Journal of Information Technology Management

Print ISSN: 2008-5893

Online ISSN: 2423-5059

Feature Selection and Hyper-parameter Tuning Technique using Neural Network for Stock Market Prediction

Karanveer Singh

School of Computer Science and Engineering, Galgotias University, Greater Noida, India. E-mail: karan.cps080798@gmail.com

Rahul Tiwari

School of Computer Science and Engineering, Galgotias University, Greater Noida, India. E-mail: rahultiwari.201307@gmail.com

Prashant Johri

School of Computer Science and Engineering, Galgotias University, Greater Noida, India. E-mail: johri.prashant@gmail.com

Ahmed A. Elngar

Faculty of Computers and Artificial Intelligence, Beni-Suef University, Beni-Suef City, Egypt. E-mail: elgnar_7@yahoo.co.uk

Abstract

The conjecture of stock exchange is the demonstration of attempting to decide the forecast estimation of a particular sector or the market, or the market as a whole. Every stock every investor needs to foresee the future evaluation of stocks, so a predicted forecast of a stock’s future cost could return enormous benefit. To increase the accuracy of the Conjecture of stock Exchange with daily changes in the market value is a bottleneck task. The existing stock market prediction focused on forecasting the regular stock market by using various machine learning algorithms and in-depth methodologies. The proposed work we have implemented describes the new NN model with the help of different learning techniques like hyperparameter tuning which includes batch normalization and fitting it with the help of random-search-cv. The prediction of the Stock exchange is an active area for research and completion in Numerai. The Numerai is the most robust data science competition for stock market prediction. Numerai provides weekly new datasets to mold the most exceptional prediction model. The dataset has 310 features, and the entries are more than 100000 per week. Our proposed new neural network model gives accuracy is closely 86%. The critical point, it isn’t easy with our proposed model with existing models because we are training and testing the proposed model with a new unlabeled dataset every week. Our ultimate aim for participating in Numerai competition is to suggest a neural network methodology to forecast the stock exchange independent of datasets with reasonable accuracy.

Keywords: Neural network, Stock market prediction, Numerai, NMR, Deep learning.

DOI: 10.22059/jitm.2020.79368

Introduction

The Prediction Stock market is an active area for the research with the help of stock market prediction, and we can quickly ascertain the upcoming values of corporate stock or economic data. So, why is stock market prediction being famous, the reason is that the investors accept that the main chance to invest in the market is the point at which it is going up. At the point when the market falls, such investors might want to remain away and return just when they are sure that the market rises again, and the better result will be available for them. Anticipating the market implies predicting how the financial exchange file moves. We can easily prognosticate the stock exchange with the method of machine learning models. In this approach, we can use a different model to get better accuracy and result in the prediction. For this prediction, we have used the NN methodology to predict the stock. We are using the real-time Numerai stock market dataset, which is an unlabeled dataset.

Numerai is a data science competition that boosts the windbreak fund (Lee, Kim, Koh, & Kang, 2019). In this competition number of data, scientists take part in the competition. The Numerai competition has similarities like we are predicting the stock of a company that is not labeled. The data set quality is good enough, and the data which we use changes every week. So, we can predict the right amount of data, and then we can upload the prediction for checking the prediction on their model. As we get an excellent score for the prediction, they give rewards in the form of their cryptocurrency, which is the name as NMR. So, for this tournament, we have used the NN model.

Neural networks are the exotic models (Parmar & et al., 2018) over the past years, and these models used across an exceptional dimension of the domain. The neural networks can use in finance, medicine, engineering, image recognition, etcetera. A triple-layer NN was utilized in the universe. These layers help in finding the result of the problem. The neural network approach takes the set of inputs (features) and computes as output as a prediction. There are other models used for prediction analysis like the random forest, decision tree, support vector machine, xgboost, and soon. However, in our method, the NN approach is used because it works well for a variety of prediction problems and can easily compare the dataset. Predicting stock costs is an essential objective within the economic world (Al-Hmouz, Pedrycz, & Balamash, 2015; Bengio, 2009; Bagheri, Peyhani, & Akbari, 2014) since correct anticipation can yield massive money edges and hedge fund risks. With the sudden increase of the net and computational technology, the frequencies for acting effort on the forecasting of stock would increase to a time-specific manner (Son, Noh, & Lee, 2012; Kimoto, Asakawa, Yoda, & Takeoka, 1990).

