

An Improved Index Price/Movement Prediction by using Ensemble CNN and DNN Deep Learning Technique

Jitendra Sisodia¹, Amar Nayak^{2*}, Rajesh Boghey³

^{1,2,3} Department of Computer Science, Technocrats Institute of Technology, (Excellence), Bhopal, M.P. India.

Received: 22.01.2024 • Accepted: 02.03.2024 • Published: 20.03.2024 • Final Version: 27.04.2024

Abstract: As it is known, stock prediction is always a challenging task. The goal of any stock prediction method is to develop a robust method for predicting stock movement piece that can be used to improve investment decisions and accurate models. The paper proposes a hybrid model that combines the strengths of deep learning models CNN and DNN, to develop a comprehensive methodology for the prediction of stock/index prices on Banknifty (NSE Bank), a highly volatile Indian sectorial Index that represents 12 major banks of the country. The hybrid model consists of two main components a CNN for feature extraction and a DNN for regression or classification tasks. In the context of stock price prediction, CNN layers can be used to extract features from input data (such as stock prices and indicators) related to an estimated future value. DNN layer can be used to combine features learned from the CNN layers. Model performance will be evaluated using various metrics including Accuracy, Precision, Recall, Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R-squared. The method also analyses the impact of various factors on stock prices, including market volatility, economic indicators, and geopolitical events. The achieved accuracy of 97.48% indicates that the model was successful in accurately predicting the stock prices of Bank Nifty. The proposed method is expected to provide investors and financial analysts with a valuable tool for making informed investment decisions.

Keywords: Share/ Index, Prediction, CNN, DNN, Ensemble model

1. Introduction

With the help of Artificial Intelligence & Machine Learning (AIML), humans can make many kinds of predictions such as Rain predictions, Astrological predictions, GDP growth, and winning predictions in games and sports [1]. Even sometimes AIML can predict the important decisions of the Governing body's policies, the Company's growth, and the future, it can also predict some public-related needs like House prices and Car prices. Apart from all these predictions Stocks/Indices prices or movement prediction is one of the important topics across the world because a country's economy, as well as the world economy, is based upon the share/stock markets [2]. The major approach used for Stocks/Indices movement prediction is to use past data, based on past data of movements and can predict the prices for the same with more and more accuracy, after taking features such as Date, Open, High, Low, Close these five features are always important to predict future prices on behalf of trading data [3]. In the case of past events, researchers can also predict the Stocks/Indices price.

* Corresponding Author: Tel.: +919826039844. E-mail address: amar.n1975@gmail.com

Apart from both situations predicting the stock index during the pandemic time as COVID-19 [4], [5] the toughest task because during the last week of March 2020 across the world all were down by 40% and stocks were gone down by 20% to 80%. Similarly, some past experiences as scams in 1992 (Harshad Mehta) then later scams done by Ketan Parekh the Market had down huge. During the COVID time, the biggest drawback of NEWS through online platforms is Misinformation [6].

After analyzing the history of the stocks anything can happen, a stock can give massive returns in a very short span as Ruchi Soya, also Patanjali stock has increased around 100 times return in less than a year. Instead of such cases, Yes Bank's stock downside from 400 points to 13 points in less than six months span. Still governing bodies have fitted validators about to rise and fall of stock i.e., close circuit and upper circuit. But fundamentally strong company stocks' movements are always predictable because such movements always remain sensible and depend upon fundamentals and growth.

Sometimes the movements of Stocks/Indices prices depend upon Government policies, Inflation; Global issues, Dollar Index, employment data, and the Governing body's policies i.e., SEBI, RBI, and IRDA, and depends upon sectorial announcements and needs. For example, Indian Public sector NTPC (Power sector giant) stock price depends upon the price of coal because company generated electricity from the coal, if the coal price will be increased then NTPC stock will be also go down and vice versa if the coal price goes down then the share price will increase there are few more factors in the stock as coal need transportation it will also go up and down because of transportation cost is increasing or decreasing. For the prediction of share price and the company's actual evaluations, a lot of factors are involved. In the case of IT sector companies in India so most of the companies are getting payments from North America in the USD and these days the price of the Dollar is going up comparatively in Indian Rupees and other major currencies such as GBP, Euro, and JPY even in India the expenses are in INR even Indian IT giants are getting more revenue if they will convert the payments in USD to INR in terms of Indian Rupees.

Equity price is important for analyzing a company and Index price is important to know about the growth of a country or a sector in the case of sectorial Index i.e., NIFTY 50 and Banknifty in India. The prediction is always required for better planning for economic developments, and in the case of the investors investors are always keen for better returns. The research aims to contribute to the field of finance and machine learning by providing a comprehensive analysis of the factors that influence stock/index prices and developing and evaluating accurate and reliable prediction models with the help of a Hybrid Network of CNN, and DNN that can help investors and financial analysts make better investment decisions.

