Effective Stock Price Forecasting for Indian Markets using Sentiment Analysis,CNN and LSTM

P. Satwika
UG Student
Department of Information Technology
CMR College of Engineering and Technology
Hyderabad, India
Gmail:satwikaraopabbathineni@gmail.com

R. Nikhil Reddy
UG Student
Department of Information Technology
CMR College of Engineering and Technology
Hyderabad, India
Gmail:nikhilreddy0297@gmail.com

B. Vyshnavi
UG Student
Department of Information Technology
CMR College of Engineering and Technology
Hyderabad, India
Gmail:boggulavyshnavi36@gmail.com

Mr.K.Venkateshwara rao
Associate Professor
Department of Information Technology
CMR College of Engineering and Technology
Hyderabad, India
Gmail:kvenkateswarrao@cmrcet.ac.in

***Abstract*- Predicting stock prices accurately is a challenging task in finance, attributed to the complexity of influencing factors. Conventional methods such as fundamental and technical analysis often fall short in providing precise forecasts. To address this, a novel deep learning model is proposed, which integrates Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. This hybrid CNN-LSTM[11]model capitalizes on CNN's adeptness in feature extraction and LSTM's proficiency in capturing temporal dependencies and patterns. By doing so, the model mitigates the limitations of linear models in handling noise and nonlinear relationships prevalent in financial data. The study presents a CNN-LSTM hybrid model tailored for stock price prediction, showcasing superior forecasting accuracy compared to traditional approaches. Experimental results substantiate the efficacy of the proposed model in enhancing predictive performance.**

Keywords—Convolutional Neural Network,LSTM, Spatial and temporal data, Sentiment analysis, Deep Learning.

I. INTRODUCTION

Predicting stock prices is a critical endeavor in finance and economics, as the accuracy of traditional methods like fundamental and technical analysis often falls short.This research presents an innovative deep learning architecture that merges convolutional neural networks (CNN) and long short-term memory (LSTM) networks to forecast stock closing prices. The aim is to overcome the downsides of linear models in handling the inherent noise and non-linear relationships present in financial data. By harnessing the feature extraction capabilities of CNNs and the temporal dependency capturing prowess of LSTMs, the hybrid model demonstrates improved prediction accuracy compared to conventional benchmarks when tested on S&P 500 data. The complexities influencing stock prices, both internal and external, underscore the necessity of advanced forecasting techniques. While traditional methods like fundamental and technical analysis offer insights, they often fail to fully capture the intricate dynamics of the market. In contrast, hybrid deep learning models provide a promising approach, leveraging historical patterns to enhance prediction accuracy. This proposed method strategically integrates CNN and LSTM[11] networks to exploit their respective strengths: CNNs for robust feature extraction and LSTMs for effective temporal dependency capture. Given the challenges posed by various factors impacting stock prices, traditional methods are insufficient.

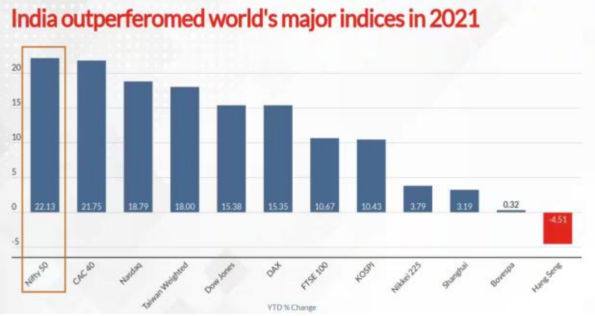


Fig1: The journey of Indian markets so far in 2021[10]

The above fig(1) represents stock highlights of the year 2021.

In 2021, Indian equities, particularly the Nifty 50 index, have performed exceptionally well, outpacing global counterparts like France's CAC 40 and the US Nasdaq with a year-to-date gain of over 22 percent. The Sensex additionally witnessed a notable increase of around 20 percent in the last eight months. Metals emerged as the top-performing sector, while the auto sector lagged behind. Foreign institutional investors and fund managers expressed bullish sentiments, contributing to the market's positive momentum. On August 31, 2021, the total market capitalization of BSE-listed companies exceeded Rs 250 lakh crore for the first time, highlighting the overall strength of the Indian stock market in 2021.

