To cite this version:

| Quang-Vinh Dang. Reinforcement Learning in Stock Trading. 2019. hal-02306522

HAL Id: hal-02306522

<https://hal.archives-ouvertes.fr/hal-02306522>

Preprint submitted on 6 Oct 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Reinforcement Learning in Stock Trading

Quang-Vinh Dang^[0000-0002-3877-8024]

Industrial University of Ho Chi Minh city, Vietnam
dangquangvinh@iuh.edu.vn

Abstract. Using machine learning techniques in financial markets, particularly in stock trading, attracts a lot of attention from both academia and practitioners in recent years. Researchers have studied different supervised and unsupervised learning techniques to either predict stock price movement or make decisions in the market.

In this paper we study the usage of reinforcement learning techniques in stock trading. We evaluate the approach on real-world stock dataset. We compare the deep reinforcement learning approach with state-of-the-art supervised deep learning prediction in real-world data. Given the nature of the market where the true parameters will never be revealed, we believe that the reinforcement learning has a lot of potential in decision-making for stock trading.

Keywords: Reinforcement Learning · Machine learning · Stock Trading.

1 Introduction

Searching for an effective model to predict the prices of the financial markets is an active research topic today [13] despite the fact that many research studies have been published for a long time [3, 11]. In the midst of financial markets prediction, stock price prediction is considered as one of the most difficult tasks [44]. Among the state-of-the-art techniques, machine learning techniques are the most widely chosen techniques in recent years, given the rapid development of the machine learning community. The other reason is that the traditional statistical learning algorithms can not cope with the non-stationary and non-linearity of the stock markets [15].

In general, there exists two main approaches to analyze and predict stock price which are technical analysis [23] and fundamental analysis [39]. The technical analysis looks into the past data of the market only to predict the future. On the other hand, the fundamental analysis takes into account other information such the economic status, news, financial reports, meeting notes of the discussion between CEOs, etc.

The technical analysis relies on the efficient market hypothesis (EMH) [25]. The EMH states that all the fluctuation in the market will be reflected very quickly in the price of stocks. In practice, the price can be updated in the magnitude of milliseconds [8], leading to a very high volatility of the stocks. In recent years the technical analysis attracts a lot of attention due to a simple fact that we

have enough information just by looking to the historical stock market, which is public and well-organized, compared to the fundamental analysis where we need to analyze unstructured dataset.

Compared to the supervised learning techniques and at a certain level, unsupervised learning algorithms, are widely used in stock price prediction, to the best of our knowledge the reinforcement learning for stock price prediction has not yet received enough support as it should be. The main issue of supervised learning algorithms is that they are not adequate to deal with time-delayed reward [22, 18]. In other words, supervised learning algorithms focus only on the accuracy of the prediction *at the moment* without considering the delayed penalty or reward. Furthermore, most supervised machine learning algorithms can only provide *action recommendation* on particular stocks¹, using reinforcement learning can lead us directly to the *decision making* step, i.e. to decide how to buy, hold or sell any stock.

In the present paper we study the usage of reinforcement learning in stock trading. We review some related works in Section 2. We present our approach in Section 4. We describe and discuss the experimental results in Section 5. We conclude the paper and draw some future research directions in Section 6.

2 Literature Review

There are two main applications of using machine learning in the stock markets: stock price prediction and stock trading.

Stock price prediction can be divided into two applications: price regression or stock trend prediction. In the first application, the researchers aim to predict exactly the numerical price, usually based on day-wise price [15] or closed price of a stock. In the second approach, the researchers usually aim to predict the turning point of a stock price, i.e. when the stock price change the moving direction from up to down or vice versa [44]. Traditionally time-series forecasting techniques such as ARIMA and its variant [43, 26] are adapted from the econometric literature. However, these methods cannot cope with non-stationary and non-linearity nature of the stock market [2].

It is claimed that the stock price reflects the belief or opinions of the market on the stock rather than the value of the stock itself [7]. Several research studies propose to analyzing the social opinions to predict the stock price. In the research study of [33], the authors used Google Trends, i.e. to analyze the Google query volumes for a particular keyword or search term, they can measure the attention level of the public to a stock. The research is based on one idea that a decision making process will start by information collection [37].