- (i) Identified the most significant financial and economic indicators that influence stock/index prices in the case of the Indian sectorial Index NSE BANK (BankNifty).
- (ii) Developed a technique using CNN and DNN deep learning to predict the stock/index and improve the accuracy and performance of automatic prediction.
- (iii) Compared the various machine learning algorithms including neural networks, decision trees, and support vector machines, in predicting future stock/index prices.
- (iv) Provided insights and recommendations for investors and financial analysts on how to use predictive models to make informed investment decisions and analyze the impact of different time intervals and prediction horizons on model performance.

The paper is organized as follows, section 2 represents related work, section 3 represents proposed method, section 4 provides implementation details and result analysis, and section 5 provides conclusion and future research direction.

2. Related Work

DNN [7] is a Deep Neural Network prediction work in which researchers used five models and took different kinds of data in all five models for HAN'S model they took News Information from Twitter and in this case, the accuracy was 47.8%. They have used the ND-SMPF Model and took historical price as well as Twitter data and because of this, they have improved the accuracy by 58.63%. In between researchers did some more experiments cause these experiments' accuracy was dropped and hiked. But finally, got a high accuracy across all the models i.e. 65.16% after taking years of trading data and two years of Twitter's NEWS data. Fundamental data, and for model-2 they have taken trading data and News/Events Data.

A Deep Learning-based Long Short-Term Memory (LSTM) Algorithm [8] has given 83.88% prediction results in the case of SBI. About NSE Nifty50 other companies for HDFC BANK 83.56%, Infosys 83.49%, BPCL 83.00%, TCS 82.85%, and ITC 83.67%, for L&T 83.48%, for UPL 82.99%, for CIPLA 83.34% and for Indian OIL 83.13%. In this method, the researchers have picked 10 years of data i.e. 10 December 2011 to 10 December 2021. Picked 10 years of data used for the model's training and testing after normalizing the data.

In some research, the researchers used the Linear Regression Model, the same can be applied when the repetition of errors or error variance is constant [9]. When stock / Index price data is nonlinear, and the error frequency has no variance across the time XGBoost [10] and DNN [7] are more useful methods i.e., for the nonlinear data. XGBoost is a sensible work for stock price forecasting with accuracy, feature importance analysis, and the ability to handle complex feature-outcome relationships. Apart from all these XGBoost can handle non-linear relationships, scalability, and versatility, which means it can be used for both regression and classification. XGBoost method identified some disadvantages mentioned as Overfitting, Complexity cause of its black-box nature and XGBoost data quality issues it's needed always high-quality data.

During the research, the author in [11] found that the movement of share price depends upon news and events also. In this investigation, the researchers used both linear and non-linear models to predict the stock price the important part is that the author had taken publicly available news related to the stock market from Reuters and Bloomberg from October 2006 to November 2013 also mentioned that during 2007-2010 was economic downtime and 2011-2013 was the modest recovery time, this information was most important for the objective of the research mentioned that 106,521 documents from Reuters News and 447,145 documents from Bloomberg News, News title and contents are extracted from website and mainly focused on S&P 500 Index apart from the NEWS and Events details picked up the Stocks and Indices prices from Yahoo Finance. The results are analyzed for S&P 500 the dev and test result's accuracies are 59.60% and 58.94%. In the case of individual stocks, the Wall-Mart achieved better accuracy which was 70.45% (dev) and 69.87% (test).

Selvin et al. [12] Stock Price Prediction using LSTM, RNN, and CNN-sliding window model, it's a pure technological approach and it was a better form of prediction model which used three different deep learning architectures for the price prediction and comparison of their performances of National Stock Exchange's Companies. The method used a sliding window approach for predicting future values on a short-term basis and the performance of the models can be calculated using percentage error.

III. Proposed Method

This section describes how to predict the Stock Market/Equity Market, i.e., Nifty Bank's (NSE BANK) future price/movement.

3.1 System architecture

The proposed method uses an ensemble CNN and DNN Deep Learning method to predict the future price of the highly volatile Index "Nifty Bank, " a sectoral Index representing the Indian Banking sector in the NSE (National Stock Exchange). The method has five phases' Data Collection, Data pre-processing, Model Development, Model evaluation, and Prediction.

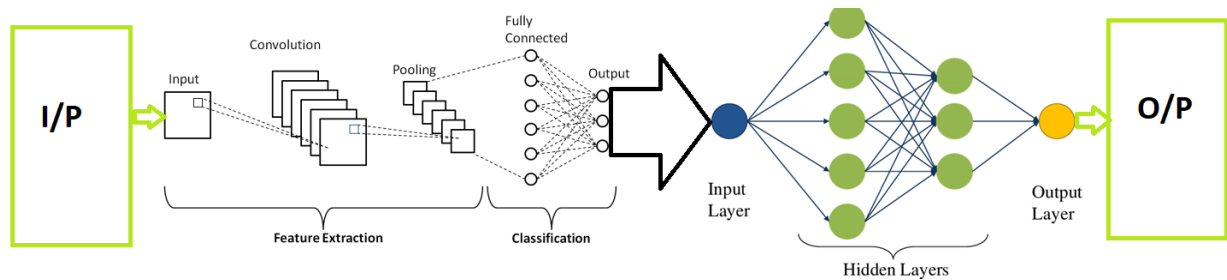


Figure 1. Proposed System Architecture

Figure. 1 represents the proposed system architecture consisting of an I/P layer, feature extraction layer, classification layer, hidden layer, and output layer. The ensemble model combining Convolutional Neural Networks (CNN) and Deep Neural Networks (DNN) for Bank Nifty future price prediction can leverage the strengths of both architectures to improve accuracy and capture relevant features in the data. In this method, an ensemble model consists of two main components a CNN and DNN for prediction tasks.