The neural networks model is used to prognosticate the prediction of the stock market because of the ability to assimilate nonlinear mapping between inputs and outputs. The neural network is trained to perform a variety of financial-related tasks. In Numerai competition, the dataset which we use for the prediction is unlabeled, and it is a regression problem. With the help of the neural network, it is easy to map the features which are given in the dataset. The neural networks have the ability for nonlinear function approximation and information processing, which other models do not have. The neural network well described the issues in which the relationship among the data is robust, and the training data sets are large enough.

In this problem, there are many different resources from which we can get free data like Yahoo finance. In any case, most financial stock exchange information shockingly isn’t openly accessible. Utilizing just a Yahoo finance dataset to build a model resembles utilizing just a single pixel in a picture to figure out how to perceive handwritten digits. The high quality of financial stock exchange data is protected by hedge funds and data monopolies. The excellent quality of datasets will become more private and more costly. For the stock market prediction problem, there is no free, excellent quality, and public dataset for machine learning.

So, forgetting the high-quality dataset we have used the dataset from the Numerai tournament. Numerai is attempting to make an ML algorithm that can most precisely time and foresee the market. The manner in which they are doing this is by having data scientists test our prediction on the Numerai model and then they provide the reward in the form NMR. If the prediction is good then the Numerai hedge fund will grow quicker.

In this topic many researchers did the analysis with the dataset which is not so good quality dataset and they have not used the live dataset of the stock market. Our approach is on the high-quality dataset and the dataset which we used is changed every week. With this help we can test our model on every week prediction which helps us to check if the model is performing good on the dataset or not. The predictions which our model predicts are uploaded on the Numerai tournament for checking the predictions are performing well on their model also.

Related Work

Related works in this stock market sector, we separate the methods which help to compute the prediction of stock market issues. The first class of connected procedure is economic science simulation, which has a Hellenic economic science hypothesis for prediction. Common ways are the auto-regressive technique (AR), the MA, the ARMA, and, therefore, the auto-regressive-integrated-moving-average (ARIMA) (Lathuiliere, Mesejo, Alameda-Pineda, & Horaud, 2020). In other words, this hypothesis takes every untested sign as a strident attenuated haplotype of the previous signs and noisy freelance sets. However, many of the people think about some sturdy assumptions in which reference to the noisy term's loss functions. In the second class, we have soft computational based models. Soft computational may be a term that covers computing that mimics biological processes. These techniques embrace NN methods, SVM, PSO, and some others. Several researchers have tried to agitate opacity beside randomness in possibility pricing models.

Dang Lien Minh, Huy, Min, & Moon (2018) has proposed the model TGRU for the prediction of stock in which they have used the dataset between October 2006 and November 2013, which they get from the yahoo finance. They conducted implementation on the NVIDIA kit with Keras interface version 1.2.3 involves Python version 2.7. The hyperparameter they used in the model was done training by 35 epochs, collection size n = 64, imprinting rate = 0.0001, and imprinting rate decay t = 0.00001. They got an overall accuracy of about 66.32% on the proposed model.

Yujie Wang has proposed the Hybrid Time-Series NN (HTPNN) hypothesis for the Conjecture of stock exchange. They have used the yahoo finance dataset. In the HTPNN model, they have used the 2-convolution layer, 2-LSTM layer, learning rate 0.005, and they have used 1000 iteration. The experimental result showed that the model got 69.51 Accuracy (Wang, Liu, Guo, Xie, & Zhang, 1970).