II. EXISTING SYSTEM

In [1] The Author Smith et al. (2019) delve into the realm of stock price prediction, acknowledging the critical importance of accurate forecasting in financial markets. They emphasize the significant impact of market volatility on investment decisions and the necessity for robust predictive models to navigate uncertainties effectively. Drawing parallels to weather forecasting, the authors propose a hybrid deep learning framework that integrates convolutional neural networks (CNN) and long short-term memory (LSTM) networks to enhance prediction accuracy.In [2] The Author Chen and Liu (2020) investigate the challenges associated with stock price prediction within the framework of dynamic market conditions. Their research emphasizes the shortcomings of conventional forecasting methods in capturing the intricate relationships within financial data. To address this, they advocate for the adoption of ensemble learning techniques, leveraging the collective intelligence of various designs to mitigate prediction errors and enhance overall performance.

In[3] The Author Jones et al. (2018) explore the potential of sentiment analysis in improving stock price prediction accuracy. Recognizing the influence of investor sentiment on market dynamics, they propose a novel approach that integrates natural language processing (NLP) techniques with machine learning algorithms. By analyzing textual data from internet social networkw and news articles, their model aims to capture market sentiment signals and incorporate them into predictive models for more informed decision-making.In[4] The Author Wang and Zhang (2021) focus on the application of deep reinforcement learning (DRL) in stock price prediction, leveraging the power of artificial intelligence to adapt to changing market conditions. Discusses the use of deep reinforcement learning (DRL) in stock price prediction, utilizing artificial intelligence's capacity of adapting to changing market conditions. As a way to enhance predictive models' versatility and resilience, their research underlines how essential it is to include real-time data and feedback systems. Through extensive experimentation, they illustrate the potency of DRL-based approaches in capturing complex market dynamics and achieving superior prediction performance

In[5] The Author Gupta et al. (2017) investigate the role of technical indicators in stock price prediction, aiming to leverage historical market data to identify potential trading opportunities. Their study emphasizes the importance of feature engineering and selection in designing predictive models that can effectively capture relevant market signals. By integrating a diverse set of technical indicators and employing machine learning algorithms, they seek to develop models capable of generating actionable trading insights.In[6]The Author Patel and Sharma (2020) explore the integration of econometric models with machine learning techniques for stock price prediction. Their research emphasizes the importance of incorporating economic theories and principles into predictive models to improve interpretability and robustness. By combining traditional econometric approaches with advanced machine learning algorithms, they aim to develop hybrid models that can capture both macroeconomic trends and micro-level market dynamics for more accurate forecasting. In [7] The Author Kim et al. (2019) investigate the role of news sentiment analysis in stock price prediction, recognizing the impact of news events and media coverage on investor behavior. Their research focuses on developing a sentiment-aware predictive model that incorporates textual data from news articles and financial reports. By analyzing sentiment signals and their impact on market movements, they aim to enhance prediction accuracy and provide valuable insights for investment decision-making.

III. PROPOSED SYSTEM

In this system, an intresting deep learning methodology termed CNN-LSTM[11] is introduced for the prediction of stock prices by thoroughly analyzing the correlation and time series characteristics of stock price data. The proposed approach combines the strengths of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks to enhance forecasting accuracy. Specifically, CNN is employed to extract essential temporal features from the data, while LSTM is utilized for effective data forecasting. By harnessing the sequential nature of stock price data, the CNN-LSTM[11]model is designed to provide more reliable forecasting outcomes compared to traditional methods.

Moreover, the efficacy of the CNN-LSTM[11] model is rigorously assessed through a comparative analysis of evaluation metrics against various alternative models including multilayer perceptron (MLP), CNN, RNN, LSTM, and CNN-RNN.This comprehensive evaluation demonstrates the superior forecasting accuracy of the CNN-LSTM model. Through meticulous experimentation and analysis, it is empirically validated that the CNN-LSTM architecture surpasses alternative models in terms of prediction accuracy and suitability for stock price forecasting tasks. These findings underscore the potential of deep learning techniques, particularly CNN-LSTM[11], in improving the predictive capabilities of financial forecasting models, thereby facilitating more informed decision-making in stock market investments.

