

Review

A Survey of Forex and Stock Price Prediction Using Deep Learning

Zexin Hu [†], Yiqi Zhao [†] and Matloob Khushi ^{*†} 

School of Computer Science, The University of Sydney, Building J12/1 Cleveland St.,
Camperdown, NSW 2006, Australia; zehu4485@uni.sydney.edu.au (Z.H.); yzha9512@uni.sydney.edu.au (Y.Z.)

* Correspondence: matloob.khushi@sydney.edu.au

† The authors contributed equally; both authors should be considered the first author.

Abstract: Predictions of stock and foreign exchange (Forex) have always been a hot and profitable area of study. Deep learning applications have been proven to yield better accuracy and return in the field of financial prediction and forecasting. In this survey, we selected papers from the Digital Bibliography & Library Project (DBLP) database for comparison and analysis. We classified papers according to different deep learning methods, which included Convolutional neural network (CNN); Long Short-Term Memory (LSTM); Deep neural network (DNN); Recurrent Neural Network (RNN); Reinforcement Learning; and other deep learning methods such as Hybrid Attention Networks (HAN), self-paced learning mechanism (NLP), and Wavenet. Furthermore, this paper reviews the dataset, variable, model, and results of each article. The survey used presents the results through the most used performance metrics: Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), Mean Square Error (MSE), accuracy, Sharpe ratio, and return rate. We identified that recent models combining LSTM with other methods, for example, DNN, are widely researched. Reinforcement learning and other deep learning methods yielded great returns and performances. We conclude that, in recent years, the trend of using deep-learning-based methods for financial modeling is rising exponentially.

Keywords: deep learning; stock; foreign exchange; financial prediction; survey



Citation: Hu, Z.; Zhao, Y.; Khushi, M.

A Survey of Forex and Stock Price
Prediction Using Deep Learning.

Appl. Syst. Innov. **2021**, *4*, 9.

<https://doi.org/10.3390/asi4010009>

Received: 24 December 2020

Accepted: 19 January 2021

Published: 2 February 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The share market is a snapshot of future growth expectations of companies as well as the economy. Many factors have attributed to stock price fluctuation, which includes but is not limited to macroeconomic factors, the market anticipation, and confidence in the company's management and operation. The advancement of technology allows the public to access a larger quantity of information in a timelier manner. This means that stock analysis has become more and more difficult as a considerable amount of data has to be processed in a relatively short time. People hope that the progress made in big data, especially in the deep learning field, can help them analyze stock information [1].

Foreign exchange (Forex) is one of the largest financial markets in the world. The prediction of the exchange rates can provide investors with useful decision-making references to increase return and to reduce risk. However, the exchange rate is always under the influence of many factors, such as countries' economies, politics, society, international situation, etc., so the complexity of the matter has made Forex prediction and forecasting a challenging research topic [2]. Nowadays, Forex forecasting tasks apply many different deep learning models as the computer and artificial intelligence technology mature.

Forex and stocks are similar in many aspects. For example, they both have comparable technical indicators, both have similar charts (candle chart), and both could be affected by a country's market sentiment. Therefore, this paper discusses the application of deep learning in both Forex and the stock market and explores the impact of different deep learning methods on price trend prediction accuracy.

Continuous development in the AI field leads to the wide use of deep learning techniques in many research fields and practical scenarios. Applications include natural language processing, image recognition, medical predictions, and more. The neural networks used in these applications have also developed and improved due to the rise of deep learning. For example, reinforcement learning has gained popularity since AlphaGo defeated the best chess player at the time by using it, and reinforcement learning has been implemented in the financial prediction field since then [3]. These technological breakthroughs have given the stock and Forex prediction models a solid foundation to start and greater room to improve.

The highly complex nonlinear relationship of deep learning can fully describe the complex characteristics of the influencing factors. Many other fields have verified the accuracy of a deep learning model for prediction accuracy, such as image classification and gene analysis. Research results are also obtained for time-series data analysis and prediction with a deep learning algorithm; for example, deep learning is used to predict offline store traffic [4]. Overall, deep learning models have excellent performances in other research fields. Therefore, it is feasible to predict stock and Forex trends with deep learning.

Financial researchers around the world have been studying and analyzing the changes in the stock and Forex markets. The broadening application of artificial intelligence has led to an increasing number of investors using deep learning model to predict and study stock and Forex prices. It has been proven that the fluctuation in stock and Forex price could be predicted [5]. Different from traditional statistical and econometric models, deep learning can describe complex influencing factors.

Therefore, this paper investigates the different effects of different deep learning methods on stock and Forex forecasting according to existing published papers. This survey will analyze each paper from the following aspects: 1. What is the dataset of this paper? 2. What is the variable of this paper? 3. What kind of deep learning model had been adopted? 4. What is the result of the prediction model?

The structure of this paper will be as follows: firstly, the introduction of Forex and stock combined with deep learning; secondly, the criteria and research methods of the article selected by the survey; thirdly, the impact and analysis of different deep learning methods on stocks and Forex prediction; fourthly, the discussion and analysis of the above methods; and finally, a conclusion of the whole paper.

2. Related Deep Learning Methods and Input Introduction

2.1. Convolutional Neural Network (CNN)

CNN was widely used in the field of image recognition because of its powerful pattern recognition ability; its use was also extended to the field of economic prediction. Similar to the traditional neural network, CNN is composed of multiple neurons connected by a hierarchical structure, and the weights and bias between layers can be trained. CNN is different from the network structure of a fully connected network such as deep belief network (DBN), Sparse Autoencoder (SAE), backpropagation (BP), as the CNN can share the weight among the neurons in each layer of the network. Hence, the model significantly reduces the weight of the network and avoids falling into dimensional disaster and local minimization [6].

If the characteristics of the stock market at a specific time point are regarded as a feature graph, CNN has the potential to extract the characteristics of the stock market at the corresponding period from these feature graphs. Therefore, CNN can be used to build a timing-selection model and can ultimately be used to complete the construction of the timing-selection strategy.

2.2. Recurrent Neural Network (RNN)

RNN belongs to the neural network, and it is good at modelling and processing sequential data. The specific expression is that the RNN is able to memorize the previous state, and the previous state can be used in the current state calculation. The different

hidden layers are non-independent, and the input of the current hidden layer includes not only the output of the input layer but also the output of the previously hidden layer. For this reason, RNN has a good performance in dealing with sequential data.

The advantage of RNN is that it considers the context of data in the process of training, which is very suitable for the scenario of stocks and Forex because the fluctuation at a particular time often contains some connection to the previous trend.

2.3. Long Short-Term Memory (LSTM)

The LSTM model is one of the variants of the RNN. Its core contribution is to introduce the design of self-loop to generate the path of a gradient which could continuously flow for an extended period. The weight of the self-loop is also updated in each iteration, which solves the gradient vanishing problem that is easily generated when the RNN model updates the weights [7].

The modelling of a time series is essentially a process of nonlinear parameter fitting. The LSTM model performs well to reveal the correlation of a nonlinear time-series in the delay state space and to realize the purpose of stock prediction [8]. The stock or Forex trend prediction model based on LSTM obtained the corresponding data characteristics from the stock or Forex history data.

2.4. Deep Neural Network (DNN)

DNN is a neural network with at least one hidden layer. It provides modelling for complicated nonlinear functions and has a high-level abstraction ability, which means that the fitting power of the model is significantly improved. Meanwhile, it is a kind of discriminant model which could be trained through the backpropagation algorithm.

Since the DNN is good at dealing with prediction problems with sizable data and complicated nonlinear mapping relations, an intelligent stock and Forex prediction system can be designed based on a DNN to predict stock and Forex trends [9]. Hopefully, the model is able to achieve far higher accuracies than human beings.

2.5. Reinforcement Learning

Reinforcement learning is one of the deep learning methods that focuses on how to act according to the current situation to profit maximization. In reinforcement learning, there are two basic elements: state and action. A strategy is defined as performing a particular action in a specific state. All the learner has to do is learn a good strategy by continually exploring and learning.

If the state is regarded as the attribute and the action is the label, it is easy to know that both supervised learning and reinforcement learning try to find a map and to infer the label/action from the known attribute/state. In this way, the strategy in reinforcement learning is equivalent to the classification/regression in supervised learning. However, in practical problems, reinforcement learning does not have such labelling information as that of supervised learning, and the results are often obtained after an attempt of the action. Therefore, reinforcement learning continuously adjusts the previous strategy through feedback from the results; for this reason, this algorithm would learn in which state one should take which step to have the most beneficial result.

Therefore, reinforcement learning learns the best timing trading action (selecting the best price, trading duration, and order size) according to the market response. It can view the contextual information (price, news, public opinion, interest rate, fees, trading action, returns, losses, etc.) of the transaction as an environment of reinforcement learning. Gains or losses could be thought of as the reward for learning, trading actions could be thought of as actions, and factors could be thought of as states or observations to realize the prediction of the stock and Forex trends.

2.6. Other Deep Learning Methods

In this paper, we also discuss the prediction of stock and Forex trends by other deep learning methods [10], for example, Hybrid Attention Networks (HAN), self-paced learning mechanism (NLP), multi-filters neural network (MFNN), and Wavenet. The frequency of these methods in the selected articles is not high so they will be discussed together in Section 4.1.6.

3. Review Methodology and Criteria

3.1. Paper Selection Methods

In the past few years, there have been many deep learning model papers on stock and Forex forecasting. The articles analyzed in this paper were all from the Digital Bibliography & Library Project (DBLP) computer science bibliography and Microsoft Academic database.

Firstly, the keywords were searched in DBLP: “CNN stock/Forex”, “LSTM stock/Forex”, “Deep learning stock/Forex”, “RNN stock/Forex”, and “Reinforcement learning stock/Forex”. These keywords are necessary to ensure the papers reviewed are relevant to our topic. The keywords were searched in Microsoft Academic, and then, the filters “2015–2021” and “Deep learning” were applied.

Secondly, the quality of the selected articles from DBLP was ensured by excluding all journals and conferences that were informally published and the quality of the selected articles from Microsoft Academic were controlled by excluding all journals and conferences that were informally published as well as those without a minimum of 5 citations. Filtering on citations as the search method was designed to detect understudied areas within this field; a paper with 5 or more citations can be an indication that the area could be potentially explored. Furthermore, it should be noted that we only considered papers with a novel model, with the implementation of existing models not analyzed in this review.

At the same time, the timeliness of the survey was provided by focusing on publications after 2015. Among them, there were 4 papers in 2015, 1 paper in 2016, 14 papers in 2017, 29 papers in 2018, 28 papers in 2019, and 10 papers in 2020; in total, 86 existing papers were reviewed.

3.2. Selected Paper Statistics

In this paper, the goal was to study the use of six different deep learning methods in Forex/stock (which are CNN, LSTM, DNN, RNN, reinforcement learning, and other deep learning methods) with different datasets, input variables, and model types. All the results were compared together for discussion and for conclusion after being individually analyzed.

Figure 1 showed the annual distribution of the papers collected and reviewed. Figure 2 showed the distribution of the different methods in this paper. Figure 3 shows the systematic literature review in this paper. Table 1 showed the distribution of different types of articles in this paper.

Table 1. The distribution of different types of articles in this paper.

| Type of Method | Total Paper | Number of Papers in Journals | Number of Papers in Conferences |
|-----------------------------|-------------|------------------------------|---------------------------------|
| CNN | 18 | 6 | 12 |
| LSTM | 38 | 13 | 25 |
| DNN | 9 | 5 | 4 |
| RNN | 5 | 2 | 3 |
| Reinforcement Learning | 8 | 1 | 7 |
| Other Deep Learning Methods | 10 | 3 | 7 |

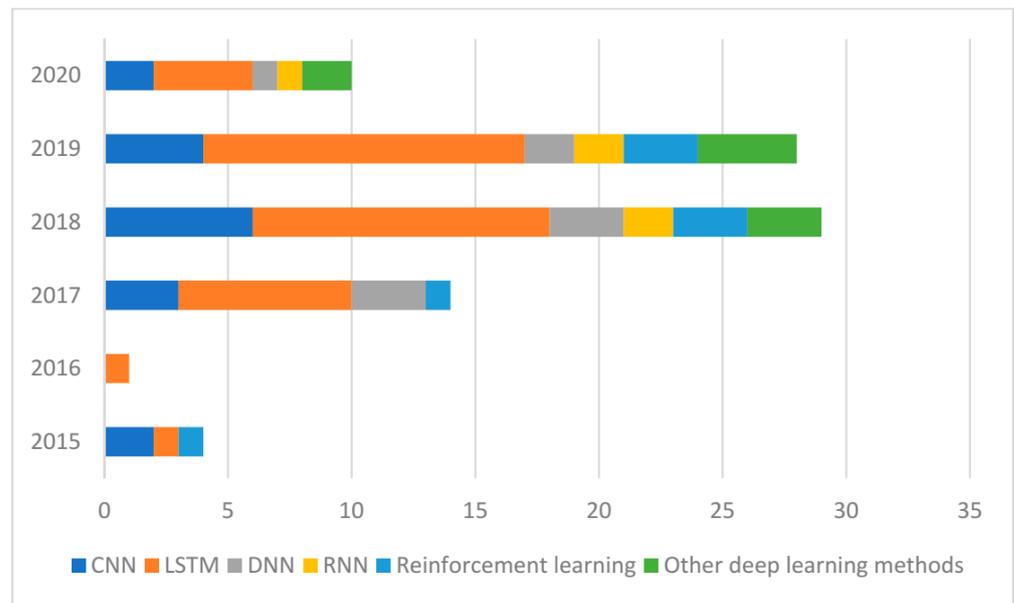


Figure 1. The annual distribution of papers collected and reviewed.