Input Layer: This is the first layer of the proposed architecture. It receives the historical data of the stock market / NSE Bank. This received data will be related to past movements of the market i.e. past trading data.

CNN (Convolutional Neural Network): CNN is a neural network that can learn features of data. In the context of stock price prediction, CNN layers can be used to extract features from input data (such as stock prices and indicators) related to an estimated future value. We designed it to provide inputs from NSE Bank data from past trading days and predict future movements.

DNN (Deep Neural Network): DNN is a type of Neural Network that can understand the nonlinear relationships in data. The method uses DNN as the last layer of the model because in this work of equity/stock price prediction on behalf of past data, a DNN layer can be used to combine features learned from the CNN layers and produce the final result.

Output Layer: This is the final layer of the structure. This layer is used for the evaluation and prediction.

3.2 Proposed flow diagram

Figure. 2 represents the flow diagram of the proposed model. The proposed model will be a hybrid model built with the combination of CNN and DNN.

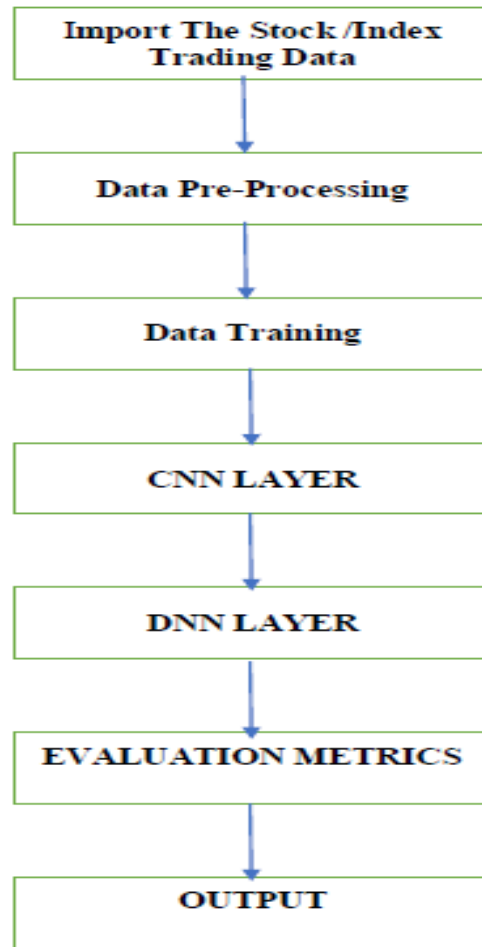


Figure 2. Proposed flow diagram

The proposed steps represented in fig. 2 consist of data collection, data pre-processing, training, CNN layer, DNN layer, evaluation matrix and output layer.

3.2.1 Data Collection:

The data is collected from Yahoo Finance [11]. The dataset date ranges are start_date = '2007-09-28' and end_date = '2022-05-04'. It has a total of 3301 rows out of which we are using 80% i.e., 2640 rows for training, and 20% i.e., 661 rows for testing.

	Open	High	Low	Close	Adj Close \
Date					
2007-09-28	7838.250000	8082.850098	7836.049805	8042.200195	8042.106934
2007-10-01	8008.549805	8085.149902	7913.299805	7987.500000	7987.407227
2007-10-03	8029.799805	8235.799805	7820.250000	8097.899902	8097.805664
2007-10-04	8083.299805	8086.700195	7828.649902	8035.899902	8035.806641
2007-10-05	8038.100098	8066.549805	7789.700195	7845.250000	7845.158691

	Volume
Date	
2007-09-28	0
2007-10-01	0
2007-10-03	0
2007-10-04	0
2007-10-05	0

Last five rows of data:

	Open	High	Low	Close \
Date				
2022-04-25	35792.351562	36270.148438	35511.398438	36082.351562
2022-04-26	36515.351562	36604.550781	36265.898438	36404.800781
2022-04-27	36067.550781	36175.898438	35747.148438	36028.851562
2022-04-28	36189.949219	36501.101562	35951.101562	36422.199219
2022-04-29	36474.050781	36718.800781	35978.949219	36088.148438

Figure 3. Dataset yfinance library

Figure. 3 represents yfinance dataset's first five and last five rows [13]. The features in yfinance datasets are date, open, high, low, close, and adj close.

3.2.2 Data Pre-processing

Data pre-processing techniques data cleaning, normalization, and feature selection is used to preprocess the data before being used in the algorithm. The dataset is divided in the ratio of 80% and 20% for the training and testing.

A. Data Cleaning: The data cleaning step handled the missing data, removed the duplicate data, handled the outliers, and corrected the errors. This process will be applied frequently.