Idrees, Alam, & Agarwal (2019) has proposed the ARIMA model, which is a univariate method of predicting future values of time series data for the Indian Stock market; they have used the stock market data of India from Jan 2013 to Dec 2017. The performance factor in assessing the stock market performance done by index and the Indian market has two main indexes known as Sensex (contains 30 stocks) and Nifty (contains 50 stocks). AR(p) represents the "autoregression" model influenced by a variable 'p'. Then to take the error term in the AR model into consideration, the "moving average" MA(q) model has used, which considers the error of the previous model, but this makes the task difficult to fit the model onto time-series data. Finally used the ARIMA model, which is much preferable for stationary time-series data. After further decomposition of data and further analysis, the L-Jung Box Test has applied results p-value for the Nifty and Sensex equal to 0.9099 and 0.8682, respectively.

Hiransha, Gopalakrishnan, Krishna Menon, Soman (2018) has performed his research work with the dataset collected from highly traded sectors of banking, automobile, and IT sector. The training was conducted on data collected from 1 January 1996 to 30 June 2015, and it contains the closing price and normalizing to a standard range. They examined four diverse proposed models for the stock expectation, and the models utilized are Multilayer Perceptron (MLP), RNN and CNN and Long-Short-Term- Memory (LSTM) for dataset collected from monetary exchange NSE and NYSE. Their CNN model performs wells against the other three models.

Liu, & Wang (2019) in his paper experiment to predict the stock values by extracting the hidden analytical information from the news. They used two sources to obtain the dataset for their research: one is Standard & Poor's 500 (S&P500). and China Security Index 300 (CSI300). The hidden stock trend information data extracted from news helps in obtaining the effective distribution of a selection of news. The social-oriented methods have used from social sites from tweets and web. By applying sentiment analysis techniques, different weights assigned to input sequences of sequence-to-sequence models for improving the attention-based method in the decoding process. Dual-information encoder was used to process the news accordingly and numerical data & for the statistical data, a nonlinear function to encode them into a relating binary vector. A Numerical Based Attention (NBA) model in which they have used the LSTM as a hidden layer. For the better result, the LSTM coders set to 64.

Ren, Wu, & Liu (2019) used the analysis of sentiments and vector machines for conjecture stock exchange. They have used the dataset from the China SSE 50 index, which is not only used for the stock exchange still for the latest dataset. They have used the fivefold cross-validation and a twisting window method. They used the SVM model.

Selvamuthu, Kumar, & Mishra (2019) proposed the neural network model for the Indian financial exchange. They used the one by one dataset and a 20-min dataset for the stock prediction. In which they have used the three algorithms, by forecasting the data for the financial exchange.

Gunduz, Cataltepe, & Yaslan (2017) used the Regression and LSTM methodology on Learning to conjecture the financial values. The dataset obtained from yahoo finance. In the Regression-based model, they have used the batch size 515 and 95 epochs. For the LSTM based model, they dropout 0.3 and used the RMSE. The confidence score of 0.86625 for the regression-based model.

Khan, & et al. (2019) used the learning algorithm for conjecture financial exchange on various analyses like public and politics. They have used data from various public and other sectors and sentiment data they have used from Twitter. They have used many algorithms like DT, SVM, RF, MLP, but the two algorithms give a better result than other algorithms. The

MLP and DT give a better result for the stock market. They achieved an accuracy of up to 68%.

Stoean, Paja, Stoean, & Sandita (2019) proposed the LSTM and CNN model for the stock market. They have used the dataset from Bucharest Exchange, which has the data of the 25 companies for the financial exchange. In the LSTM, they have used 50 units in each layer, and dropout set to 0.2. They have tried the model with 2 and 3 layers to find a better result. The LSTM performed well then CNN for their dataset.

