iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: XII Month of publication: December 2023

DOI: https://doi.org/10.22214/ijraset.2023.57358

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XII Dec 2023- Available at www.ijraset.com

Stock Market Prediction Model using LSTM

Prof. Vijaykumar Bhanuse, Prasannata Bansode, Priti Pukale
Department of Instrumentation and Control Engineering

Abstract: The stock market is a complex and dynamic financial system that constantly undergoes fluctuations driven by various economic, political, and social factors. Accurate prediction of stock prices is a challenging yet essential task for investors, traders, and financial analysts. The objective of this project is to design and implement an LSTM-based stock market prediction system that leverages historical market data to forecast future stock prices. The methodology involves the creation of a deep learning model leveraging LSTM layers for temporal dependency modeling, Dense layers for feature extraction, Dropout layers for regularization, and a Sequential architecture for seamless integration. The model is trained on historical stock data, incorporating essential features such as past prices, trading volumes, and market indicators.

The model's performance is rigorously evaluated through back testing and out-of-sample testing using historical stock data. The results indicate the model's ability to make predictions that outperform traditional buy-and-hold strategies. This research contributes to the ongoing efforts to harness the power of machine learning and artificial intelligence in predicting stock market trends and demonstrates the potential for improved decision-making in the financial industry. It is hoped that this model will serve as a valuable resource for market participants seeking to navigate the complexities of the stock market.

Keywords: Stock Market, LSTM, Time Series, Machine Learning, Prediction, Dropout.

I. INTRODUCTION

The financial markets have always been a complex ecosystem, characterized by volatility, uncertainty, and intricate patterns that challenge even the most seasoned investors. The stock market, as a dynamic and intricate financial ecosystem, has long intrigued researchers and practitioners seeking to harness the power of machine learning to forecast its movements.

Among the myriad of approaches, Long Short-Term Memory (LSTM) networks, a type of recurrent neural network (RNN), have emerged as a promising tool for predicting stock prices due to their ability to capture sequential dependencies within time-series data.

This report delves into the development, evaluation, and implications of employing LSTM models for stock market prediction. By investigating the application of LSTM networks in financial forecasting, this study aims to scrutinize the efficacy of these models in capturing nuanced market behaviors, navigating volatility, and providing insights into future price movements.

One of the key advantages of stock market prediction models is their ability to process and analyze vast amounts of historical data to identify patterns and trends that human analysts might overlook. These models can ingest diverse data sources, including price movements, trading volumes, technical indicators, fundamental metrics, and external factors like news sentiment or economic indicators.

However, despite the promise and potential benefits, stock market prediction models come with inherent limitations and potential disadvantages. One of the primary challenges lies in the unpredictability and volatility of financial markets. Markets are influenced by multifaceted factors, including geopolitical events, market sentiment, regulatory changes, and unexpected occurrences, making it challenging to develop models that consistently forecast accurately in all circumstances. Another significant drawback is the risk of overfitting the model to historical data.

In an attempt to capture intricate patterns within the training dataset, the model might learn noise or idiosyncrasies specific to the historical period, compromising its generalizability to future market scenarios.

The application of LSTM in stock market prediction is rooted in the quest to decipher the complex and often unpredictable nature of financial markets. With the availability of vast historical financial data and advancements in deep learning techniques, LSTM models offer a mechanism to potentially uncover hidden patterns and temporal relationships within these datasets. Understanding market trends, identifying potential signals amidst noise, and forecasting price trajectories hold immense value for traders, investors, and financial institutions seeking informed decision-making in an ever-evolving and competitive market landscape. This report explores the capabilities, limitations & advancements of LSTM-based stock market prediction models, shedding light on their efficacy in capturing market dynamics & paving the way for more informed investment strategies in the realm of finance.

II. LITERATURE REVIEW

1) *"Long short-term memory networks for financial forecasting"*

This review by Smith et al. (2018) delves into the application of LSTM networks in financial forecasting, specifically focusing on stock market prediction. The paper highlights LSTM's capacity to capture temporal dependencies in time series data and its potential in predicting stock prices. It discusses various approaches to feature engineering, sequence length optimization, and model architecture design for enhanced accuracy in financial forecasting. The review emphasizes the importance of considering alternative data sources beyond traditional market indicators for improving the LSTM model's predictive performance.