B. Normalization: Normalization is one of the important approaches for stock price prediction. Fluctuation always occurs in the stock market. With the help of Normalization, data can be transformed so that it can be easily analyzed and compared [14]. Min-Max Scaling is used for scaling the data so it falls within a specified range.

Scaling formulae:

$$M = (x - x_{\min}) / (x_{\max} - x_{\min}),$$

where M= New Value, X= Original Cell Value, xmin = Minumum Value of Column,

xmax = Maximum Value of Column

C. Feature Selection: During this step, the needed attributes are fed to the CNN deep learning model. The features used in implementation are DATE, OPEN, HIGH, LOW, and CLOSE as input to the CNN model. And in some cases, Moving Average, Volume, and economic factors are also used [15].

D. Splitting: In the proposed method, the data is splitting in 80:20, because it's an Ideal splitting [16].

3.2.3 Model Development: This is the third phase of the proposed work; it involves developing the model using CNN, and DNN. The model has been trained using a training data set and tested by testing the data set. After the input layer, the next layer of the model is CNN. CNNs are ideal for time series analysis the model used it to predict the stock price over time. The third layer of the model is DNN; it is also useful for getting predictions with the help of past data. The past data is training the DNN and getting the share price prediction.

```

Total Number of Rows: 3301
Training Start Date: 2007-09-28
Training End Date: 2019-08-23
Number of Rows Used for Training: 2640
Validation Start Date: 2019-08-26
Validation End Date: 2022-05-02
Number of Rows Used in Validation: 661
Ratio of Rows Used in Training: 0.7997576491972129
Ratio of Rows Used in Validation: 0.20024235080278704
Training Accuracy (R-squared): 0.9971777414652634
Validation Accuracy (R-squared): 0.9748483276160194

```

Figure 4. Data description for training and testing

Figure. 4 represents the total no of rows, rows used for training and validation, training start and end date, start and end date of validation, training, and validation accuracy.

Ensemble CNN-DNN model algorithm

Input: Preprocessed dataset from BANKNIFTY/ YFinance

Output: Accurate share price prediction

EnsembleCnnDnn(data)

For f features in data do

C=Cnn(f)

P=Dnn(C)

Calculate Precision, Accuracy, Recall

MSE(P)

return P

End For

End EnsembleCnnDnn

3.2.4 Model Evaluation: This is the fourth phase of the work; it involves evaluating the model on behalf of the performances. The model has been used using a testing dataset and the performance metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) are used to evaluate the performance of the model [17].

3.2.5 Prediction: Now using this trained model the stock price is predicted. If the model used 15 years of data and splitting 80% for training and 20% for testing then it can predict the same as per the proposed method i.e., on behalf of past data the model can always predict the future data up to the next 20% coming time period in the future.

IV Implementation and Result Analysis

4.1 Implementation

Python is used for the implementation of the proposed method. For using deep learning models CNN and DNN Keras and TensorFlow library are used [18]. Metplotlib and Seaborn are used for data visualization. Pandas, NumPy, and Scikit-learn are used for data handling. The experimentation is performed on an Intel(R) Core(TM) i5-2GHz CPU with 8GB RAM. The experiment was also performed on Google Colab when GPU was needed.

The yfinance library's Banknifty Trading days data from 28th September 2007 to 2nd May 2022 used, this data contains total rows or can say total trading days as 3301 out of which 80 data i.e., 28th September 2007 to 23rd August 2019 used for training this data has 2640 rows. And 20% of the data

i.e. 26th August 2010 to 2nd May 2022 The dataset consists of 661 rows, used for testing and the model's accuracy is 97.48%.

The epoch process is represented in figure below.

```
Epoch 1/50
44/44 [=====] - 9s 93ms/step - loss: 0.0123 - val_loss: 0.0014
Epoch 2/50
44/44 [=====] - 3s 63ms/step - loss: 9.1255e-04 - val_loss: 0.0011
Epoch 3/50
44/44 [=====] - 2s 55ms/step - loss: 7.6129e-04 - val_loss: 8.4991e-04
Epoch 4/50
44/44 [=====] - 2s 51ms/step - loss: 6.7954e-04 - val_loss: 8.5456e-04
Epoch 5/50
44/44 [=====] - 2s 51ms/step - loss: 6.6628e-04 - val_loss: 9.9024e-04
Epoch 6/50
44/44 [=====] - 2s 51ms/step - loss: 5.8186e-04 - val_loss: 7.5233e-04
Epoch 7/50
44/44 [=====] - 2s 51ms/step - loss: 5.9139e-04 - val_loss: 7.2585e-04
Epoch 8/50
44/44 [=====] - 2s 54ms/step - loss: 5.6971e-04 - val_loss: 0.0013
Epoch 9/50
44/44 [=====] - 2s 54ms/step - loss: 5.1169e-04 - val_loss: 0.0016
Epoch 10/50
44/44 [=====] - 2s 54ms/step - loss: 5.2365e-04 - val_loss: 0.0011
Epoch 11/50
```

Figure 5. Epoch process

Figure. 5 represents the epoch process of the proposed work. It represents a number of epochs, time stamps, and validation losses in each epoch. The example Epoch process is represented with duration, loss, and number of epochs.