Nguyen, & Yoon (2019) has proposed the DTRSI model. The DTRSI stands for Deep Transfer with Related Stock Information framework, which performs well then, the SVM, RF, and KNN model for stock prediction. Here the LSTM methodology used with the input-layer has same number of features and 20-time steps, two LSTM layers with 16 units, and dropout to 0.5. In the output layer, it uses the one sigmoid activation unit. The dataset used is from the stock market indices, i.e., the S&P 500 and the KOSPI 200 from 31 July 2012 to 31 July 2018. The further portion of the document arranged in the following method. In the next sections, details discussed the proposed method and experimental result analysis.

Abdulsalam and et al. (2011) Stock prediction analysis has also extracted by using the method of regression analysis. This paper shows data mining step which involves Knowledge Discovery in Databases (KDD) on data using data mining software tool they build their database based of information obtained from the daily summary of 18 months published by Nigerian Stock Exchange.

Yuan, Yuan, Jiang, & Ain (2020) they have proposed for the China stock market by using ML algorithms like random forest, SVM, and ANN model, also they have used the feature selection approach for better features. The dataset they have used is 8-years data of the Chinese share market. The random forest model performed well among the other models.

Ghosh, Neufeld, & Sahoo (2020) In this experiment using the methodology of Fischer & Krauss (2018) and Krauss et al. (2017), they divided the dataset collected from January 1990 to December 2018, through which they found overlapping periods in the dataset. So, to obtain various insights, they applied techniques for feature generation like LSTM & Random Forest. Also, for the sake to make the model robust against outliers, the feature normalization technique viz. Robust Scaler standardization applied, what is more, thus expels the middle and afterward scales the data utilizing the interquartile range. Since LSTMs is very tedious and to proficiently use the intensity of GPUs, they performed a test utilizing CuDNNLSTMs with parameters like categorical cross-entropy as Loss function, optimizer used is RMSProp with the learning rate of 0.001, and a batch size of 512. In their setting, LSTM performs much better than random forests, which is as per the results of Krauss and Fischer (2018), demonstrating that LSTM has a touch of advantage diverged from the memory-free strategies separated in Krauss et al. (2017).

Chong, Han, & Park (2017) in his research paper, has used the deep learning neural-network for the stock market analysis. In which they used the PCA, Autoencoder, Restricted-Boltzmann-Machine, and 3-layer deep neural network model for the KOSPI dataset, which collected from the stock market in South Korea. The dataset consists of the stock prices, which collected in the interval of 4 Jan 2010 to 30 Dec 2014. There deep neural network model performed very well on the KOSPI dataset.

Khan and et al (2020) used the machine learning algorithms on the dataset, which contains the social media and financial news for understanding the behavior of the investors. They have used the HP Inc dataset for the stock analysis. They have used the feature selection technique to get the useful features for the review, and they also used a machine-learning algorithm like SVM, Gaussian Naïve Bayes, LR, K-nearest neighbor, and Random forest. However, the random forest model performs well among all the other models. In the random forest, they have used some parameters like n_estimators to 20 and random_state to 123, and the min_sample_split 5 helps to perform well on the HP Inc. dataset.

Parmar and et al. (2018) they have used the LSTM model and regression model for the stock prediction. They have used the dataset from the Yahoo Finance in which they have 9 lakh records of stock price values and other relevant values. In the regression model, they have used the linear regression model with parameters like batch size 512 and epochs set to 90, and for the LSTM model, they have used two LSTM layers with dropout 0.3 and output value with 256. The LSTM model performed well on the Yahoo Finance data, and they got the MSE value to 0.00875.

Selvin, Vinayakumar, Gopalakrishnan, Menon, & Soman (2017), with his team, performed research with the dataset gathered from July 2014 to June 2015, which consists of the minute wise price for 1721 companies listed in NSE. Sliding-window-approach executed for a prediction of short term near future. The size of windows set to 100 minutes with 90 minutes of overlap, and this setup was able to result in the prediction of a 10-minute interval. The preparation happened for models with 1000 epochs by shifting the size of the layer for fine-tuning. On the off chance that the loss (MSE) for the epoch is not precisely the worth acquired in the past epoch, the weight grids for that epoch put away. The CNN model performs much accurate than the other two models viz: LSTM and RNN are for the reason that CNN takes information from the current window, not from the previous predictions. In the case of CNN, it predicted with a smaller error percentage as compared to others.