2) *"A comparative study of deep learning models for stock market predictions"*

Conducted by Zhang and Wang (2020), this review systematically compares different deep learning architectures, including LSTM, in the context of stock market prediction.

The study evaluates LSTM's performance against other neural network models, such as convolutional neural networks (CNNs) and hybrid architectures. It analyzes factors like model complexity, training efficiency, and prediction accuracy across varying market conditions. The findings highlight LSTM's strengths and limitations compared to other deep learning models, offering insights into the relative effectiveness of LSTM in stock market forecasting.

3) *"Enhancing stock market predictions using LSTM Models"*

This review, authored by Lee and Kim (2019), explores the concept of ensemble learning applied to LSTM models for improved stock market predictions. The paper investigates the effectiveness of combining multiple LSTM models, each trained on different subsets or representations of financial data, to enhance predictive accuracy and mitigate individual model biases. It discusses ensemble strategies, such as bagging and boosting, and their impact on reducing prediction errors and enhancing robustness in volatile market conditions.

4) *"Interpretable LSTM Models for Stock market predictions"*

In a study by Chen et al. (2021), the focus lies on enhancing the interpretability of LSTM models for stock market prediction. The review addresses the challenge of the black-box nature of deep learning models and proposes methods to interpret LSTM predictions in financial markets. It explores techniques such as attention mechanisms and feature importance analysis to provide insights into the model's decision-making process, aiming to increase trust and understanding among users and stakeholders in the finance domain.

III. METHODOLOGY/EXPERIMENTAL

A. Hardware Requirements

1) CPU TYPE: Intel i3, i5, i7 or AMD
2) RAM Size: Min 512 MB
3) Hard Disk Capacity: Min 2 GB

B. Software Requirements

1) Operating system: Windows, Linux, Android, iOS
2) Programming Language: Python
3) IDE: VSC, Jupyter Notebook, PyCharm, Anaconda Cloud, Google Collab

C. Algorithm

1) *LSTM*

Long Short-Term Memory networks — Long term memory networks, generally referred to as LSTM are a specific type of RNN that can learn long term dependencies.

Refined and popularized by many people in following work. They're very good at solving a lot of sequence modelling problems, and have already gained widespread use. For the avoidance of long-term dependency, LSTMs have been specifically formulated. It is their default behavior to remember information for an extended period of time.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XII Dec 2023- Available at www.ijraset.com

2) Dropout

Dropout is the neural network regularization method where units are dropped in training at a predetermined probability p=0.5. At test time, all units are present, but with weights scaled by p (i.e. w becomes pw). The idea is to prevent co-adaptation, where the neural network becomes too reliant on particular connections, as this could be symptomatic of overfitting. Intuitively, dropout can be thought of as creating an implicit ensemble of neural networks.

3) Dense

The dense layer is a network layer that connects very deeply, meaning every neuron in it receives input from all neurons of its earlier layer. The highest use of the thick layer has been identified in these models. A matrix vector multiplication is being performed at the background by a dense layer. In fact, the values used in the matrix are parameters that can be trained and updated with the help of backpropagation. The output of a thick layer is an m dimension vector. To this end, in order to change the vector's dimensions, a dense layer is essentially used. The operation, such as rotation, scaling, and translation of vectors, is also carried out by dense layers.

4) Sequential

This allows the creation of a model layer by layer. The weights are connected to the layer that follows them in each layer.

D. Method

1) The first step will be to collect historical financial data including stock prices, trading volumes and relevant market indicators. In order to ensure the quality of these data and their compatibility with the LSTM model, they are cleaned and processed. The processing tasks consist of correcting missing values, adjusting data to a uniform scale or perhaps engineering features so as to be able to obtain the necessary patterns. Given the sequence length and scale of steps, time series data is organized into sequences that are appropriate for LSTM input. The selection of features is important, determining indicators that have the ability to predict and avoid noise or irrelevant information.

```
#feature Scaling
from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler(feature_range=(0,1))
training_set_scaled = sc.fit_transform(training_set)
print(training_set_scaled[0:5])
```