The Index/Share price prediction of Bank Nifty using a hybrid network of CNN and DNN achieved an accuracy of 97.48%. The method aimed to predict the stock prices of Bank Nifty (NSE Bank) by using historical trading data. The method utilized a hybrid network of CNN and DNN to extract features from the data and make accurate predictions. The proposed model was trained on a large dataset of historical stock trading data and was able to learn the underlying patterns and trends in the data. The method used several evaluation metrics to assess the performance of the model. The achieved accuracy of 97.48% indicates that the model was successful in accurately predicting the stock prices of Bank Nifty. These hybrid model findings demonstrate the effectiveness of using a hybrid network of CNN and DNN in stock price prediction and the importance of historical data in developing accurate prediction models. The results can be used by investors and financial analysts to make informed decisions and better understand market trends and patterns.

4.2 Result analysis

Actual Movements of Nifty, Sensex, and Bank Nifty: With the help of Python's Data Analytics ability, the line graphs for Sensex, Bank Nifty, and Nifty are generated. The description of the same is as follows:

Sensex: The upper graph represents the closing prices of the Sensex index from May 1, 2018 to May 4, 2022. The Sensex is plotted in purple. The chart shows the movement of the Sensex index over a given period of time, indicating changes in closing prices. The x-axis represents the date and the y-axis represents the closing price of the Sensex. It provides a visual representation of the performance of the Sensex over a given period.

Percentage return for Sensex (2018-05-01 to 2022-05-04): 61.97% The percentage return for the Sensex is calculated by comparing the closing price on the opening date (May 1, 2018) with the closing price on the closing date (May 4, 2022). The calculated percentage return is 29.45%, indicating the overall percentage increase in the Sensex over the 4-year period.

Nifty 50: The lowest order line graph represents the closing prices of the Nifty 50 index from May 1, 2018 to May 4, 2022. The Nifty 50 is plotted in blue. The chart shows the movement of the Nifty 50 index over a given period of time, indicating changes in the closing prices. The x-axis represents the date and the y-axis represents the closing price of Nifty 50. It provides a visual representation of the performance of Nifty 50 over a given period.

Percentage return for Nifty 50 (2018-05-01 to 2022-05-04): 59.26%

The percentage return for Nifty 50 is calculated by comparing the closing price on the start date (May 1, 2018) with the closing price on the end date (May 4, 2022). The calculated percentage return is 49.27%, indicating the total percentage increase of the Nifty 50 over a 4-year period.

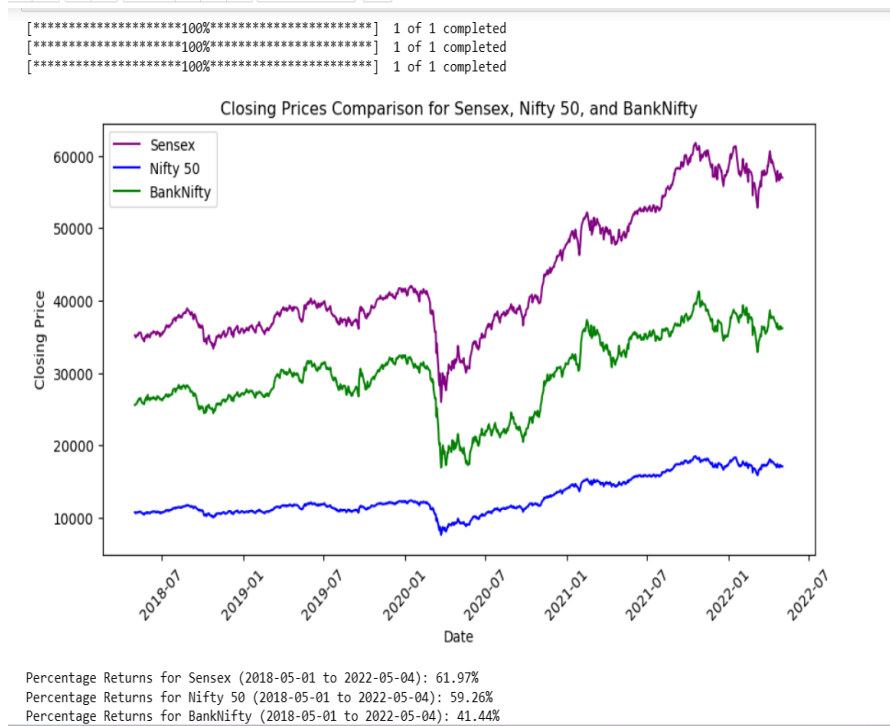


Figure 6. 5 Years Movements and Returns of Sensex, Nifty and Bank Nifty

Figure. 6 represents the 10 Years Movements and Returns of Sensex, Nifty and Bank Nifty. Bank Nifty: The middle line graph represents the closing prices of the BankNifty index from May 1, 2018 to May 4, 2022. BankNifty is plotted in green. The graph shows the movement of the BankNifty index over a given period, indicating changes in closing prices. The x-axis represents the date and the y-axis represents the closing price of BankNifty. It provides a visual representation of BankNifty's performance over a given period. Percentage returns for BankNifty (2018-05-01 to 2022-05-04): 41.44%. The percentage return for BankNifty is calculated by comparing the closing price on the start date (May 1, 2018) with the closing price on the end date (May 4, 2022). The calculated percentage return is 73.51%, which indicates the total percentage increase of BankNifty over the 4 years. These percentage returns represent the overall performance and profitability of each index over a specified 4-year period.