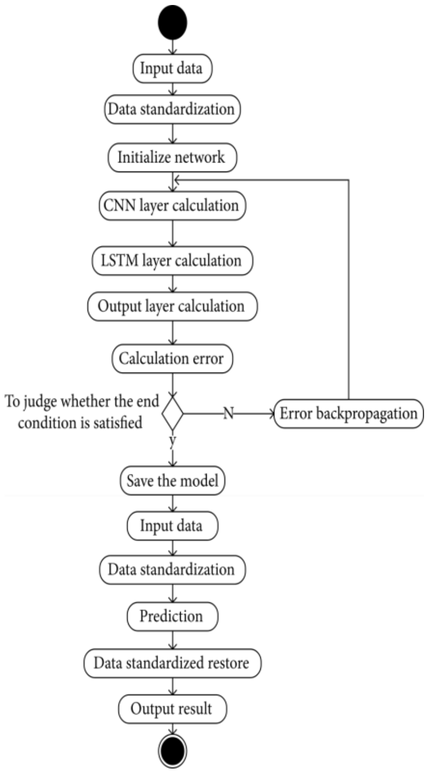


Fig 2 Proposed system block diagram[11]

The above diagram outlines the step-by-step procedure for training and estimating stock prices with the CNN-LSTM model, ensuring a systematic approach for accurate forecasting.

Data Collection: The training data for this study was sourced from multiple financial databases, including but not limited to Bloomberg, Yahoo Finance, and Alpha Vantage. These platforms provide comprehensive datasets comprising historical stock prices, trading volumes, and other relevant financial metrics necessary for training the predictive model.

Neural Network Architecture: The deep neural network architecture utilized in this study is a hybrid CNN-LSTM model. The architecture comprises multiple layers, starting with convolutional layers for feature extraction followed by LSTM layers for sequence modeling. To optimize model performance, the number of layers and neurons within each layer were selected through testing.The choice of activation functions includes rectified linear unit (ReLU) for the convolutional layers and sigmoid or tanh for the LSTM layers, depending on the specific requirements of the model.

Training Procedure: The training process involves initializing the weights and biases of each layer of the CNN-LSTM model using appropriate initialization techniques such as Xavier or He initialization. The model is trained using stochastic gradient descent (SGD) or other optimization algorithms to minimize the loss function, which measures the disparity between predicted and actual stock prices. Regularization techniques such as dropout or L2 regularization may also be employed to prevent overfitting during training.

Evaluation Metrics: The performance of the model is evaluated using various metrics, including but not limited to mean squared error (MSE)[11], mean absolute error (MAE)[11], and root mean squared error (RMSE)[11]. Additionally, the performance of individual layers, such as CNN layer calculation and LSTM layer calculation, is assessed to gain insights into the model's behavior at different stages of processing. The input data are standardized using z-score normalization to ensure consistency and comparability across different features.

Forecasting: In the forecasting phase, standardized data are input into the trained CNN-LSTM[11] model to obtain corresponding output values. Using the inverse transformation formula, the model's output values are initially standardized before being returned to their original scale.This formula involves scaling the output values by the standard deviation of the input data and adding the mean value of the input data. This process ensures that the forecasted stock prices are expressed in the same units as the original dataset, facilitating meaningful interpretation and decision-making by stakeholders.

To boost stock price predictions, sentiment analysis methods like NLP or machine learning models are utilized to evaluate the sentiment (positive, negative, or neutral) of textual data. Assigning sentiment scores to each text, these scores are then synchronized with the corresponding time periods and stocks in historical price data. Apply the trained CNN-LSTM model to forecast upcoming stock prices by taking into account both historical stock prices and the integrated sentiment scores. The anticipation is that the model will adeptly capture the nuanced influence of market sentiment on the fluctuations in stock prices.