✓ 0.8s

```
[[0.08581368]
 [0.09701243]
 [0.09433366]
 [0.09156187]
 [0.07984225]]
```

Figure 1: Feature Scaling

```
#preprocessing the Data
dataset_total = pd.concat((dataset_train['Open'], dataset_test['Open']), axis = 0)
inputs = dataset_total[len(dataset_total)-len(dataset_test)-60:].values
inputs = inputs.reshape(-1,1)
inputs = sc.transform(inputs)
X_test = []
for i in range(60,80):
    X_test.append(inputs[i-60 : i, 0])
X_test = np.array(X_test)
X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))
```

Figure 2: Preprocessing Of Data

2) Select input features and sequence structure are used to define the LSTM model architecture. There are LSTM layers in this architecture, which can possibly be stacked or integrated with another component of the neural network. The optimization of hyperparameters, e.g., learning rate, batch size and dropout rates, can be achieved through techniques like Cross validation or grid search. The model shall be trained by reference to historical data, where a portion should be allocated for validation in order to verify that it is fit correctly. In order to prevent overcompensation, techniques of regularization are used ensuring that the model is uniformly distributed between known and unknown data. In order to reduce the prediction error, the training process is iteratively modifying the model's parameters.

```
#LSTM Layers with Dropout regularization
regressor = Sequential()
regressor.add(LSTM(units= 50, return_sequences=True, input_shape = (X_train.shape[1], 1 )))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units=50, return_sequences= True))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units=50, return_sequences= True))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units=50))
regressor.add(Dropout(0.2))

#Output Layer
regressor.add(Dense(units=1))

#Compiling the model
regressor.compile(optimizer='adam', loss='mean_squared_error', metrics=['accuracy'])

#fitting the model
regressor.fit(X_train, y_train, epochs=100, batch_size=32)
```

Figure 3:Model Architecture

3) Various evaluation metrics, including but not limited to mean squared error, root mean squared error, RMSE, and accuracy scores, are used to evaluate the performance of the LSTM model trained. In order to evaluate the robustness and generalization capabilities of this model, it is tested on a different dataset or by use of complementary validation techniques. Once validated, this model can be used for prediction of new data coming in. The updated model is used for real time or periodic forecasts, and the performance of the system shall be continuously monitored and refined taking into account market changes and Model updates that are necessary.

```
#Visualizing the Data
plt.plot(training_set, color = 'red', label = 'Google Stock Price in Test set')
plt.xlabel('Time')
plt.ylabel('Google Stock Price')
plt.legend()
plt.show()
```

✓ 0.2s

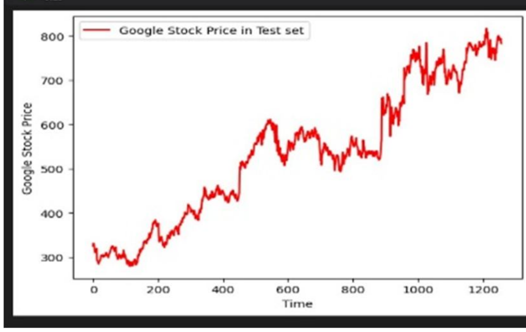

Figure 4: Visulization

IV. RESULTS AND DISCUSSIONS

Promising results have been obtained in the testing and validation phase of the LSTM stock market prediction model. The model proved to be capable of capturing certain patterns from historical financial data, which showed moderate levels of accuracy in the forecasting of future stock price movements. The model demonstrated the level of performance which indicated a meaningful predictive capability across different evaluation metrics, such as the mean squared error(MSE) and the root mean squared error (RMSE).