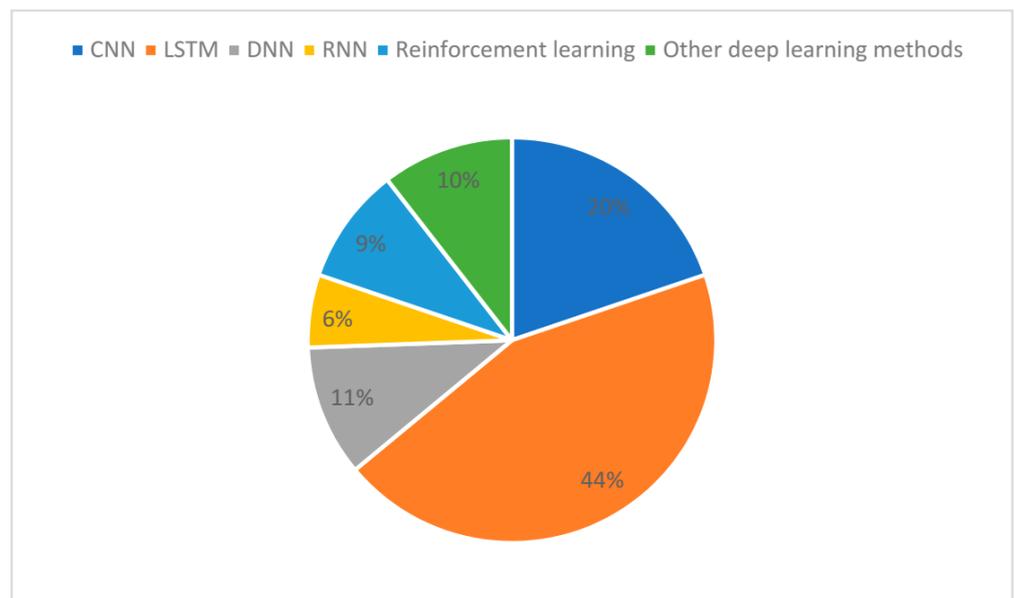


Figure 2. The distribution of the different methods in this paper.

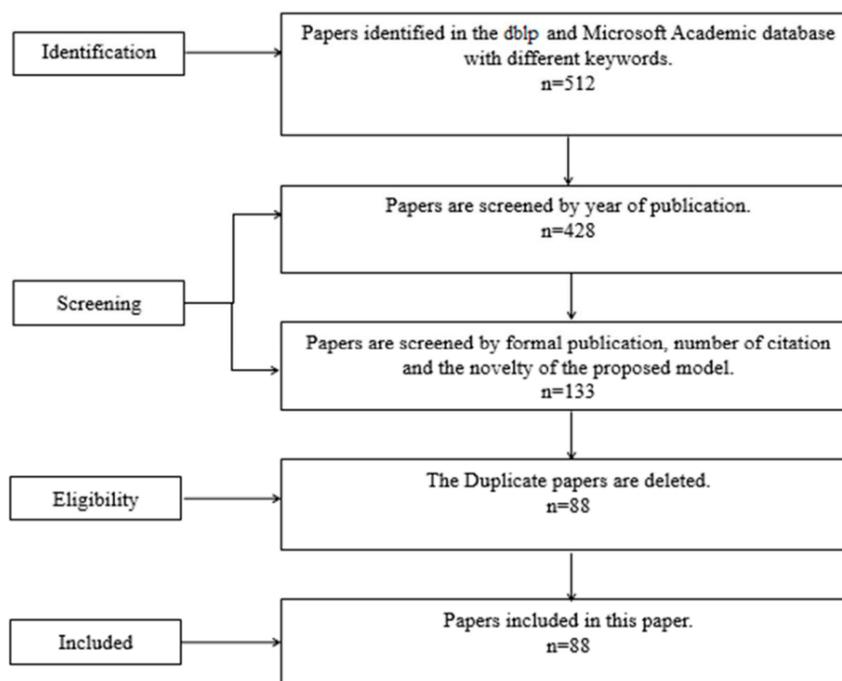


Figure 3. The systematic literature review in this paper.

4. Results

4.1. Papers Descriptions

4.1.1. CNN

Seventeen articles that used CNN technology for stock and Forex prediction are briefly described below. Table 2 shows the author, variables, dataset, and model of the papers mentioned.

Table 2. Information on the papers which used the Convolutional neural network (CNN) model.

| Reference No. | Author | Dataset | Variables | Model |
|---------------|----------------|--|---|--|
| [11] | Maqsood, H. | 1. Top 4 performing companies of US, Hong Kong, Turkey, and Pakistan 2. Twitter dataset | 1. Open price, high price, low price, AdjClose price, volume, and close price 2. Sentiment (positive, neutral, and negative sentiments). | CNN |
| [12] | Patil, P. | 1. News collected on a financial website. 2. 30 stocks of fortune 500 companies, such as WMT, XOM, and AAPL | Adjacency matrix calculated by the correlation coefficient and using news co-mentions | Graph convolution neural network (GCN) |
| [13] | Sim, HS. | S&P 500 min data | Close price | CNN |
| [14] | Hoseinzade, E. | S&P 500, NASDAQ, Dow Jones industrial average, NYSE, and Russell | 82 variables including high, low, and close price; volume, RSI, KD, WR, etc. | 2D-CNN and 3D-CNN |
| [15] | Eapen, J. | Standard and Poor’s (S&P) 500 stock dataset | Close price | CNN-bidirectional LSTM |
| [16] | Yang, H. | S&P 500 Index ETF (SPY) | High, low, and close price; volume, RSI, KD, WR, ROC, and CCI | CNN with MICFS |
| [17] | Cai, S. | 1. Crawling financial news 2. Baidu Index | 1. word vector; headline and keyword training set in the news 2. Close price | CNN-LSTM |
| [18] | Oncharoen, P. | Reuters and Reddit Standard & Poor’s 500 Index (S&P500) and Dow Jones Industrial Average (DJIA) | Word vectors of headlines Close prices, Bollinger band, RSI, and stochastic oscillator | CNN-LSTM |
| [19] | Liu, Y. | 1. Thomson Reuters 2. Standard & Poor’s 500 index (S&P 500) | 1. Financial news corpus with headlines from Apple 2. Open price; close price; high price; low price; volume; stochastic oscillator (%K); Larry William (LW) %R indicator; and Relative Strength Index (RSI) | TransE-CNN-LSTM |
| [20] | Selvin, S. | NSE listed companies: Infosys, TCS, and CIPLA | Close price | CNN sliding-window model |

Table 2. Cont.

| Reference No. | Author | Dataset | Variables | Model |
|---------------|---------------------|--|---|---|
| [21] | Liu, S. | Chinese stocks from the SINA FINANCE (not given specific data) | Close price | CNN-LSTM |
| [22] | Gudelek, M.U. | Exchange-Traded Funds (ETF) | Close price, RSI, SMA, MACD, MFI, Williams %R, the stochastic oscillator, and the ultimate oscillator | 2D-CNN |
| [23] | Ding, X. | 1. S&P 500 index 2. Reuters and Bloomberg | 1. close price 2. long-term, mid-term, and short-term feature vectors of news headlines | NTN-CNN |
| [24] | Zhao, Y. and Khushi | USDJPY exchange rate | 1. Momentum indicators: RSI5, RSI10, RSI20, MACD, MACDhist, MACDsignal, Slowk, Slowd, Fastk, Fastd, WR5, WR1, WR2, ROC5, ROC1, ROC20, CCI5, CCI10, and CCI20 2. Volume indicators: ATR5, ATR10, ATR20, NATR5, NATR10, NATR20, and TRANGE | Wavelet denoised-ResNet CNN with light GBM |
| [3] | Chen, S. and He, H. | Chinese stock market | Closing price | CNN |
| [25] | Chen, Jou-Fan | Taiwan index futures | Open, high, low, and closing price | GAF + CNN |
| [26] | Wen, M | S%P500 | Open, high, low, close, adjusted close, and volume. | CNN |

Maqsood, H. proposed a CNN model that made use of the price and sentiment analysis as input and compared the proposed model with linear regression and Support vector machine (SVM). He concluded that not all significant events have a serious impact on stock exchange prediction. However, more important local events could affect the performance of prediction algorithms [11].

Patil, P. proposed a new network using graph theory and CNN, which leveraged spatiotemporal relationship information between different stocks by modelling the stock market as a complex network. Meanwhile, the model used both stock indicators and financial news as input [12].

Sim, H.S. proposed a CNN network that uses 9 technical indicators (close price, Simple moving average (SMA), Exponential moving average (EMA), Rate of change (ROC), Moving average convergence/divergence (MACD), Fast%K, Slow%D, upper band, and lower band) to verify the applicability of the CNN method in the stock market. He concluded that the use of technical indicators in stock price forecasting by CNN has no positive effect [13].

Hoseinzade, E. proposed two models: 2D-CNN and 3D-CNN using 82 different technical indicators. These two structures could improve the predictive performance of the baseline algorithm by about 3–11% [14].

Eapen, J. proposed a model that had multiple pipelines of CNN and bidirectional LSTM units. It could improve prediction performance by 9% using a single pipeline deep learning model and by over a factor of six using support vector machine regressor model on the S&P 500 grand challenge dataset [15].

Yang, H.M. proposed a multi-indicator feature selection for stock index prediction based on a multichannel CNN structure without subsampling, named the MI-CNN framework. In this method, candidate indicators were selected by the maximal information coefficient feature selection (MICFS) approach to ensure the correlation with stock movements whilst reducing redundancy between different indicators [16].

Cai, S. proposed the CNN and LSTM forecasting system with financial news and historical data of the stock market. It generated seven prediction models. According to the ensemble learning method, the seven models were constructed into one ensemble model to obtain an aggregated model. Unfortunately, all models had a lower prediction accuracy [17].

Oncharoen, P. proposed a new framework to train a DNN for stock market prediction. A new loss function was developed by adding a risk–reward function, which was derived from the trading simulation results [18].

Liu, Y. proposed to incorporate a joint model using the TransE model for representation learning and a CNN, which extracted features from financial news articles. Then, he

combined the model with LSTM to predict the price movement. The model could improve the accuracy of text feature extraction while reducing the sparseness of news headlines [19].

Selvin, S. used CNN, LSTM, and RNN architectures to forecast the price of NSE-listed companies and compared their performance. The final results showed that CNN is the best architecture to predict the price of stock because it could identify the trend of the directional change [20].

Liu, S. proposed a CNN-LSTM model, and the model performed a basic momentum strategy and benchmark model for which the return rates were 0.882 and 1.136, respectively. The CNN part could extract useful features even from low signal-to-noise time-series data, and the LSTM part could predict future stock prices with high accuracy. Then, the predicting outcomes were used as timing signals [21].

Gudelek, M.U. proposed a 2D-CNN model. This model adopted a sliding window approach and then generated images by taking snapshots that are bounded by the window over a daily period. The model could predict the next day's prices with 72% accuracy and ended up with 5:1 of the initial capital [22].

Ding, X. proposed a deep learning method for the event-driven stock market prediction. Firstly, events were extracted from news text and represented as dense vectors, trained using a novel neural tensor Network (NTN). Secondly, CNN was used to model both the short-term and long-term influences of events on stock price movements [23].

Zhao, Y. proposed a Wavelet Denoised-ResNet CNN for Forex exchange rate prediction. The technical indicators were treated as an image matrix. The image matrix was first denoised using Wavelet method and then processed by ResNet CNN. Lastly, LightGBM was used to replace the softmax layer to output a prediction [24].

Chen, S. and He, H. proposed a CNN model with a novel architecture, where it generated better results than the benchmark RNN model [3].

Chen, Jou-Fan proposed a novel CNN model which utilized the power of Gramian Angular Field. The results produced were average, but it is an interesting research direction [25].

Wen, M. proposed a CNN model which relied on the reconstructed of time series, it turned the time series into segments. Then, CNN was used to classify each segment. The model generated good results [26].

4.1.2. RNN

Five articles that used RNN technology for stock and Forex prediction are briefly described below. Table 3 shows the author, variables, dataset, and model of the papers mentioned.