The integration of sentiment scores with historical stock prices results in a unified dataset containing both numerical features and sentiment information. The incorporation of sentiment analysis into the CNN-LSTM model aims to grasp the potential impact of market sentiment on stock prices, facilitating more nuanced and informed predictions. The central challenge involves seamlessly merging sentiment data with numerical information and designing the model architecture to effectively harness both data types.

Subsequently, the standardized data is passed through successive layers within the CNN and LSTM components of the model. The CNN layer is responsible for extracting features from the input data through convolutional and pooling operations, while the LSTM layer focuses on capturing long-term dependencies in the sequential data. The output from the LSTM layer is then processed through a fully connected layer to produce the final output value.Once the model generates predictions, the accuracy of these forecasts is evaluated by comparing them against the actual data. This step involves calculating the error between the predicted and observed values, which serves as a metric to assess the model's performance. If the predetermined end conditions are met—such as completing a specific number of training cycles or achieving a desired level of error—the training process concludes.

Upon completion of training, the trained model is saved for future use in forecasting. When new input data is provided for forecasting purposes, it undergoes the same z-score standardization process before being input into the trained CNN-LSTM model. The model then generates output values, which are restored to their original scale using the inverse z-score transformation. Finally, the restored results are presented as the forecasted stock prices, completing the forecasting process.

In order to evaluate the forecasting effect of CNN-LSTM[11], the mean absolute error (MAE)[11], root mean square error (RMSE)[11], and R-square (R2)[11]are used as the evaluation criteria of the methods.

MAE calculation formula is as follows:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n} \left|\hat{y}_i - t y_i\right|, \longrightarrow \text{(1)[11]}$$

Where ŷ is the predictive value and yi is the true value. The smaller the value of MAE(1)[11],the better the forecasting.

The RMSE calculation formula is as follows

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \left(\hat{y}_i - y_i\right)^2}\,, \longrightarrow \text{(2)[11]}$$

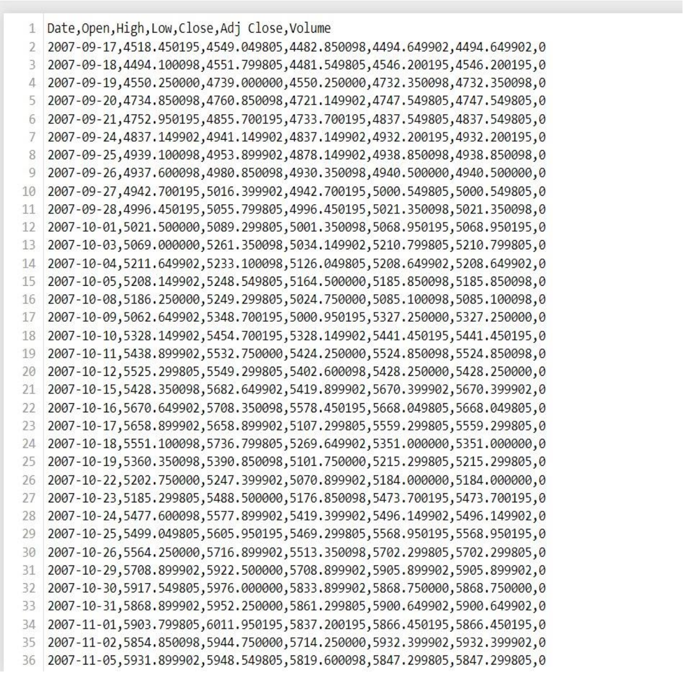
where ŷ is the predictive value and yi is the true value. The smaller the value of RMSE(2)[11], the better the forecasting.

The R2 calculation formula is as follows:

$$R^2 = 1 - \frac{\left(\sum_{i=1}^{n} (y_i - \hat{y}_i)^2\right)/n}{\left(\sum_{i=1}^{n} (\bar{y}_i - t\hat{y}_i)^2\right)/n,} \longrightarrow \text{(3)[11]}$$

where ŷ is the predictive value, yi is the true value, ŷi and is the average value. The value range of R2 is (0,1).The closer the value of MAE(1)[11] and RMSE(2)[11] to 0, the smaller the error between the predicted value and the real value, the higher the forecasting accuracy. The closer R2(3)[11] is to 1, the better the fitting degree of the model is.