Over centuries, the researchers and practitioners have developed many technical indicators to predict the stock price [29]. Technical indicators are defined

¹ According to NASDAQ standard, recommendation from analysts can be Strong Buy, Buy, Hold, Underperform or Sell. Reference: <https://www.nasdaq.com/quotes/analyst-recommendations.aspx> accessed on 07-September-2019.

as a set of tool that allow us to predict the future stock market by solely looking to the historical market data [31]. Originally the technical analysis are not highly supported in academia [27] even though it is very common in practice [35]. Nevertheless, with the development of the machine learning community, technical analysis gains attention of researchers in recent years. [32] derived what the authors called “Trend Deterministic Data Preparation” from ten technical indicators then combined them with several machine learning techniques such as Support Vector Machine (SVM) or Random-Forest for the stock price movement prediction. The “Trend Deterministic Data Preparation” are simply the indication from the technical indicators that the price will go up or down, so the approach of [32] can be considered as ensemble learning from local experts [17].

The authors of [44] inherited the idea of Japanese candle stick in stock analysis [30] to develop a status box method combined with probabilistic SVM to predict the stock movement. We visualized a Japanese candlestick in Figure 1. The status box developed by [44] is presented in Figure 2. The main idea is instead of focusing on only one time period as the traditional Japanese candlestick method, the status box focus on a wider range of time, that allow us to overcome the small fluctuation in the price.

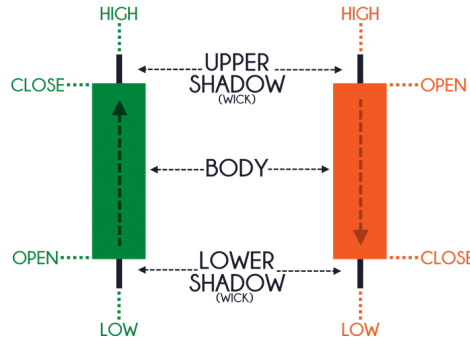


Fig. 1. Japanese candle stick using in stock analysis. A candlestick shows us the highest and lowest price of a stock in a period of time, as well as opening and closing price of this stock.

It is quite straightforward to combine technical analysis and fundamental analysis into a single predictive model. The authors of [42] combined news analysis with technical analysis for stock movement prediction and claimed that the combination yields a better predictive performance than any single source.

Deep learning, both supervised and unsupervised techniques, have been utilized for stock market prediction. One of very first research work in this segment belongs to the work of [40] published in 1996 to use recurrent neural networks (RNN) in ARIMA-based features. Many other feature extraction methods based on supervised or unsupervised learning have been developed since then [21].

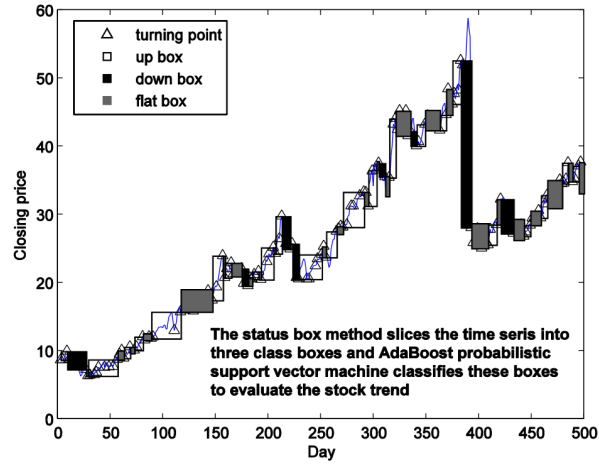


Fig. 2. A status box based on [44].

The authors of [4] used three different unsupervised feature extraction methods: principal component analysis (PCA), auto-encoder and restricted Boltzmann machine (RBM) for the auto-regressive (AR) model. In the same direction, the authors of [24] designed a multi-filters neural network to automatically extract features from stock price time-series. The authors combine both convolutional and recurrent filters to one network for the feature extraction task. In [45] the authors used Empirical Mode Decomposition [34] with neural networks for the feature extraction.