Table 3. Information on the papers which used the Recurrent Neural Network (RNN) model.

| Reference No. | Author | Dataset | Variables | Model |
|---------------|---------------------|--|---|---------------------------|
| [27] | Ni, L. | EURUSD, AUDUSD, XAUUSD, GBPJPY, EURJPY, GBPUSD, USDCHE, USDJPY, and USDCAD | Open price, close price, highest price, and lowest price | C-RNN |
| [28] | Li, C. | China Security Index: CSI300, CSI200, and CSI500 | Open price, high price, low price, close price, volume, and amount. | Multi-task RNN with MRFs |
| [29] | Chen, W | HS300 | 1. Technical features: open price, high price, low price, close price, volume, price change, price limit, volume change, volume limit, amplitude, and difference. 2. Content features: sentiment features and LDA features | RNN-Boost |
| [30] | Zhang, R. | Sandstorm sector of the Shanghai Stock Exchange | Open price, close price, highest price, lowest price, and the daily trading volume Momentum indicators: average directional movement index, absolute price oscillator, Aroon oscillator, balance of power, commodity channel index, Chande momentum oscillator, percentage price oscillator, moving average convergence divergence, Williams, momentum, relative strength index, stochastic oscillator, and triple exponential average | C-RNN (DWNN) |
| [31] | Zeng, Z. and Khushi | USDJPY exchange rate | Volatility indicators: average true range, normalized average true range, and true range | Attention-based RNN-ARIMA |

Ni, L. proposed a CRNN model to predict the price of 9 kinds of Forex pairs. The results showed that the proposed model performed much better than the LSTM model and the CNN model [27].

Li, C. proposed a multi-task RNN model with Markov Random Fields (MRF). The multi-multilayer perceptron (MMLP) was proposed to automatically extract diversified and complementary features from individual stock price sequences, which means that there is no need for the technical indicators. Features learned by MMPL were passed to a binary MRF with a weighted lower linear envelope energy function to utilize intra-clique higher-order consistency between stocks [28].

Chen, W. proposed an RNN-Boost model that made use of the technical indicators and sentiment features, and Latent Dirichlet allocation (LDA) features to predict the price of stocks. Its results showed that the proposed model outperformed the single-RNN model [29].

Zhang, R. proposed a Deep and Wide Neural Networks (DWNN) model, where CNN’s convolution layer was added to the RNN’s hidden state transfer process. The results showed that the DWNN model could reduce the prediction mean squared error by 30% compared to the general RNN model [30].

Zeng, Z proposed a novel Attention-based RNN (ARNN), where wavelet denoised input was passed to ARNN. The forecast was calculated using the Autoregressive integrated moving average (ARIMA) and the output of the ARNN model [31].

4.1.3. LSTM

Twenty-seven articles that used LSTM technology for stock and Forex prediction are briefly described below. Table 4 shows the author, variables, dataset, and model of the papers mentioned.

Table 4. Information on the papers which used the Long Short-Term Memory (LSTM) model.

| Reference No. | Author | Dataset | Variables | Model |
|---------------|------------------------|---|--|--|
| [32] | Nikou, M. | iShares MSCI United Kingdom | Close price | LSTM |
| [33] | Fazeli, A. | S&P 500 | Open price, high price, low price, close price, adjusted close price, volume, volatility, Williams %R, and RSI | LSTM |
| [34] | Xu, Y. | Microsoft (MSFT), PO logistics (XPO), and AMD Daily stock price data are collected from Yahoo Finance from 11 industries Finance tweets from a social media company StockTwits | Open price, high price, low price, close price, AD, ADX, EMA, KAMA, MA, MACD, RSI, SAR, AMA, etc. and finance tweet sentiment | Attention-based LSTM |
| [35] | Lakshminarayanan, S.K. | Dow Jones Industrial Average (DJIA) | Close price, moving average, crude oil price, and gold price | LSTM |
| [36] | Rana, M. | Spanish stock company Acciona | Close price | LSTM |
| [37] | Naik, N. | CIPLA stock, ITC stock, TCS stock, ONGC stock, and Nifty index | Close price | RNN with LSTM |
| [38] | Nguyen, D.H.D. | NASDAQ stock market: GE, AAPL, SNP, and FB | Trade volume, open, close, high, low, and adjusted close prices | Dynamic LSTM |
| [39] | Lai, C.Y. | Foxconn, Quanta, and Formosa Taiwan Cement and Taiwan Semiconductor | KD, OBV, MACD, RSI, and the average of the previous five days’ stock market information (open, high, low, and volume, and close) | LSTM |
| [40] | Hossain, M.A. | S&P500 | Open price, close price, volume | LSTM with GRU |
| [41] | Baek, Y. | KOSPI200 and S&P500 | Close price | LSTM with prevention module, prediction module |
| [42] | Kim, H.Y. | KOSPI 200 | Close price | GEW-LSTM |
| [43] | Li, H. | CSI-300 | Open price | Attention-based Multi-Input LSTM |
| [44] | Cheng, L.-C. | Data from Taiwan Stock Exchange Corporation (not given specific data) | Open price, close price, low price, high price, volume, KD, MA, RSV, etc. | Attention-based LSTM |
| [45] | Shah, D. | Tech Mahindra (NSE: TECHM) BSEsensex | Close price | LSTM |
| [46] | Lin, B.-S. | Taiwan Stock Exchange Corporation (TWSE) | Trade volume, transaction, open price, highest price, lowest price, close price, KD, RSI, and Bollinger Bands (BBands) | LSTM |
| [47] | Skehin, T. | Facebook Inc. (FB), Apple Inc. (AAPL), Amazon.com Inc (AMZN), Netflix Inc. (NFLX), and Alphabet Inc. (GOOG) in NASDAQ of S&P 500 | Close price | ARIMA-LSTM-Wavelet |

Table 4. Cont.

| Reference No. | Author | Dataset | Variables | Model |
|---------------|--------------------------------------|---|---|--|
| [48] | Zhang, X. | China's A-share market | The open price, close price, highest price, lowest price, and trading volume and 11 indicators | RNN with LSTM |
| [49] | Achkar, R. | Facebook stocks, Google stocks, and Bitcoin stocks collected from Yahoo finance | Close price | RNN with LSTM |
| [50] | Zeng, Y. | SSE50 index | N/A | LSTM |
| [51] | Shao, X. | Ping An Bank | Close price | Kmeans-LSTM |
| [52] | Zhao, Z. | SSE Composite Index, SSE 50, CSI 500, CSI 300, and SZSE Composite Index | Close price | Time-weighted LSTM |
| [53] | Nelson, D.M. | IBovespa index from the BM&F Bovespa stock exchange | Open price, close price, high price, low price, and volume exponentially weighted moving averages, etc. (175 indicators in total) | LSTM |
| [54] | dos Santos Pinheiro | 1. Standard&Poor's 500 index 2. Reuters and Bloomberg | 1. Financial domain news text (headlines instead of the full content) 2. Close price | NLP + LSTM |
| [55] | Buczowski, P. | Expert recommendation from TipRanks company | A stock identifier, a list of expert recommendations of varying length, and optional target labels (class) | LSTM with GRU |
| [4] | Akita, R. | Morning edition of the Nikkei newspaper Nikkei 225 | 1. Paragraph Vector 2. Open, close, highest, and lowest price | LSTM |
| [56] | Chen, K. | China stock market in Shanghai and Shenzhen from Yahoo finance | Volume, high, low, open, close price | LSTM |
| [57] | Qi, Ling and Khushi | Forex Exchange rate | Technical indicators: stock data, stock news, capital stock and shareholders, and financial analysis | LSTM |
| [58] | Pang, Xiong Wen | Chinese stock | Stock data | DeepLSTM with encoder |
| [59] | Feng, Fuli, et al. | NASDAQ and NYSE | Open, high, low, closing, volume, and technical indicators | LSTM with ranking relation |
| [60] | Chung, Hyejung, and Kyung-shik Shin. | Korea Stock Price Index | Closing price and sentiment data | LSTM with GA |
| [61] | Li, Jiahong, et al. | Chinese Stock Market | | LSTM with Naïve Bayes |
| [10] | Zhang, Kang, et al. | S&P 500 Index, Shanghai Composite Index, IBM from NYSE, MSFT from NASDAQ, and PingAn Insurance Company of China (PAICC) | Open, high, low, closing, volume, and technical indicators | LSTM with Generative Adversarial Network (GAN) |
| [62] | Jin, Zhigang, et al. | Apple stock price | Sentiment and stock price data | LSTM with sentiment analysis model |
| [63] | Long, Jiawei, et al. | Chinese Stock market | Market information including transaction records | LSTM with CNN |
| [64] | Chen, MY. | Chinese stock market | Sentiment information | LSTM |
| [65] | Qian, F. Chen, X | Chinese stock market | Closing price | LSTM with ARIMA |
| [66] | Li, Z. Tam, V. | Asian stock indexes | Closing price and technical indicators | LSTM with Wavelet Denoising |

Nikou, M. proposed an LSTM model and compared it with the ANN model, Support Vector Regression (SVR) model, and RF model. The results showed that the LSTM model performed better in the prediction of the closing prices of iShares MSCI United Kingdom than the other models mentioned in paper [32].

Fazeli, A. proposed an LSTM model; his result showed an improvement using LeakyReLU. The Mean square error (MSE) was below 0.1, whilst compared with the ReLU function, the MSE was well above 0.2. He examined the effect of the Relative Strength Index (RSI), Williams %R, and volatility on the loss of the model. It was shown that the model's loss was reduced by using only RSI, which contributed to the performance of the model [33].

Xu, Y. proposed an attention-based LSTM model that performs better than the regular LSTM. It was found that finance tweets that were posted from market closure to market open the next day had more predictive power on the next day's stock movement. The weighted sentiment on the max follower on StockTwits also performed much better than other methods [34].

Lakshminarayanan, S.K. proposed an LSTM model combined with crude oil price, gold price, and moving average, which performed much better than the LSTM model without them and the SVM model. It showed that the crude oil and gold prices had some impact on stock price prediction [35].

Rana, M. proposed an LSTM model that outperformed the LR and SVR models. He also compared the different activation functions with different optimizers and concluded that the tanh activation with the Adam algorithm performs best with an accuracy of 98.49% [36].

Naik, N. proposed an RNN with the LSTM model. The model had the ability to keep the memory of historical stock returns in order to forecast future stock return output. It was worth noting that recent stock information rather than old, related stock information was stored and used. The network also outperformed the Feed Forward ANN model [37].

Lai, C.Y. proposed a dynamic LSTM model. The results showed that stock prediction accuracy based on Mean absolute error (MAE), Mean absolute percentage error (MAPE), Root Mean Square Error (RMSE), and Mean square error (MSE) obtained by the dynamic LSTM model were much better than that by the static LSTM model. The dynamic model also consistently outperformed the linear models SA-5 and EMA-0.5 when predicting four stocks [38].

Lai, C.Y. proposed an LSTM model which used the average of the previous five days' stock market information (open, high, low, volume, and close) as the input value. The initial prediction was calculated using this value. The prediction was then used as part of the average of the stock price information for the next five days through the ARIMA method. Moreover, he utilized technical analysis indicators to consider whether to buy stocks, or to continue to hold stocks or to sell stocks [39].

Hossain, M.A. proposed an LSTM model followed by Gated recurrent unit (GRU). Both LSTM and GRU were powerful recurrent networks that could perform better and faster in terms of accuracy in regression-based prediction. The proposed model outperformed the LSTM only, GRU only, and GRU followed by LSTM models [40].

Baek, Y. proposed a novel data augmentation approach for stock market index forecasting through the ModAugNet framework, which consisted of two modules: an overfitting prevention LSTM module and a prediction LSTM module. The overfitting problems were mainly caused by the limited availability of data points for training [41].

Kim, H.Y. proposed some LSTM models to forecast stock price volatility that combined the LSTM model with various generalized autoregressive conditional heteroscedasticity (GARCH)-type models. He found that the Geneva Emotion Wheel (GEW)-LSTM model, which combined all three models, GARCH, exponential generalized autoregressive conditional heteroscedasticity (EGARCH), and exponential weighted moving average (EWMA), with LSTM, performed best [42].

Li, H. proposed an improved MI-LSTM based on LSTM and attention mechanism, which achieved better performance in extracting potential information and filtering noise. The model could assign different weights to different input series to keep the dominant status of the mainstream while absorbing information from leaky input gates [43].

Cheng, L.-C. proposed an attention-based LSTM model that could solve the problem of exploding and vanishing gradients and thus did not effectively capture long-term dependencies [44].

Shah, D. proposed an LSTM model which was compared with the DNN model. The proposed model was able to predict volatile movements in the true data. In general, it was able to recognize the directionality of the changes in the true data more accurately than the DNN [45].

Lin, B.-S. proposed an LSTM model to predict the price of the top 10 industries included in Taiwan Stock Exchange Corporation (TWSE). In the experimental results, most of the results were reasonable except for the Midland Holdings Limited (MTK) stock [46].

Skehin, T. proposed a linear Autoregressive Integrated Moving Average (ARIMA) model and LSTM network for each series to produce next-day predictions. Wavelet methods decomposed a series into approximation and detail components to better explain behavior over time. He combined these techniques in a novel ensemble model in an attempt to increase forecast accuracy [47].

Zhang, X. proposed a simple but efficient method to predict future stock return ranking without handcrafted features [48].

Achkar, R. proposed an approach to predict stock market ratios using artificial neural networks. It considered two different techniques—Best performance algorithm (BPA)-

Multilayer perceptron (MLP) and LSTM-RNN—their potential, and their limitations. And the LSTM-RNN model outperformed the other one slightly [49].

Zeng, Y. proposed an LSTM model based on the dataset SSE50, and the results showed that the accuracy was above 65% [50].

Shao, X. proposed a Kmeans-LSTM model that used the time window to divide the stock price sequential data into several equal subsequences. The K-means algorithm was used to cluster the stock price subsequences (He did not give the specific result data.) [51].