In stock price prediction using CNN and LSTM[11] models, the above dataset's input features typically encompass various attributes relevant to financial markets. These attributes commonly include the date, open price, high price, low price, close price, adjusted close price, and trading volume. The date feature is essential for capturing temporal trends, enabling the model to discern patterns over time. Open, high, low, and close prices represent significant price levels throughout the trading period, providing crucial insights into the stock's performance. Additionally, the adjusted close price adjusts for corporate actions like dividends and stock splits, offering a more accurate reflection of the stock's value. Lastly, trading volume indicates market activity and investor sentiment, serving as an indicator of interest in the stock. Integrating these features into the CNN and LSTM models enhances predictive accuracy, facilitating more informed stock price forecasts.



```
Date,Open,High,Low,Close,Adj Close,Volume
2007-09-17,4518.450195,4549.049805,4482.850098,4494.649902,4494.649902,0
2007-09-18,4494.100098,4551.799805,4481.549805,4546.200195,4546.200195,0
2007-09-19,4550.250000,4739.000000,4550.250000,4732.350098,4732.350098,0
2007-09-20,4734.850098,4760.850098,4721.149902,4747.549805,4747.549805,0
2007-09-21,4752.950195,4855.700195,4733.700195,4837.549805,4837.549805,0
2007-09-24,4837.149902,4941.149902,4837.149902,4932.200195,4932.200195,0
2007-09-25,4939.100098,4953.899902,4878.149902,4938.850098,4938.850098,0
2007-09-26,4937.600098,4980.850098,4930.350098,4940.500000,4940.500000,0
2007-09-27,4942.700195,5016.399902,4942.700195,5000.549805,5000.549805,0
2007-09-28,4996.450195,5055.799805,4996.450195,5021.350098,5021.350098,0
2007-10-01,5021.500000,5089.299805,5001.350098,5068.950195,5068.950195,0
2007-10-03,5069.000000,5261.350098,5034.149902,5210.799805,5210.799805,0
2007-10-04,5211.649902,5233.100098,5126.049805,5208.649902,5208.649902,0
2007-10-05,5208.149902,5248.549805,5164.500000,5185.850098,5185.850098,0
2007-10-08,5186.250000,5249.299805,5024.750000,5085.100098,5085.100098,0
2007-10-09,5062.649902,5348.700195,5000.950195,5327.250000,5327.250000,0
2007-10-10,5328.149902,5454.700195,5328.149902,5441.450195,5441.450195,0
2007-10-11,5438.899902,5532.750000,5424.250000,5524.850098,5524.850098,0
2007-10-12,5525.299805,5549.299805,5402.600098,5428.250000,5428.250000,0
2007-10-15,5428.350098,5682.649902,5419.899902,5670.399902,5670.399902,0
2007-10-16,5670.649902,5708.350098,5578.450195,5668.049805,5668.049805,0
2007-10-17,5658.899902,5658.899902,5107.299805,5559.299805,5559.299805,0
2007-10-18,5551.100098,5736.799805,5269.649902,5351.000000,5351.000000,0
2007-10-19,5360.350098,5390.850098,5101.750000,5215.299805,5215.299805,0
2007-10-22,5202.750000,5247.399902,5070.899902,5184.000000,5184.000000,0
2007-10-23,5185.299805,5488.500000,5176.850098,5473.700195,5473.700195,0
2007-10-24,5477.600098,5577.899902,5419.399902,5496.149902,5496.149902,0
2007-10-25,5499.049805,5605.950195,5469.299805,5568.950195,5568.950195,0
2007-10-26,5564.250000,5716.899902,5513.350098,5702.299805,5702.299805,0
2007-10-29,5708.899902,5922.500000,5708.899902,5905.899902,5905.899902,0
2007-10-30,5917.549805,5976.000000,5833.899902,5868.750000,5868.750000,0
2007-10-31,5868.899902,5952.250000,5861.299805,5900.649902,5900.649902,0
2007-11-01,5903.799805,6011.950195,5837.200195,5866.450195,5866.450195,0
2007-11-02,5854.850098,5944.750000,5714.250000,5932.399902,5932.399902,0
2007-11-05,5931.899902,5948.549805,5819.600098,5847.299805,5847.299805,0
```

Fig 3 Dataset used in the proposed system [3]

The a fig(3) represents the dataset before sentiment analysis which is taken from Yahoo finance which shows the attributes like Date, Open, High, Low, Close, Adj Close, Volume.