Actual V/S Training and Actual V/S Prediction

We have got this graph as a result after taking 15 Years of data i.e. start_date = '2007-09-28' end_date = '2022-05-04' from yfinance Library out of which 80% of data is used for turning and 20% Data is used for testing resulting in 97.48% accuracy in Actual V/S Prediction.

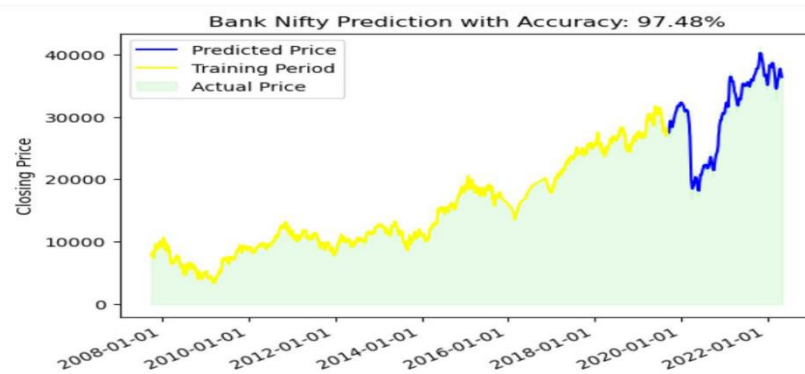


Figure 7. Actual V/S Training and Actual V/S Prediction

Figure. 7 shows the Actual V/S Training and Actual prediction. In Figure. 7, the yellow Line graph is the Training Period which is 80% time of the total duration taken. The Light Green Area graph is the Actual movement of Bank Nifty throughout the total duration and the Blue Line graph shows during the testing period as a predicted Line graph.

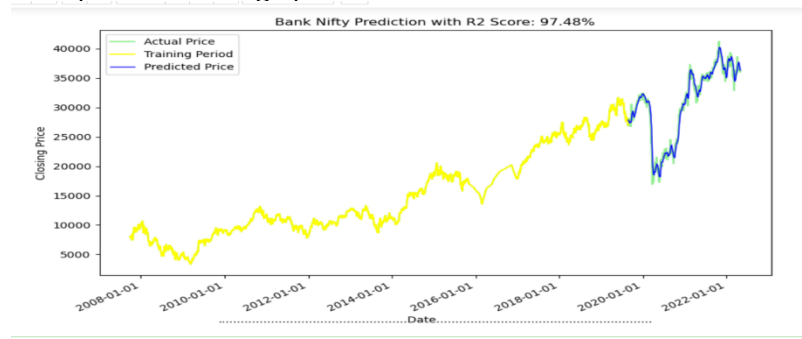


Figure 8. Actual V/S Training and Actual V/S Testing

Figure. 8 represents Actual V/S Training and Actual V/S Testing in which the Green Line and Blue Line are Actual vs Predicted graphs and the Yellow Line graph is the training Line graph. The training period is from start_date = '2007-09-28' end_date = '2022-08-23'.

Testing- Actual V/S Predicted:

Figure. 9 Line graph shows the Testing- Actual V/S Predicted on 80% of taken data i.e. from 2019-08-26 to '2022-05-04'. In the Green Line graph shows the actual movement of Bank Nifty during 20% of the total duration the Blue Line graph shows the Predicted line graph with 97.48% of accuracy and R-squared=0.974848

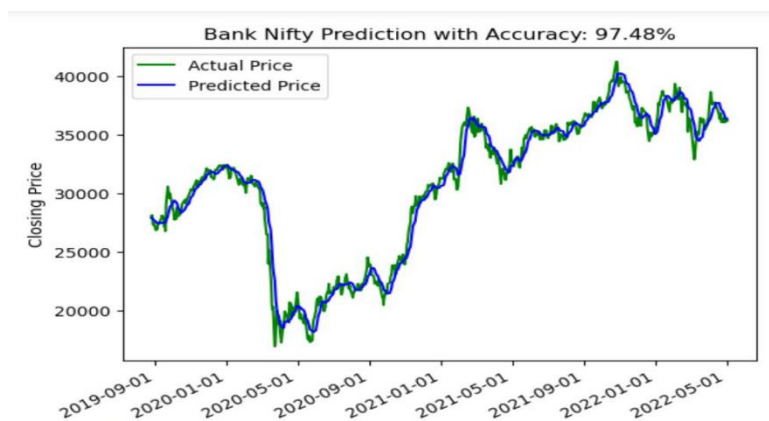


Figure 9. Testing- Actual V/S Predicted Line Graph

Validation Loss:

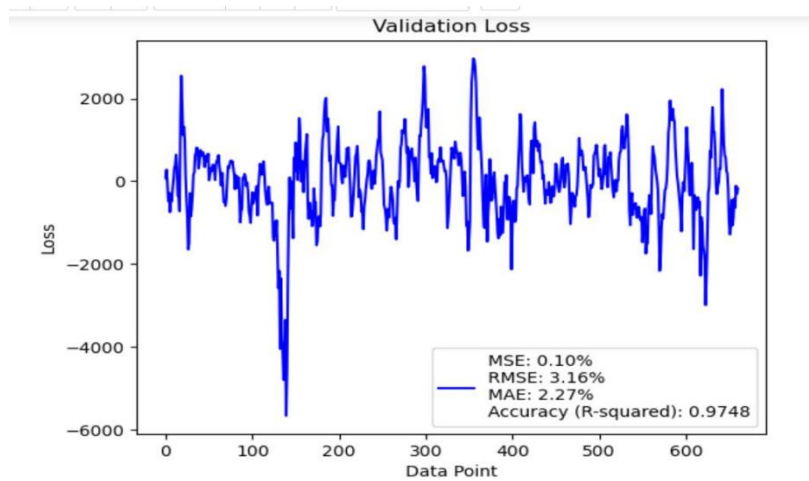


Figure 10. Validation Loss

Figure. 10 represents the validation loss during the testing period. As per the fig, 10 the MSE is 0.10%, RMSE is 3.16%, MAE is 2.27% and R-squared is 0.9748%.

Evaluation Metrics: The Evaluation Metrics consists of training loss, validation loss, training and validation accuracy of MSE, RMSE, MAE and R-squared.

```

Training Loss (MSE): 0.07068396332442098 %
Validation Loss (MSE): 0.0999814561486119 %
Training Loss (RMSE): 2.6586455823298634 %
Validation Loss (RMSE): 3.1619844425393984 %
Training Loss (MAE): 1.9755309561750818 %
Validation Loss (MAE): 2.273714829994833 %
Training Accuracy (R-squared): 0.9971777414652634
Validation Accuracy (R-squared): 0.9748483276160194

```

Figure 11. Evaluation matrix

Figure. 11 represents the training and validation loss and RMSE, MSE and MAE values of CNN-DNN hybrid method. As from figure training and validation loss are decreasing after epoch. The evaluation matrix represents the metrics with their values. As from the above results we can find the MSE Loss, RMSE Loss, MAE Loss in % during training and validation and the most important R-Squared Accuracy during the training and validation i.e. 99.71% (Training Accuracy) and 97.48% Testing Accuracy.

Closing Price V/S predicted Price:

Date	Close	Predicted
2019-08-26	27951.349609	27875.690276
2019-08-27	28126.150391	27845.072355
2019-08-28	27804.300781	27852.603029
2019-08-29	27305.199219	27781.102798
2019-08-30	27427.849609	27702.076978
...	...	...
2022-04-26	36404.800781	36821.140628
2022-04-27	36028.851562	36662.423211
2022-04-28	36422.199219	36529.717258
2022-04-29	36088.148438	36391.999201
2022-05-02	36163.750000	36335.394851

Figure 12. Closing Price V/S predicted Price

Figure 12 below shows the first 5 rows of the closing price V/S predicted price. As from figure 12 close price are very much closed to predicted price.

Comparison: In the following table shown parameters are used in prediction.

Table 1: Parameters Used in Prediction

1	Unit= 64	Drop out:0.2	Activation: Sigmoid	Loss=binary_cross sentropy	Optimizer=adam	Epoch 30
2	Unit =64	Drop out:0.3	Activation: Relu	Loss=binary_cross sentropy	Optimizer=adam	Epoch 35

3	Unit= 64	Drop out:0.1	Activation: Softmax	Loss=binary_cross sentropy	Optimizer=adam	Epoch 40

Table 1 represents the parameters used in the prediction such as unit as 64, drop-out values from 0.1 to 0.3, activation functions as Sigmoid, Relu, and Softmax, loss as binary_cross_sentropy, optimizer and adam, and epoch from 30-40.used in the implementation.

4.3 Model Evaluation

Accuracy = (TP+TN) / (TP+TN+FP+FN)

Precision = TP/(TP+FP)

Recall= TP/(TP+FN)

Where: TP=True Positive, FP=False Positive, T=True Negative, FN=False Negative

F1= (2*Precision*Recall) / Precision + Recall

Table 2: Comparison between the Existing Model and the Proposed Model

Method	Accuracy	Precision	Recall
LSTM [6]	83.88	83.21	79.25
RNN [10]	91.23	88.25	82.12
DNN [10]	96.22	93.88	90.25
CNN [10]	92.31	90.25	89.11
Proposed CNN DNN	97.48	98.21	91.02

Table 2 represents the accuracy, precision, and recall of various methods and the proposed method.

F1 = (2*98.21*91.02) / (98.21+91.02) =94.47

F1 Score for Proposed Model = 94.47

From the results proposed method accuracy is 97.48, precision is 98.21 and recall is 91.02 which is better as compared to other traditional methods.