3 Our Contribution

Compared to supervised/unsupervised learning approaches, the difference of our approach is that we generate a trading strategy rather than only stock price prediction as in existing research studies [10, 19]. Stock price prediction definitely is a very important task, but eventually we need to build a strategy to decide what to buy and what to sell in the market that requires a further research step. In this study, we employ a simple baseline greedy strategy given the prediction of a supervised learning algorithm (RNN-LSTM) but definitely studying a strategy based on the prediction of another algorithm is not a trivial task.

Several research works in using reinforcement learning have been presented [22, 6, 1] in literature. However, the work of [22] is presented in 2001 using only TD(0) algorithm, while the works of [6] or [1] used external information such as news for the trading task. In our approach we do not use any external information but only the historical data of stock prices for the trading.

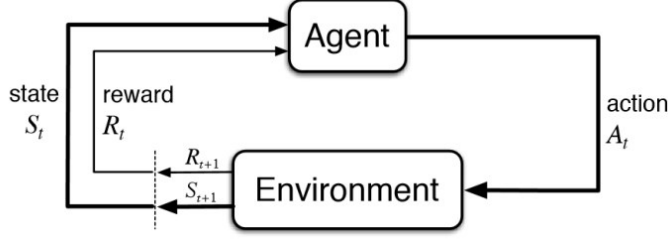


Fig. 3. The interaction between agent and environment in reinforcement learning.

4 Reinforcement Learning

Reinforcement learning [38] is visualized in Figure 3. Different from supervised learning techniques that can learn the entire dataset in one scan, the reinforcement learning agent learns by interacting repeatedly with the environment. We can imagine the agent as a stock trader and the environment as the stock market [22]. At a time step t , the agent performs an action A_t and receives a reward $R_{t+1} = R(S_t, A_t)$. The environment then move to the new state $S_{t+1} = \delta(S_t, A_t)$. The agent needs to learn a policy $\pi : S \rightarrow A$, i.e. learn to react to the environment in which it can maximize the total reward as:

$$V^\pi(S_t) = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \quad (1)$$

Here, the coefficient γ represents the decay factor, usually consider as interest rate in finance, reflects the “one dollar today is better than one dollar tomorrow” statement. It means any trading strategy should beat the risk-free interest rate because otherwise a reasonable investor should not invest to this strategy at all - she should invest money to the risk-free rate such as buying T-bonds or opening a saving account. The latter option will give her a better profit and lower risk. However in high-frequency trading and short period of time we can set γ close to 1. The optimal policy is notated as π^* .

In this paper we employ Deep Q-learning [28] by approximate the optimal policy function by a deep neural network. The term “Deep” here refers to Deep Convolutional Neural Networks (CNNs) [36]. Here we parameterize the Q-function by a parameter set θ .

In our settings, the actions are similar to other stock trading studies. The possible actions include buy, hold or sell. We defined the rewards as the profit (positive, neutral or negative) after each action.

The loss function is:

$$L(\theta) = \frac{1}{N} \sum_{i \in N} (Q_\theta(S_i, A_i) - Q'_\theta(S_i, A_i))^2 \quad (2)$$

with

$$Q'_\theta = R(S_t, A_t) + \gamma \max_{A'_i}(S'_i, A'_i) \quad (3)$$

The network is updated by a normal gradient descent:

$$\theta \leftarrow \theta - \alpha \frac{\partial L}{\partial \theta} \quad (4)$$

In deep Q-network, the gradient of the loss function is calculated as:

$$\nabla_{\theta_i} L(\theta_i) = E_{S, A \sim P(\cdot), S' \sim \epsilon} [(R_{t+1} + \gamma \max_{A'} Q(S_{t+1}, A') - Q(S, A, \theta_i)) \nabla_{\theta_i} Q(S, A, \theta_i)] \quad (5)$$

5 Experiments

5.1 Datasets

We use the daily stock price of more than 7,000 US-based stocks collected up to 10-November-2017². For each stock, we always use the period of time from 01-January-2015 until 10-November-2017 for testing, and the data from 01-January-2015 until 31-December-2016 as the training set. Hence, there are 504 samples for training and 218 samples for testing. The sample size is so small compared to well-known supervised learning problem such as ImageNet [20] that contains one million labelled images, but as we will present in Section 5.2, we still can generate positive profit strategies.

The stock price of Google is displayed in Figure 4.



