INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

**ISSN: 2454-132X**
**Impact factor: 6.078**
***(Volume 7, Issue 1)***
Available online at: https://www.ijariit.com

# Stock market price prediction using Neural Networks

*Deebak S.*
*deebak.selvaraj@gmail.com*
*United Institute of Technology, Coimbatore, Tamil Nadu*

*Dr. P. Sindhuja*
*sindhujapushparaj@gmail.com*
*United Institute of Technology, Coimbatore, Tamil Nadu*

## ABSTRACT

***In this research analysis, in addition to the conventional ARIMA model, specific long short-term memory (LSTM), stacked-LSTM, and concept-based LSTM were used to calculate next day expenses. Furthermore, using our expectation, we developed two modes of transmission, different and important identity. Our database information not only includes the usual end-of-day expense and transfer modules, but also includes corporate bookkeeping insights, which are effortlessly selected and used in samples. With the regular ARIMA model, learning the next model in anticipation of the next day's stock costs, especially long short-term memory model (LSTM), stacked-LSTM, and concept-based LSTM. In addition, using our forecast, we developed two exchange procedures and developed differential and scale. Our database information not only includes regular end-of-day cost and transfer modules, but also includes corporate bookkeeping metrics, which are deliberately selected and used in samples. Bookkeeping information is considered information and cost plans for a company that no longer relies on expanding the pioneering power of the model. The effect indicates that the LSTM beats any remaining model in relation to the forecast error and shows a lot better yield in our transfer practice on different models. Besides, we found that the stacked-LSTM model does not improve the advance control over the LSTM.***

***Keywords**: Stock Market Prediction, Tensor Flow, Deep Learning with Stock Market Analysis, and Neural Networks with Stock Market Price Prediction*

## 1. STOCK MARKET PRICE PREDICTION

### 1.1 Introduction

In the field of quantity exchange, the expectation of future security returns is at the focal point of the endeavor because the futures exchange process is constantly being sent and then done depending on our perspective on the currency market. There are two policy different strategies in the area of exchange, especially federal investigation and quantity exchange. The central strategy rests on the choice of exchange depending on the emotional outlook of the business or the future curriculum of the company, which is mainly dependent on public data, for example, market news, corporate measurements as well as the advancement of financial summary distributions. Again, the quantity transfer technique uses numerical models to solve the problem, thus avoiding interference with human subjectivity and perception. In general, the measurement system involves the use of direct repetitions and the ARIMA model, similar to the GARCH model, to capture the features of the time processes and the consistency of the unpredictability. These strategies were powerful for a certain period of time in older systems. As the computer movement in the currency business occurred, these models became less powerful. The quantity transfer industry has moved into this mode of 'deep learning period' which is now mostly used in a day. In this analysis, we expected stocks to return using the in-depth learning model, and more explicitly the LSTM and the LSTM models under consideration. The regular currency time arrangement uses the cost and quantity to anticipate future expenses. In this work, the information covers fixed costs and volumes just like corporate statistics. Typically these measurements are issued quarterly by companies and completely affect future value developments. The yield of our sample is the next day's cost or measured cost. When future cost is expected, we will develop a quantity transfer mechanism depending on the expectation. The arrival of our practice is different and the market.

### 1.2 Related Works

During the pre-deep learning era, monetary statistic modeling has principally focused within the field of ARIMA and any modifications on this, and also the result has established that the normal time-series model will offer good prognostic power to a limit. as an example, thanks to the uneven distribution in monetary statistic come, Minyoung Kim has replaced the normal most probability Estimation with associate uneven loss perform. [7] C.K. Lee et al. compared the foretelling performance of ARIMA and artificial neural networks on the Korean stock indicant. The work showed that ARIMA provided high correct forecasts than the back-propagation neural network. [8] recent deep learning ways have incontestable higher performances due to improved machine power and also the ability of learning non-linear relationships self-enclosed in numerous monetary options. Sreelekshmy Selvin et al. They terminated that CNN design is capable of distinguishing changes in trend of stocks outperforms different models. Yan and

Ouyang combined the wave remodel of the monetary series of your time with the LSTM and showed that the ensuing model beat the performance of ancient Support Vector Machine, and K-nearest Neighbors. [12] Thien Hai Nguyen et al. incontestable that integration of sediment options extracted from social media will improve the accuracy of prediction. [3, 5] what is more, Kim Won has developed a hybrid approach to mix LSTM and GARCH models and also the ensuing model has abundant lower prediction errors. [6]