Zhao, Z. proposed a time-weighted LSTM. Unlike other models which treat data indiscriminately, the proposed model could carefully assign weights to data according to their temporal nearness towards the data that was used for prediction. The results showed that the proposed model outperformed the SVM, RNN, and Random Forest models [52].

Nelson, D.M. proposed an LSTM model which was based on the price history and technical analysis indicators. The results showed that the proposed model was better than the MLP and Random Forest models [53].

Dos Santos Pinheiro proposed a character-based neural language model for an event-based trading model that was combined with NLP and LSTM. The results showed that the proposed model performed better than some model proposed in other papers such as WI-RCNN and SI-RCNN [54].

Buczowski, P. tackled the problem of stock market predicting feasibility, especially when predictions were based only on a subset of available information, namely financial experts' recommendations. The analysis was based on data and results from the ISMIS 2017 Data Mining Competition [55].

Akita, R. proposed an approach that converted newspaper articles into their distributed representations via Paragraph Vector and modelled the temporal effects of past events on open prices about multiple companies with LSTM [4].

Chen, K. proposed an LSTM model to predict the price of the Chinese stock. The results showed that the accuracy was only 27.2% [56].

Qi, L. proposed an LSTM model to predict the price of the forex exchange rate. The model used technical indicators to calculate events, and then, the LSTM was used to make predictions based on the events [57].

Pang, X.W. proposed two improved deep LSTM models with embedded layers. The results showed its improvement over the benchmark [58].

Feng, F., et al. proposed a novel LSTM model that combined with stock relational model; it had a great performance compared with its benchmarks [59].

Chung, H. and Shin, K.S. proposed an LSTM model which was optimized by the genetic algorithm (GA). The novel hybrid model outperformed the benchmark [60].

Li, J., et al. proposed an LSTM model that was combined with a Naïve Bayes' model. This hybrid model outperformed the benchmarks and delivered promising results [61].

Zhang, K., et al. proposed a novel generative adversarial network based on MLP and LSTM. The model was able to achieve better results than some of the vanilla models [10].

Jin, Z., et al. proposed to incorporate a sentiment analysis model into LSTM; they successfully created a novel model which deliver a reasonable result [62].

Long, J., et al. proposed a hybrid model which utilized CNN to extract transactional information, and then, it was passed through an LSTM to predict the stock price. The model showed improvements over some of the vanilla benchmarks [63].

Chen, M. proposed an LSTM model which used the sentiment data collected. It combined both sentimental models as well as LSTM to produce a good result [64].

Qian, F. proposed an LSTM model which used ARIMA to improve its prediction power. The model yielded a reasonable result [65].

Li, Z. and Tam, V. proposed an LSTM model that used the wavelet denoising technique before passing through the LSTM model; the model produced good results [66].

4.1.4. DNN

Thirteen articles that used DNN technology for stock and Forex prediction are briefly described below. Table 5 shows the author, variables, dataset, and model of the papers mentioned.

Table 5. Information on the papers which used the Deep neural network (DNN) model.

| Reference Number | Author | Dataset | Variables | Model |
|------------------|------------------------------|---|--|--|
| [67] | Song, Y. | KOSPI and KOSDAQ | 715 novel input features (including moving average and disparity of stock price) | DNN |
| [68] | Naik, N. | NSE ICICI Bank SBI Bank Kotak Bank Yes Bank | SMA, exponential moving average, momentum indicator, stochastic oscillator, moving average convergence divergence, relative strength index, Williams R, accumulation distribution index, and commodity channel index | DNN |
| [69] | Chatzis, S.P. | FRED database and the SNL | Stock price, exchange rates, VIX index, gold price, TED spread, oil price, effective federal funds rate, and high yield bond returns | DNN |
| [70] | Abe, M. | MSCI | 25 variables: 1. book-to-market ratio; 2. earnings-to-price ratio; 3. dividend yield; 4. sales-to-price ratio; 5. cash flow-to-price ratio; 6. return on equity; 7. return on asset; 8. return on invested capital; 9. accruals; 10. sales-to-total assets ratio; 11. current ratio; 12. equity ratio; 13. total asset growth; 14. investment growth; 15. investment-to-assets ratio; 16. EPS revision (1 month); 17. EPS revision (3 months); 18. market beta; 19. market value; 20. past stock return (1 month); 21. past stock return (12 months); 22. volatility; 23. skewness; 24. idiosyncratic volatility; and 25. trading turnover | DNN |
| [71] | Nakagawa, K. | TOPIX index | 16 variables: 60 VOL, BETA, SKEW, ROE, ROA, ACCRUALS, LEVERAGE, 12-1MOM, 1MOM, 60MOM, PSR, PER, PBR, PCFR, CAP, and ILIQ | Deep factor model (DNN) with layer-wise relevance propagation and multiple factors |
| [72] | Chong, E. | KOSPI | Close price | DNN with autoencoder |
| [73] | Singh, R. | NASDAQ | 36 variables: Open price, high price, low price, close price, MA5, MA10, MA20, BIAS5, BIAS10, DIFF, BU, BL, K, D, ROC, TR, MTM6, MTM12, WR%10, WR%5, OSC6, OSC12, RSI6, RSI12, PSY, and the derivation of their calculation | (2D ² PCA) + DNN |
| [74] | Yu, Pengfei, and Xuesong Yan | S&P 500, DJIA, the Nikkei 225 (N 225), the Hang Seng index (HSI), the China Securities index 300 (CSI 300), and the ChiNext index | Closing price | DNN with phase-space reconstruction (PSR) and LSTM |
| [75] | Yong, Bang Xiang, et al. | Singapore stock market | Intraday prices | DNN |

Song, Y. proposed a DNN model with 715 novel input features configured on the basis of technical analysis. He also proposed a plunge filtering technique to improve the accuracy of the training model by collecting similar stocks. It is worth noting that the proposed model had great profitability [67].

Naik, N. proposed a DNN model that used the Boruta feature selection technique to solve the problem of technical indicator feature selection and identification of the relevant technical indicators. The results showed that his model performed much better than the ANN and SVM models [68].

Chatzis, S.P. proposed a DNN model which used Boosted approaches to predict stock market crisis episodes. According to his research, it was meaningful to know the stock market crisis to predict the price, even though his research was not specific to certain prediction methods [69].

Abe, M. proposed a DNN network, and his results showed that DNNs generally outperform shallow neural networks and that the best networks also outperformed representative machine learning models [70].

Nakagawa, K. proposed a deep factor model and a shallow model with DNN. The deep factor model outperformed the linear model. This implied that the relationship between the stock returns in the financial market and the factors is nonlinear rather than linear. The deep factor model also outperformed other machine learning methods including

SVR and random forest. The shallow model was superior in accuracy, while the deep model was more profitable [71].

Chong, E. proposed DNN networks and examined the effects of three unsupervised feature extraction methods including principal component analysis, auto-encoder, and the restricted Boltzmann machine on the network's overall ability to predict future market behavior. The empirical results suggested that DNNs could extract additional information from the residuals of the auto-regressive model and could improve prediction performance; the same could not be said when the auto-regressive model is applied to the residuals of the network [72].

Singh, R. proposed a 2-Directional 2-Dimensional Principal Component Analysis (2D²PCA) + DNN, which outperformed the RNN and (2D²PCA) + RBFNN. The paper found that the best results were generated from a window size of 20 and a dimension of 10 × 10. The deep learning method for higher dimensions and large window sizes gave a limited performance [73].

Yu, P. and Yan, X. proposed a novel DNN model which incorporated LSTM as well as phase-space recognition. The model had produce promising results [74].

Yong, B.X., et al. proposed a DNN model with 40 nodes which showed reasonable results and appeared to be a highly profitable model [75].

4.1.5. Reinforcement Learning

Eight articles that used reinforcement learning technology for stock and Forex prediction are briefly described below. Table 6 shows the author, variables, dataset, and model of the papers mentioned.

Table 6. Information on the papers which used the reinforcement learning model.

| Reference No. | Author | Dataset | Variables | Model |
|---------------|--------------|---|--|---|
| [76] | Li, Y. | US stock dataset | Close price and volume | DQN |
| [77] | Shin, H.-G. | KOSPI | A candlestick chart, four moving average curves (5, 20, 60, and 120 days), a bar graph of trading volume, DMI, and SSO | Reinforcement Learning combined with LSTM and CNN |
| [78] | Jia, W. | Chinese stock codes: 002415, 600016, 600028, 600547, 600999, and 601988 | Open price, high price, low price, close price, volume, DEA, MACD EXPMA, CDP, TRIX, BBI, ASI, KDJ, RSI, PSY | Reinforcement Learning with LSTM-based agent |
| [79] | Carapuço, J. | EUR/USD | VR, ADX, CCI, WR, dm up, dm down | Reinforcement Learning |
| [80] | Kang, Q. | S&P500 index | bid/ask prices, and volumes | Reinforcement Learning with A3C algorithm |
| [81] | Zhu, Y. | S&P500 index | Open, low, high, close price, and trading volume | Reinforcement Learning with ABN |
| [82] | Si, W. | Stock-IF, stock-IC, and stock-IF | Open, low, high, close price, volume, MACD, MA6, MA12, RSI6, RSI12, and KD | Multi-objective deep reinforcement learning with LSTM |
| [83] | Pastore, A. | FTSE100 stock index | Close price | Reinforcement learning |
| | | | Date, type, stock, volume, price, and total | |

Li, Y. proposed three different reinforcement learning methods to predict the price of stock. The results showed that the best-performing deep reinforcement learning model is Deep-Q Network (DQN), not Double DQN. This paper also demonstrated Duelling DQN, which was an improved model based on DQN [76].

Shin, H.-G. proposed a reinforcement learning model combined with LSTM and CNN. The model generated various charts from stock trading data and used them as inputs to the CNN layer. The features extracted through the CNN layer were divided into column vectors and inputted to the LSTM layer. The reinforcement learning defined the agents' policy neural network structure, reward, and action and provided buying, selling, and holding probabilities as final output [77].

Jia, W. proposed a reinforcement learning with an LSTM-based agent that could automatically sense the dynamics of the stock market and could alleviate the difficulty of manually designing indicators from massive data. This paper compared a wide range of different input sets [78].

Carapuço, J. proposed a reinforcement learning-Q network model; three hidden layers of ReLU neurons were trained as Reinforcement Learning (RL) agents through the Q-learning algorithm under a novel simulated market environment framework. The framework was able to consistently induce stable learning that generalized to out-of-sample data [79].

Kang, Q. proposed to apply the state-of-art Asynchronous Advantage Actor-Critic algorithm (A3C algorithm) to solve the portfolio management problem and designed a standalone deep reinforcement learning model [80].

Zhu, Y. proposed an adaptive box-normalization (ABN) stock trading strategy based on reinforcement learning, which improved the original box theory. In his ABN strategy, the stock market data was independently normalized inside each oscillation box [81].

Si, W. proposed a reinforcement learning model which had multi-objective and LSTM agents. It was found that feature learning could contribute to better performances. The LSTM network made continuous decisions and could change positions at the right time, which reduced the transaction cost, and the multi-objective structure made good profits within the acceptable risk [82].

Pastore, A. analyzed the data for 46 players extracted from a financial market online game and tested whether reinforcement learning could capture these players’ behavior using a riskiness measure based on financial modelling [83].

4.1.6. Other Deep Learning Methods

Eight articles that used other deep learning technology for stock and Forex prediction are briefly described below. Table 7 shows the author, variables, dataset, and model of the papers mentioned.

Table 7. Information on the papers which used other deep learning methods.

| Reference No. | Author | Dataset | Variables | Model |
|---------------|--------------------|--|---|---|
| [84] | Long, W. | CSI300 | Open price, high price, low price, close price, and volume | MNFF |
| [85] | Wu, J.-L. | ANUE | Stock messages as information to form the text feature of each stock news (title, summary, and keywords) | HAN |
| [86] | Cho, C.-H. | CATHAY HOLDINGS, Fubon Financial, CTBC HOLDINGS, ESFH, and FFHC | Open price, high price, low price, close price, volume, MACD, CCI, ATR, BOLL, EMA12/20, MA5, MA1MOM6, MOM12, ROC, RSI, WVAD, exchange rate, and interest rate | Wavenet |
| [87] | Minh, D.L. | S&P 500, VN-index, and cophieu68; Bloomberg, Reuters | Open price, high price, low price, close price, volume, stochastic oscillator, William (%R), and RSI | Document preprocessing–document labeling–Stock2Vec embedding–BGRU |
| [88] | Hu, G. | Financial Times Stock; Exchange 100 Index (FTSE 100) | Processed news article | Convolutional AutoEncoder (CAE) |
| [89] | Hu, Z. | Chinese stock price; News (not given specific data) | Candlestick charts (images rather than annotation data) | |
| [90] | Kim, T. and Khushi | Nine Dow Jones companies representing each sector: industrials (MMM), financials (JPM), consumer services (PG), technology (AAPL), healthcare (UNH), consumer goods (WMT), oil and gas (XOM), basic materials (DD), and telecommunications (VZ) From Yahoo | 1. Close price and volume 2. News corpus sequence | HAN with SPL |
| [91] | Zhang, and Khushi | Forex exchange rates | Open price, high price, low price, close price, and volume | 2D Relative-Attentional Gated Transformer |
| [92] | Shi, Lei, et al. | Apple Inc. and S&P 500 | Trend indicators: moving average, exponential moving average, double exponential moving average, triple exponential moving average, and vortex indicators Momentum indicators: relative strength index, and stochastic oscillators Volatility indicators: Bollinger bands and Ichimoku indicators | Genetic Algorithm |
| [92] | Shi, Lei, et al. | Apple Inc. and S&P 500 | News and financial data | Hybrid of RNN, LSTM, and CNN |

Long, W. proposed a novel end-to-end model named multi-filters neural network (MFNN) specifically for feature extraction on financial time-series samples and price movement prediction task. Both convolutional and recurrent neurons were integrated to build the multi-filters structure. The results showed that the proposed model is better than the RNN, CNN, and traditional machine learning methods [84].