Patel, Shah, Thakkar, & Kotecha (2015) in his paper tends to issue of forecasting the significance of the evolution of the stock and stock value record for Indian-stock-markets.

The investigation looks at four expectation models, ANN, SVM, Naive-Bayes and Random Forest, with two methodologies for contribution to these models. The main aim of this research is to successfully assess the future direction of movement for stocks and their indices. Ten years (2003–2013) of correct information of S&P BSE Sensex, CNX Nifty, Infosys Ltd. moreover, Reliance Industries from Indian stock markets. The exactness of 89.33%, 89.98%, 86.69%, and 90.19% is accomplished by ANN, SVM, random forest, and naive-Bayes (Multivariate Bernoulli Process) separately.

Bao, Yue, & Rao (2017) utilized a denoising approach utilizing WTs for data pre-processing and afterward used stack autoencoders (SAEs) to get better features. Their proposed structure beat their standard techniques as far as MSE. However, the more significant part of the past studies planned for finding the appropriate structure that can gain from non-linear and stock data dependent on assumptions about adequate information, though, gathering increasingly stock data is strict with just around 240 datapoints focuses accessible in a year. Meanwhile, there work targets to find a better methodology that can train a LSTM model regardless of having not much data.

Qiu, Wang, & Zhou (2020), in their paper, builds a production skeleton to predict opening stock prices. Data of stocks accumulated from three different stock indices: Dow-Jones-Industrial Average (DJIA), Hang-Seng index (HSI) and S&P 500. The data obtained is very noisy, unstable & complex, so wavelet transform was taken into action since it is much suitable for unconventional financial series since it can perform both time domain and frequency domain analysis. Results obtained are excellent on applying an attention-based LSTM neural network on processed data from the wavelet transform. The observed outcomes show that compared with the generally employed GRU, and LSTM NN models with wavelet transform, the proposed model has a superior fitting degree and improves precision of the expected results. In this manner, the model has broad application likelihoods and is profoundly dangerous with existing models.

Babu, Geethanjali, & Satyanarayana (2012) published their research, which breaks down the significant clustering algorithms: K-Means, reverse K means and Hierarchical clustering algorithm and think about the presentation of these three considerable clustering algorithms on the part of accurately class keen group building capacity of the algorithm. A powerful clustering strategy, Hierarchical agglomerative and Recursive K-means clustering (HRK) proposed, to foresee the transient stock value developments after the arrival of budgetary reports. The proposed approach comprises three stages. To begin with, we convert each money related description into a component vector and utilize the hierarchical-agglomerative-clustering strategy to partition the changed over component vectors into groups. Second, for each bunch, they recursively used the k-means clustering technique to parcel each group into subgroups with the goal that most element vectors in each subcluster have a place with a similar class. At that point, for each sub bunch, they pick the centroid as the agent highlight vector. At long last, we utilize the agent highlight vectors to foresee the stock value developments. The trial result shows that the proposed strategy beats support vector machine as far as exactness and standard benefits.

Kusuma, Kao, & Hua (2019) prepared and assessed their experimental model on two diverse stock markets. Dataset has gathered from 50 organization stock markets for Taiwan and ten organization stock markets for Indonesia as a top stock market in the two nations. Break down the connection between various period times; data changed over it into a candlestick diagram using library Matplotlib. At last, a CNN learning algorithm employed to develop our forecast for the stock market. They found that the Convolutional Neural Network can locate the unknown design inside the candlestick graph pictures to gauge the development of the particular stock market later on, and at last, their analysis accomplished 92 % for accuracy.

Proposed Method

We have performed on the proposed work in windows 10 with Intel Pentium configuration. The project was done with the help of the Google Colab server for better computational power. In the Google Colab server, we have used the GPU as the runtime for running the code efficiently. We have used some libraries such as TensorFlow version 1.x, Pandas, NumPy, Seaborn, Matplot library, Sklearn library, and Keras.