### 1.3 Dataset and Features

The data we tend to utilized to train/develop and check our model embody 2 aspects: one. The daily costs and volumes for each SP five hundred stock from 2004 to 2013. 2. The accounting and company statistics for the SP five hundred stocks from 2004 to 2013. 2 sets were integrated by date and forward crammed missing statistics between 2 emotional date. once sizable amount of iterations in choosing the acceptable input, the input we tend to use square measure 'adjustment shut price', 'trading volume', 'Debt-to-Equity Ratio', 'Return on Equity', 'Price-to-Book' magnitude relation, 'Profit Margin', 'Diluted Earnings Per Share' and 'Company Beta'. we tend to applied the min-max scale to normalize the info between zero and one, thence stop the magnitude of bound options overwhelms others. for every stock in SP five hundred we've the daily knowledge mentioned on top of from 2004 to 2013. we tend to use the approximate magnitude relation of 70-15-15 to separate the info for every stock in coaching, development, and testing knowledge. In alternative words, we tend to use knowledge from 2013 to 2011 as coaching knowledge, 2012 as development knowledge and 2013 as testing knowledge. Our performance metrics and commercialism ways square measure thence designed on the info on 2013. A sample piece of our knowledge is shown Figure one.

| date | adj_close | volume | DE Ratio | Return on Equity | Price/Book | Profit Margin | Diluted EPS | Beta |
|---|---|---|---|---|---|---|---|---|
| 2019-12-17 | 0.950181 | 0.019849 | 0.652485 | 0.01414 | 0.046009 | 0.626554 | 1.0 | 0.24031 |
| 2019-12-18 | 0.964769 | 0.036554 | 0.652485 | 0.01414 | 0.046009 | 0.626554 | 1.0 | 0.24031 |
| 2019-12-19 | 0.966209 | 0.023068 | 0.652485 | 0.01414 | 0.046009 | 0.626554 | 1.0 | 0.24031 |
| 2019-12-20 | 0.980317 | 0.062588 | 0.652485 | 0.01414 | 0.046009 | 0.626554 | 1.0 | 0.24031 |

**Fig. 1: Arrangement of Google stock worth and company accounting statistics, from 2008 to 2019**

In order to look at the impact of prognostic powers in several money statistic, we tend to designed 3 deep learning models further united ancient statistic model. they're 1) statistic Model (ARIMA);2) RNN with LSTM Model (LSTM); 3) RNN with Stacked-LSTM (Stacked-LSTM);4) RNN with LSTM + Attention (Attention-LSTM).

**Time Series Model**: Auto Regressive Integrated Moving Average (ARIMA) model could be a wide used method for statistic prognostication (equation 1). During this work, we tend to followed the Box-Jenkins Methodology to create associate ARIMA model as a baseline to match with Deep Learning models. [4] For the ARIMA model, solely "adjusted shut price" was accustomed work the model. we tend to used outline statistics and operates like moving average and autocorrelation function to spot information trends and therefore the parameters (p, d, and q) of ARIMA model.

$$\delta Y_t(p,d,q) = \mu + \sum_{p=1}^{p}(\phi_p \times \delta Y_{t-p}) - \sum_{q=1}^{p}(\theta_q \times e_{t-q}) \quad where \quad \delta Y_t = Y_t - Y_{t-d} \qquad (1)$$

**RNN with Single/Stacked-LSTM :** The most plan of RNN is to use the consecutive observations learned from the sooner stages to forecast future trends. Long-Short Term Memory (LSTM) model is AN updated version of RNN. It will overcome the disadvantage of RNN in capturing long run influences.

LSTM introduces the memory cell that allows long-run dependency **between** time lags. The memory cells replaces the hidden layer neurons within the RNN and filters the data through the gate structure to take care of and update the state of memory cells. The gate structure includes input gate, forget gate and output gate.