Wu, J.-L. proposed a keyword-based attention network into Hierarchical Attention Networks (HAN), namely the HKAN model, to learn the relationships between dimensional sentiments (trend and trading) and stock messages and outperformed the HAN network [85].

Cho, C.-H. proposed three different models to predict the price of the stock: LSTM, Seq2seq, and Wavenet. According to his experiments, Wavenet outperformed the other two models [86].

Minh, D.L. proposed a network to predict the directions of stock prices by using both financial news and sentiment dictionary. His results showed that Two-stram gated recurrent unit (TGRU) achieved better performance than GRU and LSTM and that Stock2Vec is more efficient in dealing with financial datasets [87].

Hu, G. proposed a Convolutional AutoEncoder model to learn a stock representation and converted a 4-channel stock time series (lowest, highest, open, and close price for the day) to candlestick charts by using the synthesis technique to present price history as images. This method successfully avoided expensive annotation. The proposed model outperformed the FTSE 100 index and many well-known funds in terms of total return [88].

Hu, Z. proposed a Hybrid Attention Networks (HAN) to predict the stock trend based on the sequence of recent related news. Moreover, he applied the self-paced learning (SPL) mechanism to achieve effective and efficient learning. The results showed that the proposed model outperformed RNN, Temporal-Attention-RNN, News-Attention-RNN, and HAN [89].

Kim, T. proposed a 2D relative-attentional gated transformer which was used to optimize portfolio return. It used general reinforcement learning with the agent incorporating 2D relative-attentional gated transformer [90].

Zhang, Z. proposed a genetic algorithm using a crossover of technique indicators as input. It successfully outperformed some of the traditional trading strategies [91].

Shi, L., et al. proposed a system that performed factor analysis and then utilized multiple deep learning methods to design a model which outperformed the benchmark [92].

4.2. Papers Results Grouped by the Method Used

4.2.1. CNN

Since the performance metrics used in different articles were different, our survey analyzed them based on various metrics. The results used the average performance of the best performing models in the mentioned papers. Table 8 shows the metrics and results for the papers which used the CNN model.

4.2.2. RNN

Since the performance metrics used in different articles were different, our survey analyzed them based on different metrics. The results used the average performance of the best performing models in the mentioned papers. Table 9 shows the metrics and results for the papers which used the RNN model.

4.2.3. LSTM

Since the performance metrics used were different in different articles, our survey analyzed them based on different metrics. The results used the average performance of the best performing models in the mentioned papers. Table 10 shows the metrics and results for the papers which used the LSTM model.

Table 8. The metrics and results for the papers which used the CNN model.

| Performance Metrics | Reference No. | Corresponding Value | Performance Metrics | Reference No. | Corresponding Value |
|---------------------|---------------------------|---------------------------|---------------------|-------------------------------|------------------------------|
| RMSE | [11] | 0.043 +/- 0.007 | MAPE | [12] | 5 |
| | [12] | 11 | Sharpe ratio | [14] | [14] 2D:0.1422, 3D:0.1413 |
| MAE | [24] | 0.395185×10^{-3} | | [18] | [18] 0.611 |
| | [12] | 6 | CEQ | [14] | 2D:0.0006681 3D:0.000664 |
| [24,31] | 0.240977×10^{-3} | [14] | | [14] (2D:1.2312 3D:1.2604) | |
| Accuracy | [10,13] | 71% | Return rate | [16] | [16] (1.3107) |
| | [16] | 60.02% | | | |
| | [19] | 55.44% | | | |
| | [22] | 71.72% | | | |
| | [23] | 65.08% | | | |
| | [3] | 75.2% | | | |
| | [25] | 57.88% | | | |
| Error Percentage | [26] | 74.753% | Mean Test Score | [18] | [18] (1.2156) |
| | [20] | 95.02% | | [21] | [21] (1.309) |
| | [14] | 2D:0.4944 3D:0.4931 | | [15] | 0.000281317 |
| F-measure | [18] | 0.6227 | MSE | [22] | 0.2631 |
| | [19] | 0.7133 | AE | [24,31] | 0.156×10^{-6} |
| | [3] | 0.73 | | [11] | 0.029 +/- -0.005 |
| | [26] | 0.6367 | | | |

Table 9. The metrics and results for the papers which used the RNN model.

| Performance Metrics | Reference No. | Corresponding Value |
|---------------------|---------------|---------------------|
| RMSE | [27] | 512–530 |
| | [29] | 0.0205 |
| | [31] | 0.00165 |
| MAPE | [29] | 0.2431 |
| | [31] | 0.232 |
| MAE | [29] | 0.0132 |
| Accuracy | [28] | 68.95% |
| | [29] | 66.54% |
| F-measure | [28] | 0.7658 |
| Recall | [28] | 0.7471 |
| Precision | [28] | 0.7855 |
| MSE | [30] | 0.057443 |

Table 10. The metrics and results for the papers which used the LSTM model.

| Performance Metrics | Reference No. | Corresponding Value | Performance Metrics | Reference No. | Corresponding Value |
|---------------------|---------------|---------------------|---------------------|---------------|---------------------|
| RMSE | [32] | 0.306543 | MAPE | [35] | 1.03 |
| | [35] | 347.46 | | [38] | 1.6905 |
| | [36] | 0.0151 | | [40] | 4.13 |
| | [37] | 25.90 | | [41] | 1.0077 |
| | | | | [57] | 0.119 |
| | | | | [60] | 0.91 |
| | | | | [10] | 1.37 |
| | | | | [62] | 1.65 |
| | | | | [66] | 0.6346 |
| | | | | [53] | 0.553 |
| MAE | [38] | 0.0242 | Precision | [53] | 0.129 |
| | [39] | 1.3 | Recall | [33] | 1.0667 |
| | [45] | 9.72 | Return rate | [32] | 0.093969 |
| | [46] | 4.24 (Average) | | [33] | 0.004845492 |
| | [47][57] | 1–100.0015 | MSE | [59] | 0.000379 |
| | [65] | 0.02295 | | [35] | 120731.4 |
| | [32] | 0.21035 | | [38] | 19.7096 |
| [35] | 262.42 | [39] | | 0.019 | |
| [37] | 0.1895 | [40] | | 0.00098 | |
| [38] | 0.0169 | [41] | | 7.56 | |
| | 0.023 | [42] | 0.00149 | | |
| | 0.01069 | [43] | 1.012 | | |
| | 1.975 | | | | |

Table 10. Cont.

| Performance Metrics | Reference No. | Corresponding Value | Performance Metrics | Reference No. | Corresponding Value |
|---------------------|---------------|---------------------|---------------------|---------------|---------------------|
| Accuracy | [34] | 54.58% | MCC | [34] | 0.0478 |
| | [36] | 98.49% | R2 | [35] | 0.83 |
| | [45] | 60.60% | HMAE | [42] | 0.42911 |
| | [50] | 65% | HMSE | [42] | 0.23492 |
| | [52] | 83.91% | IC | [48] | 0.1259 |
| | [53] | 55.90% | AR | [48] | 0.2015 |
| | [54] | 63.34% | IR | [48] | 3.0521 |
| | [56] | 27.20% | | | |
| | [58] | 53.2% | | | |
| | [61] | 87.86% | | | |
| F-measure | [62] | 70.56% | Score | [55] | 0.4271 |
| | [63] | 75.89% | | | |
| | [64] | 75.58% | | | |
| | [53] | 0.209 | | | |

4.2.4. DNN

Since the performance metrics used in different articles were different, our survey analyzed them based on different metrics. The results used the average performance of the best performing models in the mentioned papers. Table 11 shows the metrics and results for the papers which used the DNN model.

Table 11. The metrics and results for the papers which used the DNN model.

| Performance Metrics | Reference No. | Corresponding Value | Performance Metrics | Reference No. | Corresponding Value |
|---------------------|---------------|---------------------|---------------------|---------------|---------------------|
| RMSE | [71] | 0.0951 | Sharpe ratio | [71] | 1.41 |
| | [72] | 0.8214 | | [75] | 5.34 |
| | [73] | 0.00674 | Return rate | [70] | 1.0952 |
| MAE | [71] | 0.0663 | CORR | [71] | 1.1081 |
| | [72] | 0.5852 | MSE | [70] | 0.0582 |
| Accuracy | [67] | 61.90% | MSE | [70] | 0.0836 |
| | [68] | 84.50% | SMAPE | [72] | 0.9621 |
| | [74] | 58.07% | | [73] | 0.0696 |
| F-measure | [68] | 0.824 | MAPE | [73] | 0.080059 |
| | | | | [75] | 1.84 |
| | | | Volatility | [71] | 7.65% |

4.2.5. Reinforcement Learning

Since the performance metrics used were different in different articles, our survey analyzed them based on different metrics. The results used the average performance of the best performing models in the mentioned papers. Table 12 shows the metrics and results for the papers which used the reinforcement learning model.

Table 12. The metrics and results for the papers which used the reinforcement learning model.

| Performance Metrics | Reference No. | Corresponding Value |
|---------------------|---------------|---------------------|
| Sharpe ratio | [77] | 2.77 |
| | [82] | 0.12 |
| | [78] | 1.948 |
| Return rate | [79] | 1.163 ± 2.8% |
| | [81] | 2.442 |
| MSE | [81] | 0.000412 |

4.2.6. Other Deep Learning Methods

Since the performance metrics used in different articles were different, our survey analyzed them based on different metrics that were used. The results used the average performance of the best performing models in the mentioned papers. Table 13 shows the metrics and results for the papers which used other deep learning methods.

Table 13. The metrics and results for the papers which used other deep learning methods.

| Performance Metrics | Reference No. | Corresponding Value | |
|---------------------|---------------|------------------------------|------|
| RMSE | [86] | 0.6866 | |
| Accuracy | [87] | 66.32% | |
| | [89] | 47.8% | |
| | [92] | 79.7% | |
| | [84] | 4.49 | |
| Sharpe ratio | [88] | 0.8 | |
| | [90] | 0.6418 | |
| | [91] | 6.68 on EURUSD currency | |
| | [84] | 1.4228 | |
| | [87] | 1.0531 (0.25%) | |
| Return rate | [88] | 1.118 | |
| | [89] | 1.611 (0.3%) | |
| | [90] | 1.4316 | |
| | [91] | 1.0968 on EURUSD currency | |
| | MSE | [85] | 1.05 |
| | MDAE | [85] | 0.71 |
| Correlation | [86] | 0.9564 | |
| Precision | [87] | 72.1% | |
| Recall | [87] | 77.32% | |

5. Discussion

5.1. Analysis Based on the Method Used

5.1.1. CNN

Some findings can be drawn from reviewing the CNN models:

1. According to the datasets used, 6 papers used a combination of technical analysis, and sentiment and news analysis to predict the stock. The rest of them used the method of technical analysis only.
2. For the variables, the closing price was the choice of all CNN models, and five papers used closing price only.
3. It could be found that 12 of the papers changed the traditional CNN model to pursue higher performances in prediction. The combination of CNN and LSTM was the most common model.
4. The metrics used in each paper were different; there were 11 metrics used for measuring the performance of the CNN model.
5. Multiple articles selected RMSE, return rate, F-measure, Sharpe ratio, and accuracy. We found that paper [22] had the highest accuracy compared to any other papers, paper [16] had the highest return rate followed by papers [14,18,21]. Paper [24,31] had the lowest RMSE. Paper [19] had a higher F-measure than that of papers [14,18], but paper [18] achieved a higher Sharpe ratio than that of paper [14].

5.1.2. RNN

In the section on the RNN-based model, the following conclusions could be drawn:

1. According to the datasets used, there was 1 paper that used a combination of technical analysis, and sentiment and news analysis to predict the stock. The rest used technical analysis only.
2. For the variables, all the RNN-based models used a multivariable input, and open price, close price, highest price, and the lowest price were used in all models as an input.
3. It could be found that all the papers changed the traditional RNN model to pursue a higher performance in prediction. Two of the papers chose the C-RNN based model.

4. The metrics used in each paper were different; in total, there were 8 metrics used for measuring the performance of the RNN-based model.
5. RMSE and accuracy were selected by multiple articles. We note that the paper in [29] has a much lower RMSE than that of paper [27], and paper [28] had higher accuracy than that of paper [29].