A. Dataset Description

The dataset which we have used in this study is from Numerai, which we get is in the form of an unlabeled dataset. Numerai provided two datasets, and one is a training dataset used for training to our proposed model, and next is the testing dataset used for the testing model. The dataset which we use is changing every week. We have to upload our prediction on the Numerai tournament to check the prediction is working well on their model also. Table 1 shows a detailed explanation of each feature in our dataset. The training dataset has 314 columns consists of ids, eras, datatype which trained and 310 features which subdivided into different groups like intelligence1 to intelligence12, wisdom1 to wisdom46, charisma1 to charisma86, feature dexterity1 to feature dexterity14, feature strength1 to feature strength38, feature constitution1 to feature constitution114 and one target name as target_kazutsugi the values are like 0, 0.25, 0.50, 0.75 and 1 which used to train the model. The datasets about 558069 rows with different ids and eras. Heatmap represents one variable that could connect with another variable. It will be giving more effective outputs for investigations and displays more readily between factors. Figure 1 represents the correlation between 50 features in the Numerai dataset. With the help of heatmap, we can easily visualize the correlation between the features, and this made with the help of the Seaborn library. The dataset contains 314 attributes; to represent 314 columns in the heatmap, it looks messy. The tournament dataset is our test dataset, which also has the 314 columns with the same attributes. However, in the data type, it has three different types as validation, test, and live. There is no need for pre-processing in this dataset because there no missing values and outliers are present.

Table 1. Details of each feature in the Numerai dataset

Variables		Class	Scale
id	key of prediction	Categorical	Random values
era	Period of time.	Categorical	(era1 to era120)
data_type	Type in the datasets	Categorical	(Train, Test, Live, Validation)
feature_intelligence1-feature_intelligence12	Feature set 1	Numerical	(0, 0.25, 0.50, 0.75, 1)
feature_charisma1 -feature_charisma86	Feature set 2	Numerical	(0, 0.25, 0.50, 0.75, 1)
feature_strength1 -feature_strength38	Feature set 3	Numerical	(0, 0.25, 0.50, 0.75, 1)
feature_dexterity1-feature_dexterity14	Feature set 4	Numerical	(0, 0.25, 0.50, 0.75, 1)
feature_constitution1-feature_constitution114	Feature set 5	Numerical	(0, 0.25, 0.50, 0.75, 1)
feature_wisdom1-feature_wisdom46	Feature set 6	Numerical	(0, 0.25, 0.50, 0.75, 1)
target_kazutsugi	Target	Numerical	(0, 0.25, 0.50, 0.75, 1)

B. Feature Selection

Feature Selection is a process in which we select those features which gives a good impact on the model. If we have the irrelevant features in the dataset it will decrease the accuracy of our model. The feature selection is widely used for selecting the best feature from the dataset which helps to increase the accuracy of the model. It has advantages like reducing overfitting, improves accuracy of the model and it also reduces the training time of the model so the algorithm can only use the good features for training the model. For selecting the features there are some approaches like univariate analysis, bivariate analysis and correlation matrix using heatmap. For our problem we have used the heat map which is explained in the below paragraph.

With the help of feature selection technique, we will compare the relation between features with the help of heat map, which shows the correlation between the features. So, after using this, we can visualize the features by using a seaborn library which helps in generating heat maps. So, we will take only those features which have a correlation of less than 0.9 and we will remove those features which are having more correlation then the 0.9, as shown in figure 2. Now we will select the columns based on the p-value. We have used the backward elimination process which has the small regression model and then we can calculate the p values. If the value is higher than the fixed p-value, we can reject the features which have the least p-value. The fixed p-value which we used is 0.05. After removing the values, we can visualize the values with the help of the violin plot, as shown in figure 3.