The forget gate within the LSTM determines that cell state info is discarded from the model, it accepts the output from the previous time step ht one and also the new input of the present time step. Its main operate is to record the quantity of data reserved from the previous cell state to the present cell state. it'll output a worth between zero and one wherever zero suggests that complete reservation and one suggests that complete abandonment.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \qquad (2)$$

The input gate determines what proportion the present time network input crosstalk is reserved into the new cell state Ct, it avoids feeding the unimportant info into the present memory cell. it's 3 totally different components: 1) Get the state of the cell that has to be updated; 2) produce a replacement cell state; 3) Update the cell state to the present cell state.

$$i_t = \sigma(W_t \cdot [h_{t-1}, x_t] + b_i) \quad \hat{C}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad C_t = f_t * C_{t-1} + i_t * \hat{C}_t \quad (3)$$

The output gate controls what proportion the fresh created cell state are going to be discarded, the output info is first off determined by a sigmoid layer, then the fresh created cell state is processed by tanh, along with the sigmoid output to work out the ultimate output.

 

$$O_t = \sigma(W_\sigma \cdot [h_{t-1}, x_t] + b_o) \quad h_t = O_t * tanh(C_t) \tag{4}$$

Due to the upper randomness of economic statistic, we'll build up 2 models in LSTM and compare their performances: one single Layer LSTM memory model, and one Stacked-LSTM model. we have a tendency to expected the Stacked-LSTM model will capture a lot of randomness among the exchange because of its a lot of advanced structure. However, our experiments showed the other results, and that we can discuss below.

Attentions LSTM:Machine learning algorithms square measure impressed by biological phenomena and human perception. for example, we have a tendency to don't treat all data with equal importance, instead human perception focuses on the necessary elements 1st for the fresh received data. This development is analogous to the money market furthermore, because the costs of securities assign totally different levels of importance into the market data, and it prompts US to use the eye Mechanism to feature this feature into our RNN LSTM. In our model, we have a tendency to apply the soft attention, wherever we have a tendency to updated the input of the model by assignment weights to input data supported the training results and getting leads to a a lot of logical order. Mathematically, it's developed as:

$$e_t = tanh(W_a [x_1, x_2, ..., x_T] + b) \quad \alpha_t = \frac{exp(e_t)}{\sum_{k=1}^{T} exp(e_k)} \tag{5}$$

Where WA is that the weight matrix that may be a trainable parameter, it indicates the quantity of data that ought to be emphasized. The ensuing coefficient is αt, which can be accustomed weight the input. the first input are going to be replaced by the fresh weighted input and is employed for change the eye. This new input pays a lot of attention to the precise input feature sequence, extracting the key feature effectively and ignoring the redundant options victimization the eye weights. This can be the method of changing the first LSTM model into associate degree attention primarily based model. we have a tendency to use the framework setup by Qianqian for the Attention-LSTM and updated it to suit for money models.[2, 1] on paper we have a tendency to expect to ascertain higher model performance for Attention-LSTM than LSTM.

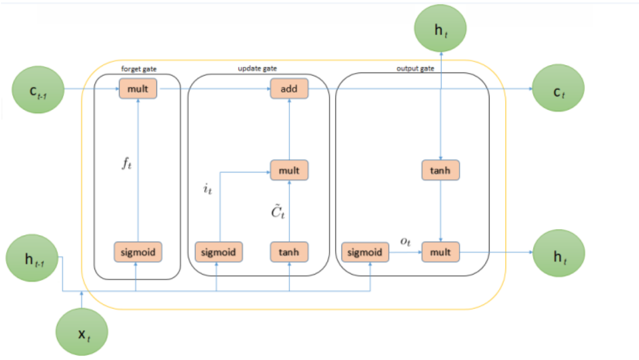


**Fig. 2: LSTM architecture**

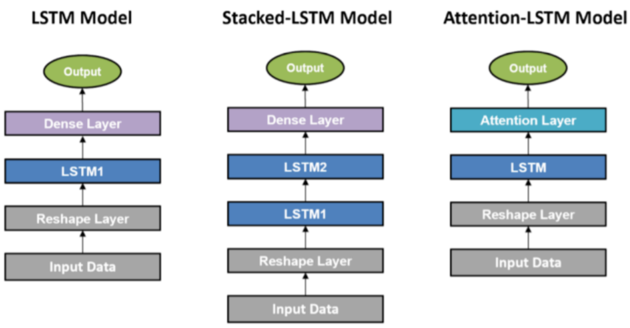


**Fig. 3: The design of 3 deep learning models**

### 1.4 Experiments/Results/Discussion

In this project, we are going to be exploitation information from the past to predict the come back on consecutive mercantilism day. To assess the model, our primary model performance metric is Mean square Error (MSE). The MSE are going to be performed on the check information that is completely unseen from the coaching and development stages. it's calculated between our foreseen worth and therefore the true worth.