5.1.3. LSTM

The following points were worth discussing from reviewing the LSTM papers:

1. According to the datasets used, 3 papers used a combination of technical analysis, and sentiment and news analysis to predict the stock. The rest used technical analysis, only with the exception of one paper which used expert recommendations.
2. For the variables, the closing price was the choice of 23 LSTM-based models, there were 8 papers that used closing price only, and 12 papers included close price, open price, high price, and low price in their input.
3. It could be found that 15 of the papers changed the traditional LSTM model to pursue a higher performance in prediction. Attention-based LSTM and LSTM with RNN were the most frequent models, showing up in three different papers. Two papers chose the method of LSTM with GRU to improve the model.
4. The metrics used in each paper were different; there were 17 metrics used for measuring the performance of the LSTM based model.
5. Multiple articles selected RMSE, MAPE, MAE, accuracy, and MSE. For RMSE, we found that paper [36], paper [65], and paper [38] were in the lowest order of magnitude, with paper [36] achieving the lowest; paper [32] was in the second-lowest order of magnitude; and papers [39,45–47] were in the third lowest order of magnitude. Paper [37] was in the fourth lowest order of magnitude, and paper [35] had the highest order of magnitude.
6. For MAPE, paper [57] had the lowest order of magnitude followed by papers [10,35,38,41,60,62,66]. Paper [40] had the highest order of magnitude for MAPE.
7. For MAE, papers [38,40,42,57] had the lowest order of magnitude, with paper [57] having the best performance; papers [32,37] were in the second-lowest order of magnitude, while paper [41] was in the third lowest order of magnitude. Paper [35] had the highest order of magnitude.
8. For MSE, paper [59] had the lowest MSE and was in the lowest order of magnitude; papers [33,40,42] were in the second-lowest order of magnitude, while papers [32,39] were in the third lowest order of magnitude. Papers [41,43] were in the fourth lowest order of magnitude, paper [38] was in the fifth-lowest order of magnitude, and paper [35] had the highest order of magnitude.
9. For accuracy, paper [36] had the highest accuracy, followed by [34,45,50,52–54,61], and paper [56] had the lowest accuracy.

5.1.4. DNN

In the section of DNN-based model, the following conclusions could be drawn:

1. According to the datasets used, no paper used a combination of technical analysis, and sentiment and news analysis to predict the stock. All used the method of technical analysis only.
2. For the variables, six of the seven DNN-based models used multivariable input, and only one paper used closing price as its sole input.
3. It could be found that 3 of the papers changed the traditional DNN model to pursue a higher performance in prediction. All of the improved models were not duplicated.
4. Because the metrics used in each paper were different, there were 11 different metrics used for measuring the performance of the DNN-based model.
5. Multiple articles selected RMSE, MAE, accuracy, return rate, and MSE. We found that paper [73] had the lowest RMSE followed by papers [71,72], with the latter two in different orders of magnitude.

6. As for MAE, paper [71] had a lower MAE than paper [72], and the accuracy of paper [68] was higher than that of papers [67] and [74]. Furthermore, paper [71] had a higher return rate than paper [70]. Paper [70] had a lower MSE than paper [72].

5.1.5. Reinforcement Learning

In the section of reinforcement learning-based model, the following conclusions could be drawn:

1. According to the datasets used, no paper used a combination of technical analysis, and sentiment and news analysis to predict the stock. All of them used technical analysis only.
2. For the variables, 7 of the reinforcement learning-based models used the multivariable input, and only one paper solely used closing price as the input.
3. It could be found that three of the 6 papers changed the traditional reinforcement learning model to pursue a higher performance in prediction. Three of the improved models were combined with LSTM.
4. The metrics used in each paper were different; in total, there were 3 metrics used in the measurement performance of the reinforcement learning-based model.
5. Multiple articles selected Sharpe ratio and return rate; we found that paper [77] had a much higher Sharpe ratio than paper [82]; paper [81] had a higher return rate than paper [78], followed by paper [79].

5.1.6. Other Deep Learning Methods

In the section of other deep learning method-based models, the following conclusions could be drawn:

1. According to the datasets used, 4 papers made use of sentiment and news analysis to predict the stock. The rest of them used the method of technical analysis only.
2. For the variables, five of the other deep learning methods models used the multivariable input and only one paper used candlestick charts alone as input.
3. It could be found that there were 5 different models in the other deep learning methods. The only model that appeared three times was HAN, which consisted of an ordinary HAN model and two modified HAN models. The rest of the models were not duplicated.
4. Because the metrics used in each paper were different, there were 9 metrics used in the measurement performance of this section.
5. Multiple articles selected accuracy, Sharpe ratio, and return rate; we found that paper [92] had the highest accuracy. Paper [84] had a much higher Sharpe ratio than paper [88].
6. For return rate, paper [89] had the highest return rate followed by papers [84,88], and paper [87] had the lowest return rate.

5.2. Discussion and Analysis Based on Performance Metrics

In this part, all of the Forex/stock price prediction models mentioned above which made use of the deep learning are discussed and analyzed. All analyses of the performance metric data are shown in Table 14. It could be found that the most commonly used performance metrics were RMSE, MAPE, MAE, MSE, accuracy, Sharpe ratio, and return rate. Hence, our review analyzes these papers in terms of the performance metrics used.

Table 14. Analysis based on performance metrics.

| Performance Metrics | Reference No. |
|---------------------|---|
| RMSE | [8,12,24,27,29,31,32,35–39,45–47,71–73,86] |
| MAPE | [10,12,29,31,35,38,40,41,60,65,73] |
| MAE | [12,24,29,32,35,37,38,40–42,66,71,72] |
| Accuracy | [3,13,16,19,22,23,25,26,28,29,34,36,45,50,52–54,56,58,61–64,67,68,74,87,89] |
| F-measure | [3,14,18,19,26,28,53,68] |
| Sharpe ratio | [14,18,71,77,82,84,88,90,91] |
| CEQ | [14] |
| Return rate | [14,16,18,21,23,33,70,71,78,79,84,87–91] |
| Mean Test Score | [15] |
| MSE | [22,24,30–33,35,38–43,59,70,72,81] |
| AE | [11] |
| Precision | [28,53] |
| Recall | [28,53] |
| R2 | [35] |
| Error Percentage | [20] |
| MCC | [34] |
| HMAE | [42] |
| HMSE | [42] |
| CORR | [70] |
| SMAPE | [73] |
| Volatility | [71] |
| IC | [48] |
| AR | [48] |
| IR | [48] |
| Score | [55] |

5.2.1. Analysis Based on RMSE

In this part, the values of the RMSE in the papers are discussed. Table 15 shows the details of each paper’s RMSE based on the numerical range. In this table, papers [11,12,24] used the CNN-based model; papers [32,35–39,45–47,57,65] used the LSTM-based model; papers [71–73] used the DNN-based model; papers [27,29,31] used the RNN-based model; and paper [86] used another deep learning method.

Table 15. Analysis based on Root Mean Square Error (RMSE).

| RMSE Range | Reference No. |
|------------|---|
| RMSE | [11,12,24,27,29,32,35–39,45–47,65,71–73,86] |
| <0.001 | [24,57] |
| 0.001–0.01 | [31,73] |
| 0.01–0.1 | [11,29,32,36,38,65,71] |
| 0.1–1 | [12,72,86] |
| 1–10 | [39,45–47] |
| 10–100 | [37] |
| >100 | [27,35] |

It was clear that papers [24,57] achieved the best performance using the DNN model, having RMSEs smaller than 0.001. Papers [11,29,31,32,36,38,71,73] had great performance, while papers [27,35,37] did not have low RMSEs.

Among all the papers, 33% of the CNN papers had an RMSE below 0.001, 33% of the CNN papers were in the range of 0.01–0.1, and 33% of the CNN papers were in the range of 0.1–1. 36% of the LSTM papers were in the ranges 0.01–0.1 and 1–10 each, and the rest

were distributed equally in ranges <0.001 , 10–100, and above 100, with 9% in each range; and 33.3% of the DNN papers were in the ranges 0.001–0.01, 0.01–0.1, and 0.1–1 each. The RNN papers were distributed equally in the ranges of 0.001–0.01, 0.01–0.1, and above 100, i.e., 33% in each range. The only paper that used another deep learning method had an RMSE in the range of 0.01–0.1.

5.2.2. Analysis Based on MAPE

In this part, the values of the MAPE are discussed. Table 16 shows the papers' MAPE performances based on numerical range. In this table, Paper [12] used a CNN-based model; papers [10,35,38,40,41,57,60,66] used an LSTM-based model; papers [73,75] used a DNN-based model; and paper [29] used an RNN-based model.

Table 16. Analysis based on Mean Absolute Percentage Error (MAPE).

| MAPE Range | Reference No. |
|------------|---------------------------------------|
| MAPE | [10,12,29,35,38,40,41,57,60,66,73,75] |
| 0–0.5 | [29,57,66,73] |
| 0.5–1 | [60] |
| 1–1.5 | [10,35,41,75] |
| 1.5–2 | [38] |
| 2–10 | [12,40] |

It could be found that papers [29,57,73] performed best, using the DNN model, the RNN model, and LSTM, respectively, and they were in the range of 1–1.5. Papers [35,41,60] had great performances, while papers [12,40] did not have low MAPEs.

Among the papers, the only CNN paper was in the range of 2–10; 37.5% of the LSTM papers were in the range of 1–1.5; 12.5% of the LSTM papers were in the ranges 0.5–1, 1.5–2, and 2–10 each; and 25% of LSTM papers were in the range 0–0.5. The two DNN papers were in the ranges 0–0.5 and 1–1.5. Lastly, the only RNN paper was in the range 0–0.5.

5.2.3. Analysis Based on MAE

In this part, the values of MAE are discussed. Table 17 shows the details of the papers' MAE. In this table, papers [12,24] used the CNN-based model; papers [34,37,39,40,42–44] used the LSTM-based model; papers [71,72] used the DNN-based model; and papers [29,31] used the RNN-based model.

Table 17. Analysis based on Mean Absolute Error MAE.

| MAE Range | Reference No. |
|-----------|------------------------------------|
| MAE | [12,24,29,32,35,37,38,40–42,71,72] |
| <0.01 | [24] |
| 0.01–0.1 | [29,38,40,42,71] |
| 0.1–1 | [31,32,37,72] |
| 1–10 | [12,41] |
| 10–100 | N/A |
| >100 | [35] |

It could be found that paper [24] performed best, which used the CNN model. The papers' MAEs were in the range of 0.1–1. Papers [29,38,40,42,71] also had good performance; however, paper [35] did not have a low MAE.

Among the papers that used MAE as a measurement, 50% of CNN papers were in the range 1–10; the other 50% had MAEs smaller than 0.01; 42.9% of the LSTM papers were in the range 0.01–0.1; 28.6% of the LSTM papers were in the range 0.1–1; 14.2% of the LSTM papers were in the ranges 1–10 and above 100 each; and 50% of the DNN papers were in

the ranges 0.01–0.1 and 0.1–1 each. The two RNN papers were in the ranges of 0.01–0.1 and 0.1–1.

5.2.4. Analysis Based on MSE

In this part, the values of the MSE are discussed. Table 18 shows the details of the papers' MSE. In this table, papers [22,24] used a CNN-based model; papers [32,33,35,38–43,59] used an LSTM-based model; and papers [70,72] used a DNN-based model. Paper [30] used an RNN-based model, and paper [81] used the reinforcement learning based model.

Table 18. Analysis based on Mean Square Error (MSE).

| MSE Range | Reference No. |
|-----------|---------------------------------------|
| MSE | [22,24,30,32,33,35,38–43,59,70,72,81] |
| <0.01 | [59] |
| 0–0.01 | [33,40,42,81] |
| 0.01–0.1 | [30,32,39,70] |
| 0.1–1 | [22,43,72] |
| 1–10 | [41] |
| 10–100 | [38] |
| >100 | [35] |

It could be found that paper [59] performed the best, which used the LSTM model; its MSE was smaller than 0.01. Papers [24,33,40,42,81] also had good performance; however, papers [35,38] did not have low MSEs.

Among the papers that used MSE as a performance measurement, the only CNN paper was in the range of 0.1–1; 40% of the LSTM papers were in the range of 0–0.01; 20% of the LSTM papers were in the range of 0.01–0.1; 10% of the LSTM papers were in the ranges of 0.1–1, 1–10, 10–100, and above 100 each; and 50% of the DNN papers were in the ranges of 0.01–0.1 and 0.1–1 each. The only RNN-based model and reinforcement learning-based model were in the ranges 0–0.01 and 0.01–0.1 respectively.

5.2.5. Analysis Based on the Accuracy

In this part, the values of accuracy are discussed. Table 19 shows the details of the papers' accuracy. In this table, papers [3,13,16,19,22,23,25,26] used a CNN-based model; papers [34,36,45,50,52–54,56,58,61–64] used an LSTM based model; papers [67,68,74] used a DNN-based model; papers [28,29] used an RNN-based model; and papers [87,89,92] used other deep learning methods.

Table 19. Analysis based on the accuracy.

| Accuracy Range | Reference No. |
|----------------|--|
| Accuracy | [3,13,16,19,22,23,25,26,28,29,34,36,45,50,52–54,56–58,61–64,67,68,74,87,89,92] |
| 0–50% | [56,89] |
| 50–60% | [19,25,34,53,58,74] |
| 60–70% | [16,23,28,29,45,50,54,67,87] |
| 70–80% | [3,13,22,26,62–64,92] |
| 80–90% | [52,61,68] |
| 90–100% | [36] |

It could be found that paper [36] performed best, which used an LSTM model, and the paper's accuracy was in the range of 90–100%. Papers [52,61,68] also had great performances, but papers [56,89] did not have high accuracy.