Fig. 4. The stock price of Google from 01-Jan-2015 to 10-November-2017

5.2 Experimental Results

We evaluate three variants of Deep Q-learning which is vanilla Deep Q-learning [28], double DQN [12] and dueling double DQN [41].

We used RNN-LSTM model with a greedy strategy approach as the baseline model. According to a recent time-series prediction³ organized in July 2019, the RNN-LSTM model achieved the highest performance in forecasting a financial time-series. The greedy strategy means that we buy every-time we predict the stock will go up and sell if we predict the stock will go down.



Fig. 5. Performance of vanilla DQN on Google stock. The profit on the test period is -838.



Fig. 6. Performance of Double DQN on Google stock. The profit on the test period is 1430.



Fig. 7. Performance of Dueling Double DQN on Google stock. The profit on the test period is 141.

We visualize the performance of four models vanilla DQN, double DQN, dueling DQN and the baseline LSTM model on the Google stock (code: GOOGL) in Figure 5, Figure 6 and Figure 7 respectively. The profit of each model are described in the figure. In each figure, we visualize the profit of each model in general, the profit of each model against the stock price and the profit of each model against the standard deviation of the stock price.

Generally speaking, the Deep Q-network yields the highest average profit compared to the Double Deep Q-Network and the Dueling Deep Q-Network. The result is consistent to other studies [14]. However, as expected, Deep Q-Network yields a higher volatility compared to two other methods. We display the distribution of profit generated by each model in Figure 8. It is clear that Double Q-Network seeks for profit, hence sometimes it generates the negative profit. We note that the LSTM model combined with the greedy algorithm is much more stable than other models because we buy and sell immediately when we detect any signal of changing the price.

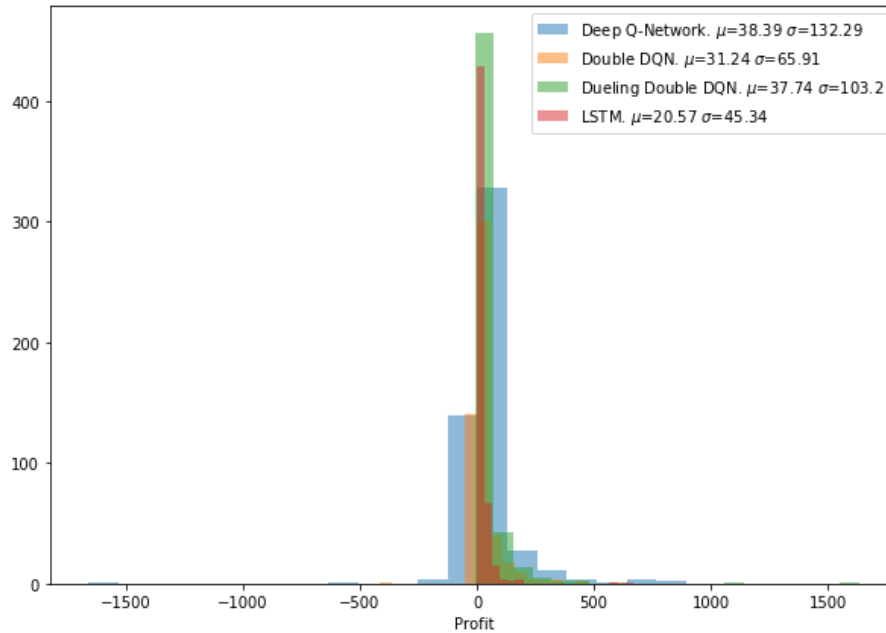


Fig. 8. Distribution of profit of three models

As described above, we visualize the profit against mean of stock price and standard deviation of stock price in the testing period in Figure 9 and 10. The

² <https://www.kaggle.com/borismarjanovic/price-volume-data-for-all-us-stocks-etfs>