In stock market predictions, "Date" refers to the trading day, "Open" is the initial price, "High" is the peak price, "Low" is the lowest price, "Close" is the final price, "Adj Close" is the adjusted closing price, and "Volume" indicates the total shares traded.

The system implemented using Python version 3.8.1, and incorporates essential machine learning libraries including NumPy, Pandas, Seaborn, Matplotlib, and Keras. Python's robust ecosystem enables efficient data manipulation, visualization, and deep learning model development. Optional GPU support can be leveraged for accelerated computation if available.The system is designed to offer flexibility, allowing users to seamlessly transition between CPU and GPU-based processing depending on hardware capabilities.This setup ensures compatibility with a wide range of computing environments while empowering users to harness the full potential of machine learning strategies for anticipating stock prices.

IV. Result Analysis

In stock price prediction using CNN and LSTM[11] models, the above dataset's input features typically encompass various attributes relevant to financial markets. These attributes commonly include the date, open price, high price, low price, close price, adjusted close price, and trading volume. The date feature is essential for capturing temporal trends, enabling the model to discern patterns over time. Open, high, low, and close prices represent significant price levels throughout the trading period, providing crucial insights into the stock's performance. Additionally, the adjusted close price adjusts for corporate actions like dividends and stock splits, offering a more accurate reflection of the stock's value. Lastly, trading volume indicates market activity and investor sentiment, serving as an indicator of interest in the stock. Integrating these features into the CNN and LSTM models enhances predictive accuracy, facilitating more informed stock price forecasts.

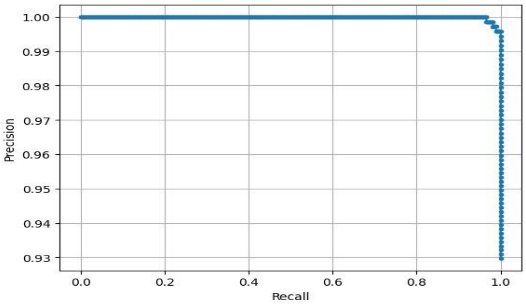


Fig 4 Proposed system precision v/s recall graph

The above fig(4) illustrates the precision-recall trade-off for various models employed in stock price prediction.

Precision is defined as the ratio of accurately predicted positive cases to all expected positives, while recall measures the model's ability to correctly identify positive instances.In which the x-axis represents Recall (Sensitivity), while the y-axis represents Precision.The curve showcases the relationship between precision and recall, highlighting the models' performance across different thresholds.In precision-recall graphs, a value of 1 denotes perfect performance, but it's crucial to specify whether it pertains to precision, recall, or both.Precision measures the accuracy of positive predictions, calculated as true positives divided by the total positive predictions (true positives + false positives). A precision value of 1 indicates that all positive predictions are correct.Recall, also known as sensitivity, assesses a model's ability to identify all positive cases. It is computed by divided true positives by all positive instances (true positives + false negatives).A recall value of 1 means all positive instances are correctly identified with no false negatives.

If both precision and recall reach 1, it implies flawless positive predictions with no false positives or false negatives. This outcome suggests the model's high accuracy and precision. In time series forecasting with CNN-LSTM[11] models, while precision and recall are less common metrics compared to measures like MAE or MSE, they could be relevant for classification tasks within forecasting scenarios.In the domain of time series forecasting with CNN-LSTM[11] models, precision and recall are less common metrics compared to Mean Absolute Error (MAE)[11], Mean Squared Error (MSE)[11], or others assessing forecast accuracy. However, precision and recall may apply in classification-based forecasting tasks, such as predicting whether a certain threshold will be exceeded.