Among the papers that had an accuracy measure metric, 23% of the CNN papers were in the range of 50–60%, 23% of the CNN papers were in the range of 60–70%, and 54% of the CNN papers were in the range of 70–80%; 8% of the LSTM papers were in the ranges of 0–50% and 90–100% each; 15% of the LSTM papers were in the range of 80–90%; 23% of the LSTM papers were in the ranges of 50–60%, 60–70%, and 70–80% each; and 33% of the DNN papers were in the ranges of 50–60%, 60–70%, and 80–90% each. All the RNN papers were in the ranges of 60–70%, and 33% of other deep learning methods were in the ranges of 40–50%, 60–70%, and 70–80% each.

5.2.6. Analysis Based on Sharpe Ratio

In this part, the values of the Sharpe ratio are discussed. The Sharpe ratio refers to the amount of return generated relative to the risk taken, where risk is calculated using the standard deviation of return. Table 20 shows the details of the papers that used the Sharpe ratio as a performance metric. In this table, papers [14,18] used a CNN-based model; papers [71,75] used a DNN-based model; papers [77] and [82] used a reinforcement learning-based model; and papers [84,88,90,91] used other deep learning methods.

Table 20. Analysis based on Sharpe ratio.

| Sharpe Ratio Range | Reference No. |
|--------------------|------------------------------|
| Sharpe ratio | [14,18,71,77,82,84,88,90,91] |
| 0.1–1 | [14,18,82,88,90] |
| 1–2 | [71] |
| 2–5 | [77] |
| 5–10 | [75,84,91] |

It could be found that papers [75,84,91] performed best, which used a DNN model, and Sharpe ratio was in the range of 5–10. Paper [77] also had a great performance, but papers [14,18,82,88,90] did not have high Sharpe ratios.

Among the paper that used the Sharpe ratio, all CNN papers were in the range of 0.1–1. The two DNN papers were in the ranges 1–2 and 5–10; 50% of the reinforcement learning papers were in the range of 0.1–1; 50% of the reinforcement learning papers were in the range of 2–5; 50% of the other deep learning method papers were in the range of 0.1–1; and 50% of the other deep learning method papers were in the range of 5–10.

5.2.7. Analysis Based on the Return Rate

In this part, the values of the return rate are discussed. Table 21 shows the details of the papers’ return rates. In this table, papers [14,16,18,21] used the CNN-based model; paper [33] used the LSTM-based model; papers [70,71] used the DNN-based model; papers [78,79] used the reinforcement learning based model; and papers [84,87–91] used other deep learning methods.

Table 21. Analysis based on the return rate.

| Return Rate Range | Reference No. |
|-------------------|---------------------------------------|
| Return rate | [14,16,18,21,33,70,71,78,79,84,87,89] |
| 1.0–1.2 | [33,70,71,79,87,88,91] |
| 1.2–1.4 | [14,16,18,21] |
| 1.4–1.6 | [84,90] |
| 1.6–1.8 | [89] |
| 1.8–2.0 | [78] |

It could be found that paper [78] performed the best, which used the reinforcement learning model, and the paper's return rate was in the range of 1.4–1.8. Papers [84,89] had good performance, but papers [33,70,71,79,87,88,91] did not have high return rates.

Among the papers that used return rate as a performance measurement, all the CNN papers were in the range of 1.2–1.4, the only LSTM paper was in the range of 1.0–1.2, all the DNN papers were in the range of 1.0–1.2, 50% of the reinforcement learning papers were in the range of 1.0–1.2, and 50% of the reinforcement learning papers were in the range of 1.8–2. The other deep learning method papers performed well, with 17% of papers in the range of 1.0–1.2, 33% of papers in the range of 1.4–1.6, and 50% of papers in the range of 1.6–1.8.

6. Conclusions

This paper provided a detailed review of 86 papers from 2015 to the present on predicting stock/Forex price movements through deep learning methods. The existing stock/Forex models were evaluated by analyzing data sets, variables, the use of different models, and the metrics of evaluation. The research review included a wide range of techniques: CNN; LSTM; DNN; RNN; reinforcement learning [93]; and other deep learning methods such as HAN, NLP, and Wavenet. Furthermore, the data sets, variables, and models and their different results were analyzed and compared within each technique. Then, our paper discusses the main performance metrics of all models. They are RMSE, MAPE, MAE, MSE, accuracy, Sharpe ratio, and return rate.

This paper aimed to contribute to the research on stock/Forex market prediction through an analysis of the above different deep learning prediction models. Through the review, it can be identified that there is a lack of studies on the combination of multiple deep learning methods, especially with respect to other deep learning methods. The hybrid networks are showing promising signs for future research. In the future, we will design a specific hybrid model based on the above analysis, incorporating the latest technologies such as advanced genetic algorithms and self-attention neural networks to predict the stock/Forex market.

Author Contributions: First two authors contributed equally; M.K. conceived and designed the study, supervised the work, and revised the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tsai, C.-F.; Hsiao, Y.-C. Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches. *Decis. Support Syst.* **2010**, *50*, 258–269. [[CrossRef](#)]
2. Khadjeh Nassirtoussi, A.; Aghabozorgi, S.; Wah, T.Y.; Ngo, D. Text mining of news-headlines for FOREX market prediction: A multi-layer dimension reduction algorithm with semantics and sentiment. *Expert Syst. Appl.* **2015**, *42*, 306–324. [[CrossRef](#)]
3. Chen, S.; He, H. Stock prediction using convolutional neural network. In *2018 2nd International Conference on Artificial Intelligence Applications and Technologies (AIAAT 2018)*; IOP Publishing: Shanghai, China, 2018.
4. Akita, R.; Yoshihara, A.; Matsubara, T.; Uehara, K. Deep learning for stock prediction using numerical and textual information. In *Proceedings of the 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, Okayama, Japan, 26–29 June 2016.
5. Sirignano, J.; Cont, R. Universal features of price formation in financial markets: Perspectives from deep learning. *Quant. Financ.* **2019**, *19*, 1449–1459. [[CrossRef](#)]
6. Hu, B.; Lu, Z.; Li, H.; Cheng, Q. Convolutional neural network architectures for matching natural language sentences. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 2042–2050.

7. Sundermeyer, M.; Schlüter, R.; Ney, H. LSTM neural networks for language modeling. In Proceedings of the Thirteenth Annual Conference of the International Speech Communication Association, Portland, OR, USA, 9–13 September 2012.
8. He, T.; Droppo, J. Exploiting LSTM structure in deep neural networks for speech recognition. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 5445–5449.
9. Souma, W.; Vodenska, I.; Aoyama, H. Enhanced news sentiment analysis using deep learning methods. *J. Comput. Soc. Sci.* **2019**, *2*, 33–46. [[CrossRef](#)]
10. Zhang, K.; Zhong, G.; Dong, J.; Wang, S.; Wang, Y. Stock market prediction based on generative adversarial network. *Procedia Comput. Sci.* **2019**, *147*, 400–406. [[CrossRef](#)]
11. Maqsood, H.; Mehmood, I.; Maqsood, M.; Yasir, M.; Afzal, S.; Aadil, F.; Selim, M.M.; Muhammad, K. A local and global event sentiment based efficient stock exchange forecasting using deep learning. *Int. J. Inf. Manag.* **2020**, *50*, 432–451. [[CrossRef](#)]
12. Patil, P.; Wu, C.-S.; Potika, K.; Orang, M. Stock market prediction using ensemble of graph theory, machine learning and deep learning models. In Proceedings of the 3rd International Conference on Software Engineering and Information Management, Sydney, Australia, 12–15 January 2020.
13. Sim, H.S.; Kim, H.I.; Ahn, J.J. Is deep learning for image recognition applicable to stock market prediction? *Complexity* **2019**, 1–10. [[CrossRef](#)]
14. Hoseinzade, E.; Haratizadeh, S. CNNpred: CNN-based stock market prediction using a diverse set of variables. *Expert Syst. Appl.* **2019**, *129*, 273–285. [[CrossRef](#)]
15. Eapen, J.; Bein, D.; Verma, A. Novel deep learning model with CNN and Bi-directional LSTM for improved stock market index prediction. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019.
16. Yang, H.; Zhu, Y.; Huang, Q. A multi-indicator feature selection for CNN-driven stock index prediction. In Proceedings of the International Conference on Neural Information Processing, Siem Reap, Cambodia, 13–16 December 2018; Springer: Berlin/Heidelberg, Germany, 2018.
17. Cai, S.; Feng, X.; Deng, Z.; Ming, Z.; Shan, Z. Financial news quantization and stock market forecast research based on CNN and LSTM. In Proceedings of the International Conference on Smart Computing and Communication, Tokyo, Japan, 10–12 December 2018; Springer: Berlin/Heidelberg, Germany, 2018.
18. Oncharoen, P.; Vateekul, P. Deep Learning Using Risk-Reward Function for Stock Market Prediction. In Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence, Shenzhen, China, 8–10 December 2018; Association for Computing Machinery: Shenzhen, China, 2018; pp. 556–561.
19. Liu, Y.; Zeng, Q.; Yang, H.; Carrio, A. Stock price movement prediction from financial news with deep learning and knowledge graph embedding. In Proceedings of the Pacific Rim Knowledge Acquisition Workshop, Nanjing, China, 27–28 August 2018; Springer: Berlin/Heidelberg, Germany, 2018.
20. Selvin, S.; Vinayakumar, R.; Gopalakrishnan, E.A.; Menon, V.K.; Soman, K.P. Stock price prediction using LSTM, RNN and CNN-sliding window model. In Proceedings of the 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Manipal, India, 13–16 September 2017.
21. Liu, S.; Zhang, C.; Ma, J. CNN-LSTM neural network model for quantitative strategy analysis in stock markets. In Proceedings of the International Conference on Neural Information Processing, Guangzhou, China, 14–18 November 2017; Springer: Berlin/Heidelberg, Germany, 2017.
22. Gudelek, M.U.; Boluk, S.A.; Ozbayoglu, A.M. A deep learning based stock trading model with 2-D CNN trend detection. In Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA, 27 November–1 December 2017; pp. 1–8.
23. Ding, X.; Zhang, Y.; Liu, T.; Duan, J. Deep learning for event-driven stock prediction. In Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015.
24. Zhao, Y.; Khushi, M. Wavelet denoised-ResNet CNN and LightGBM method to predict forex rate of change. In Proceedings of the 2020 IEEE International Conference on Data Mining Workshops (ICDMW), Sorrento, Italy, 11–17 November 2020.
25. Chen, J.-F.; Chen, W.-L.; Huang, C.-P.; Huang, S.-H.; Chen, A.-P. Financial time-series data analysis using deep convolutional neural networks. In Proceedings of the 2016 7th International Conference on Cloud Computing and Big Data (CCBD), Macau, China, 16–18 November 2016; pp. 87–92.
26. Wen, M.; Li, P.; Zhang, L.; Chen, Y. Stock market trend prediction using high-order information of time series. *IEEE Access* **2019**, *7*, 28299–28308. [[CrossRef](#)]
27. Ni, L.; Li, Y.; Wang, X.; Zhang, J.; Yu, J.; Qi, C. Forecasting of forex time series data based on deep learning. *Procedia Comput. Sci.* **2019**, *147*, 647–652. [[CrossRef](#)]
28. Li, C.; Song, D.; Tao, D. Multi-task recurrent neural networks and higher-order markov random fields for stock price movement prediction: Multi-task RNN and higher-order MRFs for stock price classification. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; Association for Computing Machinery: Anchorage, AK, USA, 2019; pp. 1141–1151.
29. Chen, W.; Yeo, C.K.; Lau, C.T.; Lee, B.S. Leveraging social media news to predict stock index movement using RNN-boost. *Data Knowl. Eng.* **2018**, *118*, 14–24. [[CrossRef](#)]