³ <https://www.isods.org/>

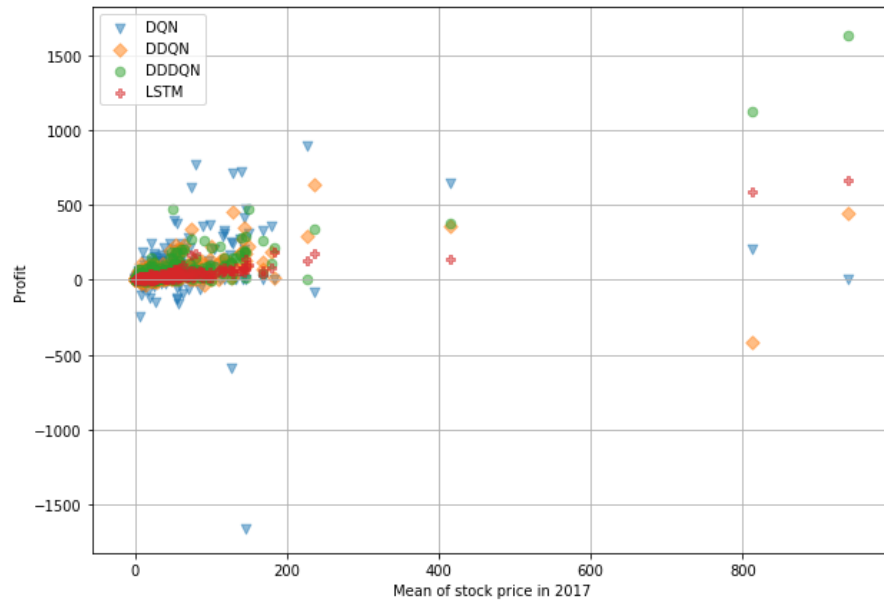


Fig. 9. Distribution of profit of three models

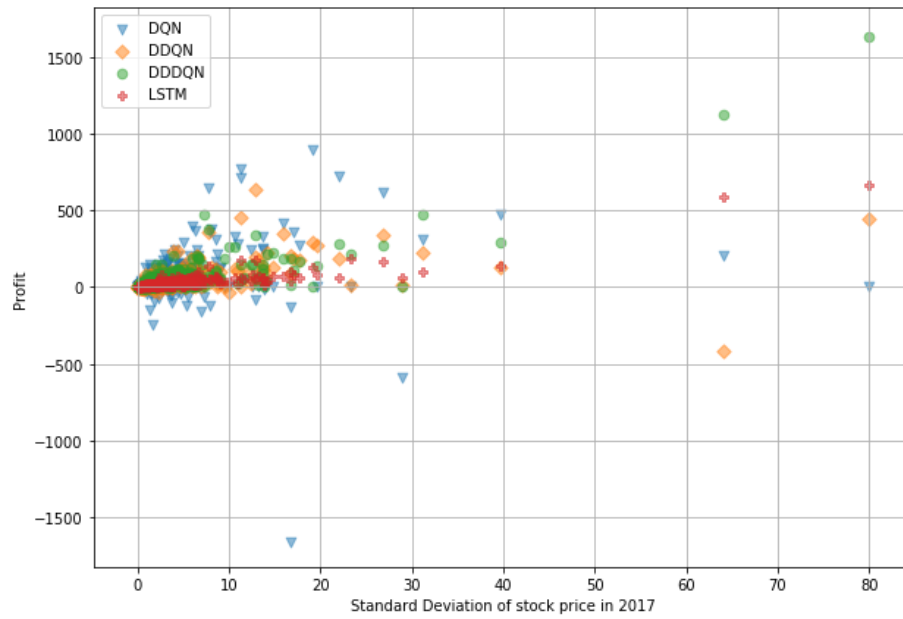


Fig. 10. Distribution of profit of three models

main idea is to see if a model behave differently or not in different segments of stock price. The general trends for all models are the expected profit range are higher given the higher stock price and stock volatility. The results are consistency with the principle of no arbitrage in finance [9] which basically stated that we cannot expect a higher profit without facing a higher risk.

6 Conclusions

In this paper we study the usage of Deep Q-Network for stock trading. We evaluated the performance of Deep Q-Network in large-scale real-world datasets. Deep Q-Network allow us to trade the stock directly without taking further optimization step like other supervised learning methods. Using only few hundreds samples, reinforcement learning algorithms variants based on Q-learning can generate the strategies that on average earning a positive profit.