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (\hat{y_i} - y_i)^2 \tag{6}$$

The secondary model performance metric is said with our mercantilism strategy. we have a tendency to build up 2 intraday mercantilism methods supported our prediction of performance in 2013:

**Long-Only Strategy:** For the prediction for every stock on consecutive day, if the prediction is positive, we have a tendency to obtain the stock at the open worth and sell the stock at shut worth within the same day. If the prediction is negative, no action is taken.

**Long-Short Strategy:** For the prediction for every stock on consecutive day, if the prediction is positive, we have a tendency to obtain the stock at the open worth and sell the stock at shut worth within the same day. If the prediction is negative, we have a tendency to short-sell the stock at the open worth out the short-sell at the close worth.

 

Notice that our mercantilism strategy is AN intraday mercantilism strategy, that means that we have a tendency to don't hold positions long, thus we have a tendency to solely take actions on identical day.

Our initial sample contains the whole universe of five hundred stocks in SP. However, when some experiments, we have a tendency to set to not use all five hundred stocks and switched our gear to the highest ten stocks in SP five hundred supported market capitalization. As larger market capitalization sometimes implicit additional stable money standing, additional predictable growth and additional complete public information from the businesses.

For the LSTM models, the tanh activation perform was used for the LSTM layers and therefore the sigmoid activation perform was used for the output layers, these activation functions are tried to be showing additional superior results than different activation functions. Moreover, the dropout likelihood of two hundredth is additionally applied for every hidden layer because the usual regularization methodology to scale back the overfitting issue. Finally, the adam improvement is employed for learning the parameters, and therefore the mean square error is used because the loss perform.

To increase our coaching speed and increase the possibility of convergence, we tend to use mini-batch in our coaching method. we've got tried an outsized variety of mini-batch sizes and eventually determined to use the scale of thirty, as bigger size doesn't guarantee convergence and typically cornered within the saddle purpose, and smaller sizes makes the coaching method quite slow. The lookback day is that the numbers of days we tend to use from the past to predict the long run, during this analysis we've got experimented an outsized variety of remember days and eventually determined to use remember days of twenty days, which means that we are going to be mistreatment the info from past twenty days to predict the stock costs within the next day.

**ARIMA model:** The parameters (p, d, q) of ARIMA model were determined to be (1, 0, 1) supported the MSE on the coaching set. This really makes the model associate ARMA model, wherever no differencing was performed to alter the stationarity. The MSE on the take a look at set of Google's knowledge was zero.0546. However, the model gave a linear prediction from take a look at knowledge, even the forecasted trend is opposite to the $64000 value trend. The results prompt ARIMA model didn't perform well in predicting non-linearity and semi permanent prediction. Hence, this model wasn't enclosed within the analysis of annual come.

**Deep Learning Models**: every model was trained by a hundred epochs for every stock, and also the MSE for the testing sample (which is that the 2013 data) is performed on every often stocks. Comparison of the results for every model is shown below in Figure three. Clearly the Attention-LSTM performs higher than each the Stacked-LSTM and LSTM models needless to say. As for semi permanent statistic prediction, the Attention-LSTM brings blessings of choosing the necessary and relevant info thence enhancing the prognostic accuracy. for sure stocks wherever it skilled giant fluctuations within the year like Google and IBM, the eye model beats the opposite 2 by quite an massive margin, more accentuation the importance of wishing on the context to predict the stock.

Typically, the deeper neural network is in a position to clarify a lot of difficult drawback than single-larger neural network. However, another fascinating finding we have a tendency to discovered is that the indisputable fact that the stacked-LSTM doesn't considerably vanquish the LSTM within the context of stock value prediction. Instead, the performance LSTM even beat the stacked-LSTM in bound instances. it's proven that the a lot of complicated representative doesn't essentially improve the prognostic power. it's probably thanks to 2 reasons: one. The a lot of difficult neural network illustration causes overfitting issue, the larger variety of parameters in stacked-LSTM memory model doesn't generalize well within the unseen knowledge.