30. Zhang, R.; Yuan, Z.; Shao, X. A new combined CNN-RNN model for sector stock price analysis. In Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, Japan, 23–27 July 2018.
31. Zeng, Z.; Khushi, M. Wavelet denoising and attention-based RNN- ARIMA model to predict forex price. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020.
32. Nikou, M.; Mansourfar, G.; Bagherzadeh, J. Stock price prediction using DEEP learning algorithm and its comparison with machine learning algorithms. *Intell. Syst. Account. Financ. Manag.* **2019**, *26*, 164–174. [[CrossRef](#)]
33. Fazeli, A.; Houghten, S. Deep learning for the prediction of stock market trends. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019.
34. Xu, Y.; Keselj, V. Stock prediction using deep learning and sentiment analysis. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019.
35. Lakshminarayanan, S.K.; McCrae, J. A comparative study of svm and lstm deep learning algorithms for stock market prediction. In Proceedings of the 27th AIAI Irish Conference on Artificial Intelligence and Cognitive Science (AICS 2019), Galway, Ireland, 5–6 December 2019.
36. Rana, M.; Uddin, M.; Hoque, M. Effects of Activation Functions and Optimizers on Stock Price Prediction using LSTM Recurrent Networks. In Proceedings of the 2019 3rd International Conference on Computer Science and Artificial Intelligence, Beijing, China, 6–8 December 2019; Association for Computing Machinery: Normal, IL, USA, 2019; pp. 354–358.
37. Naik, N.; Mohan, B.R. Study of stock return predictions using recurrent neural networks with LSTM. In Proceedings of the International Conference on Engineering Applications of Neural Networks, Xersonisos, Greece, 24–26 May 2019; Springer: Berlin/Heidelberg, Germany, 2019.
38. Nguyen, D.H.D.; Tran, L.P.; Nguyen, V. Predicting stock prices using dynamic LSTM models. In Proceedings of the International Conference on Applied Informatics, Madrid, Spain, 7–9 November 2019; Springer: Berlin/Heidelberg, Germany, 2019.
39. Lai, C.Y.; Chen, R.-C.; Caraka, R.E. Prediction stock price based on different index factors using LSTM. In Proceedings of the 2019 International Conference on Machine Learning and Cybernetics (ICMLC), Kobe, Japan, 7–10 July 2019.
40. Hossain, M.A.; Karim, R.; Thulasiram, R.; Bruce, N.D.B.; Wang, Y. Hybrid deep learning model for stock price prediction. In Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence (SSCI), Bengaluru, India, 18–21 November 2018.
41. Baek, Y.; Kim, H.Y. ModAugNet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module and a prediction LSTM module. *Expert Syst. Appl.* **2018**, *113*, 457–480. [[CrossRef](#)]
42. Kim, H.Y.; Won, C.H. Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models. *Expert Syst. Appl.* **2018**, *103*, 25–37. [[CrossRef](#)]
43. Li, H.; Shen, Y.; Zhu, Y. Stock price prediction using attention-based multi-input LSTM. In Proceedings of the Asian Conference on Machine Learning, Beijing, China, 14–16 November 2018.
44. Cheng, L.-C.; Huang, Y.-H.; Wu, M.-E. Applied attention-based LSTM neural networks in stock prediction. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018.
45. Shah, D.; Campbell, W.; Zulkernine, F.H. A comparative study of LSTM and DNN for stock market forecasting. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018.
46. Lin, B.-S.; Chu, W.-T.; Wang, C.-M. Application of stock analysis using deep learning. In Proceedings of the 2018 7th International Congress on Advanced Applied Informatics (IIAI-AAI), Yonago, Japan, 8–13 July 2018.
47. Skehin, T.; Crane, M.; Bezbradica, M. Day ahead forecasting of FAANG stocks using ARIMA, LSTM networks and wavelets. In Proceedings of the 26th AIAI Irish Conference on Artificial Intelligence and Cognitive Science (AICS 2018), Dublin, Ireland, 6–7 December 2018.
48. Zhang, X.; Tan, Y. Deep stock ranker: A LSTM neural network model for stock selection. In Proceedings of the International Conference on Data Mining and Big Data, Shanghai, China, 17–22 June 2018; Springer: Berlin/Heidelberg, Germany, 2018.
49. Achkar, R.; Elias-Sleiman, F.; Ezzidine, H.; Haidar, N. Comparison of BPA-MLP and LSTM-RNN for stocks prediction. In Proceedings of the 2018 6th International Symposium on Computational and Business Intelligence (ISCBI), Basel, Switzerland, 27–29 August 2018.
50. Zeng, Y.; Liu, X. A-stock price fluctuation forecast model based on LSTM. In Proceedings of the 2018 14th International Conference on Semantics, Knowledge and Grids (SKG), Guangzhou, China, 12–14 September 2018.
51. Shao, X.; Ma, D.; Liu, Y.; Yin, Q. Short-term forecast of stock price of multi-branch LSTM based on K-means. In Proceedings of the 2017 4th International Conference on Systems and Informatics (ICSAI), Hangzhou, China, 11–13 November 2017.
52. Zhao, Z.; Rao, R.; Tu, S.; Shi, J. Time-weighted LSTM model with redefined labeling for stock trend prediction. In Proceedings of the 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI), Boston, MA, USA, 6–8 November 2017.
53. Nelson, D.M.Q.; Pereira, A.C.M.; de Oliveira, R.A. Stock market's price movement prediction with LSTM neural networks. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017.
54. dos Santos Pinheiro, L.; Dras, M. Stock market prediction with deep learning: A character-based neural language model for event-based trading. In Proceedings of the Australasian Language Technology Association Workshop 2017, Brisbane, Australia, 7–8 December 2017.
55. Buczkowski, P. Predicting stock trends based on expert recommendations using gru/lstm neural networks. In Proceedings of the International Symposium on Methodologies for Intelligent Systems, Warsaw, Poland, 26–29 June 2017; Springer: Berlin/Heidelberg, Germany, 2017.

56. Chen, K.; Zhou, Y.; Dai, F. A LSTM-based method for stock returns prediction: A case study of China stock market. In Proceedings of the 2015 IEEE International Conference on Big Data (Big Data), Santa Clara, CA, USA, 29 October–1 November 2015.
57. Qi, L.; Khushi, M.; Poon, J. Event-driven LSTM for forex price prediction. In Proceedings of the 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Gold Coast, Australia, 16–18 December 2020.
58. Pang, X.; Zhou, Y.; Wang, P.; Lin, W.; Chang, V. An innovative neural network approach for stock market prediction. *J. Supercomput.* **2020**, *76*, 2098–2118. [[CrossRef](#)]
59. Feng, F.; He, X.; Wang, X.; Luo, C.; Liu, Y.; Chua, T.-S. Temporal relational ranking for stock prediction. *ACM Trans. Inf. Syst. (TOIS)* **2019**, *37*, 1–30. [[CrossRef](#)]
60. Chung, H.; Shin, K.-S. Genetic algorithm-optimized long short-term memory network for stock market prediction. *Sustainability* **2018**, *10*, 3765. [[CrossRef](#)]
61. Li, J.; Bu, H.; Wu, J. Sentiment-aware stock market prediction: A deep learning method. In Proceedings of the 2017 International Conference on Service Systems and Service Management, Dalian, China, 16–18 June 2017.
62. Jin, Z.; Yang, Y.; Liu, Y. Stock closing price prediction based on sentiment analysis and LSTM. *Neural Comput. Appl.* **2020**, *32*, 9713–9729. [[CrossRef](#)]
63. Long, J.; Chen, Z.; He, W.; Wu, T.; Ren, J. An integrated framework of deep learning and knowledge graph for prediction of stock price trend: An ap-application in Chinese stock exchange market. *Appl. Soft Comput.* **2020**, *91*, 106205. [[CrossRef](#)]
64. Chen, M.-Y.; Liao, C.H.; Hsieh, R.-P. Modeling public mood and emotion: Stock market trend prediction with anticipatory computing approach. *Comput. Hum. Behav.* **2019**, *101*, 402–408. [[CrossRef](#)]
65. Qian, F.; Chen, X. stock prediction based on LSTM under different stability. In Proceedings of the 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Singapore, 17–18 April 2019.
66. Li, Z.; Tam, V. Combining the real-time wavelet denoising and long-short-term-memory neural network for predicting stock indexes. In Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA, 27 November–1 December 2017.
67. Song, Y.; Lee, J.W.; Lee, J. A study on novel filtering and relationship between input-features and target-vectors in a deep learning model for stock price prediction. *Appl. Intell.* **2019**, *49*, 897–911. [[CrossRef](#)]
68. Naik, N.; Mohan, B.R. Stock price movements classification using machine and deep learning techniques—the case study of Indian stock market. In Proceedings of the International Conference on Engineering Applications of Neural Networks, Athens, Greece, 25–27 August 2019; Springer: Berlin/Heidelberg, Germany, 2019.
69. Chatzis, S.P.; Siakoulis, V.; Petropoulos, A.; Stavroulakis, E.; Vlachogiannakis, N. Forecasting stock market crisis events using deep and statistical machine learning techniques. *Expert Syst. Appl.* **2018**, *112*, 353–371. [[CrossRef](#)]
70. Abe, M.; Nakayama, H. Deep learning for forecasting stock returns in the cross-section. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Melbourne, Australia, 3–6 June 2018; Springer: Berlin/Heidelberg, Germany, 2018.
71. Nakagawa, K.; Uchida, T.; Aoshima, T. Deep factor model. In Proceedings of the ECML PKDD 2018 Workshops, Dublin, Ireland, 10–14 September 2018; Springer: Berlin/Heidelberg, Germany, 2018.
72. 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.* **2017**, *83*, 187–205. [[CrossRef](#)]
73. Singh, R.; Srivastava, S. Stock prediction using deep learning. *Multimedia Tools Appl.* **2017**, *76*, 18569–18584. [[CrossRef](#)]
74. Yu, P.; Yan, X. Stock price prediction based on deep neural networks. *Neural Comput. Appl.* **2020**, *32*, 1609–1628. [[CrossRef](#)]
75. Yong, B.X.; Rahim, M.R.A.; Abdullah, A.S. A stock market trading system using deep neural network. In Proceedings of the Asian Simulation Conference, Melaka, Malaysia, 27–29 August 2017; Springer: Berlin/Heidelberg, Germany, 2017.
76. Li, Y.; Ni, P.; Chang, V. An empirical research on the investment strategy of stock market based on deep reinforcement learning model. In Proceedings of the COMPLEXIS 2019: 3rd International Conference on Complexity, Future Information Systems and Risk, Heraklion, Greece, 2–4 May 2019.
77. Shin, H.-G.; Ra, I.; Choi, Y.-H. A deep multimodal reinforcement learning system combined with CNN and LSTM for stock trading. In Proceedings of the 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, 16–18 October 2019.
78. Wu, J.; Wang, C.; Xiong, L.; Sun, H. Quantitative trading on stock market based on deep reinforcement learning. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019.
79. Carapuço, J.; Neves, R.F.; Horta, N. Reinforcement learning applied to Forex trading. *Appl. Soft Comput.* **2018**, *73*, 783–794. [[CrossRef](#)]
80. Kang, Q.; Zhou, H.; Kang, Y. An asynchronous advantage actor-critic reinforcement learning method for stock selection and portfolio management. In Proceedings of the 2nd International Conference on Big Data Research, Weihai, China, 27–29 October 2018; Association for Computing Machinery: Weihai, China, 2018; pp. 141–145.
81. Zhu, Y.; Yang, H.; Jiang, J.; Huang, Q. An adaptive box-normalization stock index trading strategy based on reinforcement learning. In Proceedings of the International Conference on Neural Information Processing, Siem Reap, Cambodia, 13–16 December 2018; Springer: Berlin/Heidelberg, Germany, 2018.

82. Si, W.; Li, J.; Ding, P.; Rao, R. A multi-objective deep reinforcement learning approach for stock index future's intraday trading. In Proceedings of the 2017 10th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 9–10 December 2017.
83. Pastore, A.; Esposito, U.; Vasilaki, E. Modelling stock-market investors as Reinforcement Learning agents. In Proceedings of the 2015 IEEE International Conference on Evolving and Adaptive Intelligent Systems (EAIS), Douai, France, 1–3 December 2015.
84. Long, W.; Lu, Z.; Cui, L. Deep learning-based feature engineering for stock price movement prediction. *Knowl.-Based Syst.* **2019**, *164*, 163–173. [[CrossRef](#)]
85. Wu, J.-L.; Yang, C.-S.; Liu, K.-H.; Huang, M.-T. A deep learning model for dimensional valencearousal intensity prediction in stock market. In Proceedings of the 2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST), Morioka, Japan, 23–25 October 2019.
86. Cho, C.-H.; Lee, G.-Y.; Tsai, Y.-L.; Lan, K.-C. Toward stock price prediction using deep learning. In Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing Companion, Auckland, New Zealand, 2–5 December 2019.
87. Minh, D.L.; Sadeghi-Niaraki, A.; Huy, H.D.; Min, K.; Moon, H. Deep Learning Approach for Short-Term Stock Trends Prediction Based on Two-Stream Gated Recurrent Unit Network. *IEEE Access* **2018**, *6*, 55392–55404. [[CrossRef](#)]
88. Hu, G.; Hu, Y.; Yang, K.; Yu, Z.; Sung, F.; Zhang, Z.; Xie, F.; Liu, J.; Robertson, N.; Hospedales, T.; et al. Deep stock representation learning: From candlestick charts to investment decisions. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018.
89. Hu, Z.; Liu, W.; Bian, J.; Liu, X.; Liu, T.-Y. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, Los Angeles, CA, USA, 5–9 February 2018.
90. Kim, T.W.; Khushi, M. Portfolio optimization with 2D relative-attentional gated transformer. *arXiv* **2020**, arXiv:2101.03138.
91. Zhang, Z.; Khushi, M. GA-MSSR: Genetic algorithm maximizing sharpe and sterling ratio method for robotrading. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020.
92. Shi, L.; Teng, Z.; Wang, L.; Zhang, Y.; Binder, A. DeepClue: Visual Interpretation of Text-Based Deep Stock Prediction. *IEEE Trans. Knowl. Data Eng.* **2019**, *31*, 1094–1108. [[CrossRef](#)]
93. Meng, T.L.; Khushi, M. Reinforcement Learning in Financial Markets. *Data* **2019**, *4*, 110. [[CrossRef](#)]