In the future, we plan to incorporate multiple stock trading, i.e. portfolio management strategies, into the study. Furthermore, we will introduce different constraints into the model, for instance the maximum loss one can resist while using a model. Another approach is to integrate simulated behavior of users in non-cooperative or cooperative markets [5, 16].

7 Acknowledgment

We would like to thank the anonymous reviewer for valuable comments.

References

1. Azhikodan, A.R., Bhat, A.G., Jadhav, M.V.: Stock trading bot using deep reinforcement learning. In: *Innovations in Computer Science and Engineering*, pp. 41–49. Springer (2019)
2. Bisoi, R., Dash, P.K.: A hybrid evolutionary dynamic neural network for stock market trend analysis and prediction using unscented kalman filter. *Appl. Soft Comput.* **19**, 41–56 (2014)
3. Bradley, D.A.: *Stock Market Prediction: The Planetary Barometer and how to Use it*. Llewellyn Publications (1948)
4. Chong, E., Han, C., Park, F.C.: Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Syst. Appl.* **83**, 187–205 (2017)
5. Dang, Q., Ignat, C.: Computational trust model for repeated trust games. In: *Trustcom/BigDataSE/ISPA*. pp. 34–41. IEEE (2016)
6. Deng, Y., Bao, F., Kong, Y., Ren, Z., Dai, Q.: Deep direct reinforcement learning for financial signal representation and trading. *IEEE Trans. Neural Netw. Learning Syst.* **28**(3), 653–664 (2017)
7. Elton, E.J., Gruber, M.J., Brown, S.J., Goetzmann, W.N.: *Modern portfolio theory and investment analysis*. J. Wiley & sons, 9 edn. (2014)
8. Florescu, I., Mariani, M.C., Stanley, H.E., Viens, F.G.: *Handbook of High-frequency Trading and Modeling in Finance*, vol. 9. John Wiley & Sons (2016)

9. Föllmer, H., Schied, A.: Stochastic finance: an introduction in discrete time. Walter de Gruyter, 4 edn. (2016)
10. Göçken, M., Özçalici, M., Boru, A., Dosdogru, A.T.: Stock price prediction using hybrid soft computing models incorporating parameter tuning and input variable selection. *Neural Computing and Applications* **31**(2), 577–592 (2019)
11. Granger, C.W.J., Morgenstern, O.: Predictability of stock market prices. Heath Lexington Books (1970)
12. van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double q-learning. In: AAAI. pp. 2094–2100. AAAI Press (2016)
13. Henrique, B.M., Sobreiro, V.A., Kimura, H.: Literature review: Machine learning techniques applied to financial market prediction. *Expert Syst. Appl.* **124**, 226–251 (2019)
14. Hester, T., Vecerík, M., Pietquin, O., Lanctot, M., Schaul, T., Piot, B., Horgan, D., Quan, J., Sendonaris, A., Osband, I., Dulac-Arnold, G., Agapiou, J., Leibo, J.Z., Grusl, A.: Deep q-learning from demonstrations. In: AAAI. pp. 3223–3230. AAAI Press (2018)
15. Hiransha, M., Gopalakrishnan, E.A., Menon, V.K., Soman, K.: Nse stock market prediction using deep-learning models. *Procedia computer science* **132**, 1351–1362 (2018)
16. Ignat, C.L., Dang, Q.V., Shalin, V.L.: The influence of trust score on cooperative behavior. *ACM Transactions on Internet Technology (TOIT)* **19**(4), 46 (2019)
17. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E., et al.: Adaptive mixtures of local experts. *Neural computation* **3**(1), 79–87 (1991)
18. Jangmin, O., Lee, J., Lee, J.W., Zhang, B.T.: Adaptive stock trading with dynamic asset allocation using reinforcement learning. *Information Sciences* **176**(15), 2121–2147 (2006)
19. Jiang, X., Pan, S., Jiang, J., Long, G.: Cross-domain deep learning approach for multiple financial market prediction. In: IJCNN. pp. 1–8. IEEE (2018)
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. pp. 1106–1114 (2012)
21. Längkvist, M., Karlsson, L., Loutfi, A.: A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters* **42**, 11–24 (2014)
22. Lee, J.W.: Stock price prediction using reinforcement learning. In: ISIE 2001. 2001 IEEE International Symposium on Industrial Electronics Proceedings (Cat. No. 01TH8570). vol. 1, pp. 690–695. IEEE (2001)
23. Lo, A.W., Mamaysky, H., Wang, J.: Foundations of technical analysis: Computational algorithms, statistical inference, and empirical implementation. *The journal of finance* **55**(4), 1705–1765 (2000)
24. Long, W., Lu, Z., Cui, L.: Deep learning-based feature engineering for stock price movement prediction. *Knowl.-Based Syst.* **164**, 163–173 (2019)
25. Malkiel, B.G., Fama, E.F.: Efficient capital markets: A review of theory and empirical work. *The journal of Finance* **25**(2), 383–417 (1970)
26. Menon, V.K., Vasireddy, N.C., Jami, S.A., Pedamallu, V.T.N., Sureshkumar, V., Soman, K.: Bulk price forecasting using spark over NSE data set. In: International Conference on Data Mining and Big Data. pp. 137–146. Springer (2016)
27. Mitra, S.K.: How rewarding is technical analysis in the indian stock market? *Quantitative Finance* **11**(2), 287–297 (2011)
28. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M.A.: Playing atari with deep reinforcement learning. *CoRR abs/1312.5602* (2013)