2. The stacked-LSTM is a lot of appropriate in predicting classification issues instead of continuous statistic like stock costs: Due to the suboptimal performance within the stacked-LSTM model, the commercialism strategy is simply engineered on the LSTM and Attention-LSTM models. The come of our commercialism strategy in Figure four is that the total of the individual returns from every of the ten stocks on top of. The benchmark come is employed for examination the performance of the ways and is ready because the 2013 annual come of SP five hundred. The commercialism strategy supported our come prediction beat the benchmark by quite an massive margin, reflective the accuracy of the prediction from LSTM and Attention-LSTM models. Moreover, the annual come from the Attention-LSTM model beat the LSTM model, more confirming the higher prognostic power from Attention-LSTM model. because the attention model is in a position to research the context of the past and choose the foremost relevant info for the prediction, it's so higher to decide the long and short signal.

| Mean Squared Error ($*10^{-3}$) | | | |
|---|---|---|---|
| **Stock Ticker** | **LSTM** | **Stacked-LSTM** | **Attention-LSTM** |
| XOM | 25 | 25 | 11 |
| GE | 18 | 2 | 3 |
| WMT | 128 | 163 | 76 |
| MSFT | 40 | 27 | 16 |
| IBM | 28 | 7 | 6 |
| AAPL | 54 | 61 | 44 |
| GOOG | 35 | 37 | 26 |
| GS | 12 | 16 | 18 |
| PFE | 32 | 64 | 49 |
| JNJ | 57 | 123 | 41 |

**Fig. 4: MSE outline for 3 deep learning**

| Trading Strategy Annual Return | | | |
|---|---|---|---|
| Annual Return | LSTM | Attention-LSTM | Benchmark |
| Long-Only Strategy | 173% | 266% | 30% |
| Long-Short Strategy | 99% | 263% | 30% |

**Fig. 5: The come of mercantilism strategy supported LSTM, Attention-LSTM and Benchmark models**

## 2. CONCLUSION

The conclusion of the research establishes a statement framework to predict the costs of stocks. we have a tendency to leveraged the mixtures of worth, volumes and company statistics as input file. we have a tendency to projected, developed, trained and tested four models: ARIMA, LSTM, Stacked-LSTM and Attention-LSTM models, and designed up Long-Only and Long-Short mercantilism ways in line with our model predictions. The attention-LSTM shows a lot of superior results over alternative models due its its ability to assign totally different weights to the input options therefore mechanically opt for the foremost relevant options. therefore the Attention-LSTM is a lot of ready to capture the long-run dependence within the statistic and a lot of appropriate in predicting money statistic. Our superior mercantilism come from the Attention-LSTM additional validates our experimental result. Moreover, we've shown that despite the a lot of difficult model structure of stacked-LSTM over single LSTM model, the stacked-LSTM doesn't have higher model performance over the one LSTM model thanks to the potential of overfitting. One direction of future work are going to be handling the volatility of stock statistic. One issue of predicting exchange arises from its non-stationary behavior.[9] it'd be fascinating to ascertain however Attention-LSTM performs on de filtered information

## 3. REFERENCES


[1] Framework used: Keras, tensor flow.
[2] Toxic comment classification challenge, https://www.kaggle.com/qqgeogor/keras-lstm-attention- glove840b-lb-0-043/log.
[3] Ryo Akita, Akira Yoshihara, Takashi Matsubara, and Kuniaki Uehara. Deep learning for stock prediction using numerical and textual information.
[4] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung.
[5] JB Heaton, NG Polson, and Jan Hendrik Witte. Deep learning for finance: deep portfolios.
[6] Ha Young Kim and Chang Hyun Won. Forecasting the volatility of stock price index: A hybrid model integrating lstm with multiple garch-type models.
[7] Minyoung Kim. Cost-sensitive estimation of arma models for financial asset return data. Mathematical Problems in Engineering, 2015, 2015.
[8] CK Lee, Y Sehwan, and J Jongdae. Neural network model versus sARIMA model in forecasting korean stock price index (kospi).
[9] Hieu Quang Nguyen, Abdul Hasib Rahimyar, and Xiaodi Wang. Stock forecasting using m-band wavelet-based svr and rnn-lstms models
[10] Thien Hai Nguyen, Kiyoaki Shirai, and Julien Velcin. Sentiment analysis on social media for stock movement prediction.
[11] Sreelekshmy Selvin, R Vinayakumar, EA Gopalakrishnan, Vijay Krishna Menon, and KP Soman. Stock price prediction using lstm, rnn and cnn-sliding window model.
[12] Hongju Yan and Hongbing Ouyang. Financial time series prediction based on deep learning.