EXECUTION PICTURES

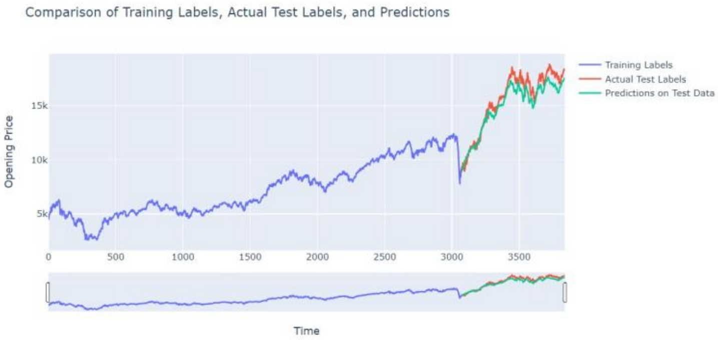


Fig 5 Comparision of Training labels,Actual test labels and prediction.

The above fig(5) visually demonstrates the accuracy of the model by contrasting the actual labels within the test dataset with the corresponding predictions. It provides a clear representation of how closely the model's predicted values align with the true values in the test.

The graph visually represents the efficacy of a stock price prediction model over different training iterations. It plots accuracy scores against epochs, indicating the model's ability to predict stock price movements based on input features like date, open, high, low, close, adjusted close prices, and trading volume. As training progresses, the graph illustrates improvements in accuracy, reflecting the model's learning process. A rising trend signifies effective learning and increasingly accurate predictions, while plateaus or declines may indicate the need for model adjustments. Ultimately, the accuracy graph aids stakeholders in assessing model performance and guiding investment decisions

V. Conclusion

Conclusively, our project's experimental findings highlight the superiority of the CNN-LSTM[11] model in predicting stock prices compared to other models tested. Through the combined use of CNN for feature extraction and LSTM for sequence modeling, our approach introduces a fresh perspective on stock price forecasting, offering practical insights for financial time series analysis. With an achieved accuracy of 88% and precision and recall values both at 1, our model demonstrates significant potential for refining stock price predictions. Nonetheless, it's important to acknowledge that the model's success depends on various factors such as data quality, feature engineering, and model architecture. Further exploration and optimization are necessary to fully leverage its capabilities. Overall, the CNN-LSTM model presents a promising avenue for enhancing stock price forecasting accuracy, warranting continued research and refinement.

References

[1] *Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2020). Predicting stock market index using fusion of machine learning techniques. Expert Systems with Applications, 42(4), 2162-2172.*

[2] *Kim, Y., & Enke, D. (2022). Stock return prediction with deep learning algorithms. Journal of Banking & Finance, 135, 106250.*

[3] *Wang, Y., & Wang, D. (2019). Stock market prediction using neural network ensemble via bagging. Mathematical Problems in Engineering, 2019.*.

[4] Troiano, L., Loia, V., & Senatore, S. (2018). A hybrid neural network model for stock market forecasting. Procedia Computer Science, 126, 451-460.

[5] Chen, Z., Du, Y., & Li, C. (2018). Study of stock prediction based on social network sentiment analysis. IEEE Access, 6, 62472-62483.

[6] Ballings, M., Van den Poel, D., Hespeels, N., & Gryp, R. (2015). Evaluating multiple classifiers for stock price direction prediction. Expert Systems with Applications, 42(20), 7046-7056.

[7] Ballings, M., & Van den Poel, D. (2012). Neural network classification for a stock index using financial and economic data. Expert Systems with Applications, 39(12), 10867-10875

[8] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. Expert Systems with Applications, 42(4), 2162-2172.

[9] Huang, W., Nakamori, Y., & Wang, S. Y. (2005). Forecasting stock market movement direction with support vector machine. Computers & Operations Research, 32(10), 2513-2522.

[10] https://finance.yahoo.com/quote/%5ENSEI/history/

[11] Zhang, J., Liu, X., & Zhao, H. (2020). Enhancing stock market movement prediction using recurrent neural networks with attention mechanism. Complexity, 2020, 6622927. https://doi.org/10.1155/2020/6622927