29. Nazário, R.T.F., e Silva, J.L., Sobreiro, V.A., Kimura, H.: A literature review of technical analysis on stock markets. *The Quarterly Review of Economics and Finance* **66**, 115–126 (2017)
30. Nison, S.: *Japanese candlestick charting techniques: a contemporary guide to the ancient investment techniques of the Far East*. Penguin (2001)
31. Park, C.H., Irwin, S.H.: What do we know about the profitability of technical analysis? *Journal of Economic Surveys* **21**(4), 786–826 (2007)
32. Patel, J., Shah, S., Thakkar, P., Kotecha, K.: Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Syst. Appl.* **42**(1), 259–268 (2015)
33. Preis, T., Moat, H.S., Stanley, H.E.: Quantifying trading behavior in financial markets using google trends. *Scientific reports* **3**, 1684 (2013)
34. Rilling, G., Flandrin, P., Gonçalves, P., et al.: On empirical mode decomposition and its algorithms. In: *IEEE-EURASIP workshop on nonlinear signal and image processing*. vol. 3, pp. 8–11. NSIP-03, Grado (I) (2003)
35. Schulmeister, S.: Profitability of technical stock trading: Has it moved from daily to intraday data? *Review of Financial Economics* **18**(4), 190–201 (2009)
36. Sewak, M.: *Deep Reinforcement Learning - Frontiers of Artificial Intelligence*. Springer (2019)
37. Simon, H.A.: A behavioral model of rational choice. *The quarterly journal of economics* **69**(1), 99–118 (1955)
38. Sutton, R.S., Barto, A.G.: *Reinforcement learning: An introduction*. MIT press (2018)
39. Thomsett, M.C.: *Getting started in fundamental analysis*. John Wiley & Sons (2006)
40. Wang, J., Leu, J.: Stock market trend prediction using arima-based neural networks. In: *ICNN*. pp. 2160–2165. IEEE (1996)
41. Wang, Z., Schaul, T., Hessel, M., van Hasselt, H., Lanctot, M., de Freitas, N.: Dueling network architectures for deep reinforcement learning. In: *ICML. JMLR Workshop and Conference Proceedings*, vol. 48, pp. 1995–2003. JMLR.org (2016)
42. Zhai, Y.Z., Hsu, A.L., Halgamuge, S.K.: Combining news and technical indicators in daily stock price trends prediction. In: *ISNN (3). Lecture Notes in Computer Science*, vol. 4493, pp. 1087–1096. Springer (2007)
43. Zhang, G.P.: Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing* **50**, 159–175 (2003)
44. Zhang, X., Li, A., Pan, R.: Stock trend prediction based on a new status box method and adaboost probabilistic support vector machine. *Appl. Soft Comput.* **49**, 385–398 (2016)
45. Zhou, F., Zhou, H., Yang, Z., Yang, L.: EMD2FNN: A strategy combining empirical mode decomposition and factorization machine based neural network for stock market trend prediction. *Expert Syst. Appl.* **115**, 136–151 (2019)