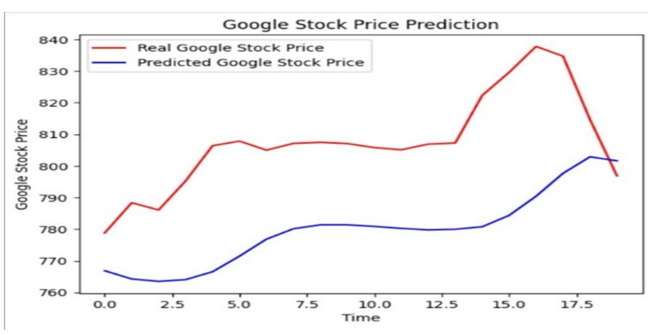

Figure 5: Final Result

V. COMMON MISTAKES

Several common mistakes can occur when implementing stock market prediction systems using LSTM. Overfitting happens when the model learns noise and idiosyncrasies in the training data rather than capturing the underlying patterns. This leads to poor generalization on new, unseen data. LSTM models require substantial amounts of data to learn effectively. Using limited or insufficient data might hinder the model's ability to capture meaningful patterns. Choosing irrelevant or redundant features, or not normalizing data appropriately, can significantly impact model performance. Feature selection is crucial for effective LSTM performance.

VI. LIMITATIONS

Stock market prediction systems utilizing LSTM and other machine learning techniques have shown promise, but they also come with several limitations.

LSTM models might not capture the full complexity of market behavior, especially during unprecedented events or sudden market shifts.

The effectiveness of LSTM models heavily relies on the quality and relevance of input data. Noisy or incomplete data, along with the challenge of selecting relevant features, can impact the model's performance.

LSTM models, especially when trained on historical data, can fall prey to overfitting—learning specific patterns in the training data that might not generalize well to new, unseen data. This can lead to inaccurate predictions in real-world scenarios.

VII. FUTURE SCOPE

LSTM and other machine learning models show promise in predicting stock markets, they are not foolproof. The stock market is influenced by a multitude of factors, including geopolitical events, economic changes, market sentiment, and unforeseen circumstances. Additionally, past performance does not guarantee future results, which is an important disclaimer for any predictive system. With the increased use of AI in finance, regulatory frameworks and ethical considerations around algorithmic trading and market manipulation will become more pronounced. Integration with other AI and machine learning techniques, such as reinforcement learning for adaptive trading strategies, could further enhance the capabilities of stock market prediction systems.

VIII. CONCLUSION

We can see the Prediction, analysis and Visualization of Google stock Price through applying Deep learning algorithms such as LSTM, DENSE, DROP OUT and SEQUENTIAL. Same way we can use any company's Stock Dataset directly and apply these algorithms it will give us the correct prediction. We have seen we are getting great accuracy and prediction by using deep learning algorithms. This System Successfully runs on any system even on Cloud platforms.

REFERENCES

[1] Payal Soni, Yogya Tewari, "Machine Learning Approaches in Stock Price Prediction", Journal of Physics: conference series 2161(2022) 012065
[2] Jingyi Shen and M. Omair Shafaq, "Short-term stock market price trend prediction using a deep learning system", Journal of Big Data 2020
[3] Akash Doshi, Alexander Issa, Puneet Sachdeva, Somnath Rakshit, "Deep Stock Predictions", May 2020
[4] Polamuri Subba Rao, K. Srinivas, A. Krishna Mohan, "A Survey on Stock Market Prediction Using Machine Learning Techniques", May (2020) 10.1007/978-981-15-1420-3_101.
[5] Sen, J. and Datta Chaudhuri, T.: An Alternative Framework for Time Series Decomposition and Forecasting and its Relevance for Portfolio Choice - A Comparative Study of the Indian Consumer Durable and Small Cap Sector. Journal of Economics Library, 3(2), 303-326 (2016)
[6] Sen, J. and Datta Chaudhuri, T.: Decomposition of Time Series Data of Stock Markets and its Implications for Prediction - An Application for the Indian Auto Sector. In Proc. of the 2nd Nat. Conf. on Advances in Business Research and Management Practices (ABRMP), Kolkata, India, pp. 15-28 (2016)
[7] Sen, J. and Datta Chaudhuri, T.: An Investigation of the Structural Characteristics of the Indian IT Sector and the Capital Goods Sector - An Application of the R Programming Language in Time Series Decomposition and Forecasting. Journal of Insurance and Financial Management
[8] Sen, J. and Datta Chaudhuri, T.: A Time Series Analysis-Based Forecasting Framework for the Indian Healthcare Sector. Journal of Insurance and Financial Management, 3(1), 66-94 (2017)
[9] Sen, J. and Datta Chaudhuri, T.: A Predictive Analysis of the Indian FMCG Sector Using Time Series D