V. Conclusions and Future work

Predictions should be always reliable because from the wrong prediction an investor has to bear a huge loss in the market and it may affect the planning and economy. The paper proposed ensemble CNN and DNN for prediction of stock market. The ensemble model combines the strengths of deep learning models CNN and DNN, and developed a comprehensive methodology for the prediction of stock/index prices on Banknifty (NSE Bank), a highly volatile Indian sectorial Index that represents 12 major banks of the country. The CNN layer is used to extract features from input data related to an estimated future value. DNN layer is used to combine features learned from the CNN layers. The result shows that the method provides high accuracy and reliability as compared to other state-of-the-art models. The achieved accuracy of 97.48% indicates that the model was successful in accurately predicting the stock prices of Bank Nifty.

Later in the future for better reliability and accuracy prediction of all 12 banks individually on behalf of past movement data, fundamentals of the company, economic factors, news, and events with the help of a hybrid network of CNN, RNN, LSTM, and DNN will be performed. After that these 12 Banks' prices and market caps will come, on behalf of these prices and market caps and can be calculate as per NSE Bank (Bank Nifty) spot as per NSE Index construction rules. These concepts can apply on any Index and its participants as on Sensex 30, Nifty 50, Dow Jones 30, Nikkei 225 and on S&P 500.

References

- [1] A. Gupta, Akansha, K. Joshi, M. Patel and V. Pratap (2023) "Stock Market Prediction using Machine Learning Techniques: A Systematic Review." International Conference on Power, Instrumentation, Control and Computing (PICC): pp. 1-6.
- [2] K. V. Kumar and R. Anitha (2022) "A Detailed Survey to Forecast the Stock Prices by Applying Machine Learning Predictive Models and Artificial Intelligence Techniques." International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS): pp. 1-6.
- [3] Zahra Fathali, Zahra Kodia&Lamjed Ben Said (2022) "Stock Market Prediction of Nifty 50 Index Applying Machine Learning Techniques.", Applied Artificial Intelligence, Taylor & Francis **36(1)**: 3100-3123.
- [4] Baek, S., Mohanty, S.K., & Glambosky, M. (2020) "COVID-19 and Stock Market Volatility: An Industry Level Analysis." Finance Research Letters: pp. 37-46.
- [5] Beckmann M. (2017) "Stock Price Change Prediction Using News Text Mining." Federal University of Rio de Janeiro: pp. 156-162
- [6] Choudrie, J., Banerjee, S., Kotecha, K., Walambe, R., Karende, H., & Ameta, J. (2021) "Machine Learning Techniques and Older Adults Processing of online Information and Misinformation: A COVID-19 Study." Computers in Human Behavior: pp. 119-121.
- [7] F. Ronaghi, M. Salimibeni, F. Naderkhani, and A. Mohammadi (2020) "ND-SMPF: A noisy deep neural network fusion framework for stock price movement prediction." IEEE Int. Conf. Inf. Fusion (FUSION): pp. 1–7.
- [8] Pushpendra Singh Sisodia, Ashish Gupta, Yogesh Kumar, Gaurav Kumar Ameta (2022) "Stock Market Analysis and Prediction for Nifty50 using LSTM Deep Learning Approach." International Conference on Innovative Practices in Technology and Management (ICIPTM): pp. 324-335.
- [9] O. O. Aalen (1989) "A linear regression model for the analysis of life times." Statist. Med **8(8)** : pp. 907-925.
- [10] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, and H. Cho (2021) "Xgboost: Extreme gradient boosting." R Package **1(4)**: pp. 389-401
- [11] Xiao Ding, Yue Zhang, Ting Liu, Junwen Duan (2015) "Deep Learning for Event-Driven Stock Prediction.", IJCAI : pp. 2327-2333.
- [12] Sreelekshmy Selvin, R Vinaykumar, E.A. Gopalkrishnan, Vinay Krishna Menon, K.P. Soman (2017) "Stock Price Prediction using LSTM, RNN and CNN-sliding window model.", IEEE : pp. 718-720.
- [13] <https://www.kaggle.com/competitions/g-research-crypto-forecasting/data>
- [14] Mehar Vijh, (2020) "Stock closing price prediction using machine learning techniques." International Conference on Computational Intelligence and Data Science: pp. 599– 606
- [15] Sheng Chen (2018) "Stock Prediction Using Convolutional Neural Network." IOP Conf. Series: Materials Science and Engineering: pp. 435-445.
- [16]. Jiaqi Pan, Yan Zhuang, Simon Fong (2016) "The Impact of Data Normalization on Stock Market Prediction: Using SVM and Technical Indicators." ResearchGate: pp. 216-225.
- [17] K.K Suresh Kumar (2012) "Performance analysis of stock market prediction using Artificial Neural Network." Global Journal of Computer science and Technology 12(1): pp. 735-742.
- [18] Abadi, M.v(2016) "Tensorflow: Large-scale Machine Learning on Heterogeneous Distributed Systems" IEEE: pp. 76-82.