$$P - Value < 0.05 \ \rightarrow \ Moderate\ certainty\ in\ the\ result.$$

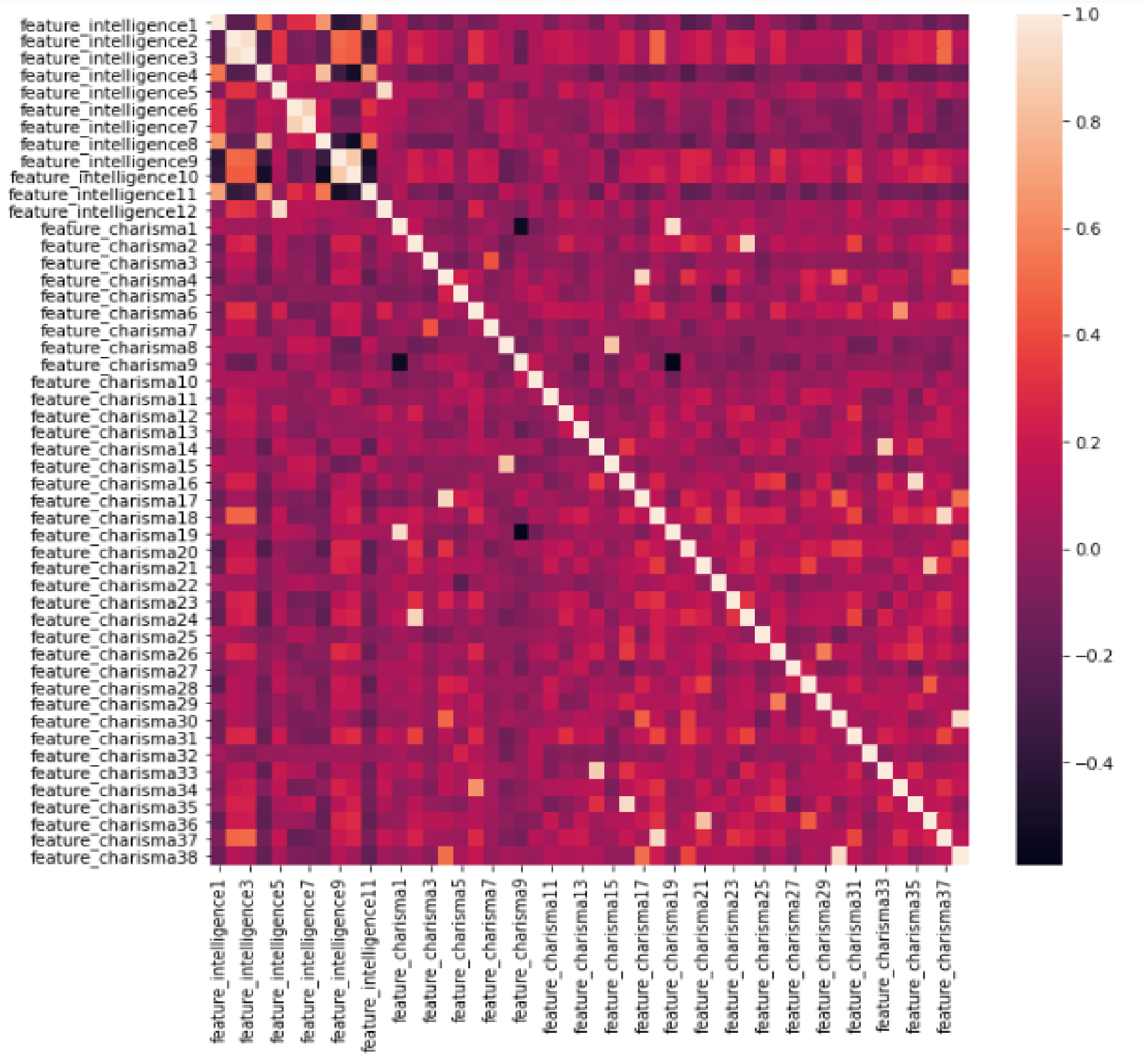


Figure 1. Correlation between features represented in heat map

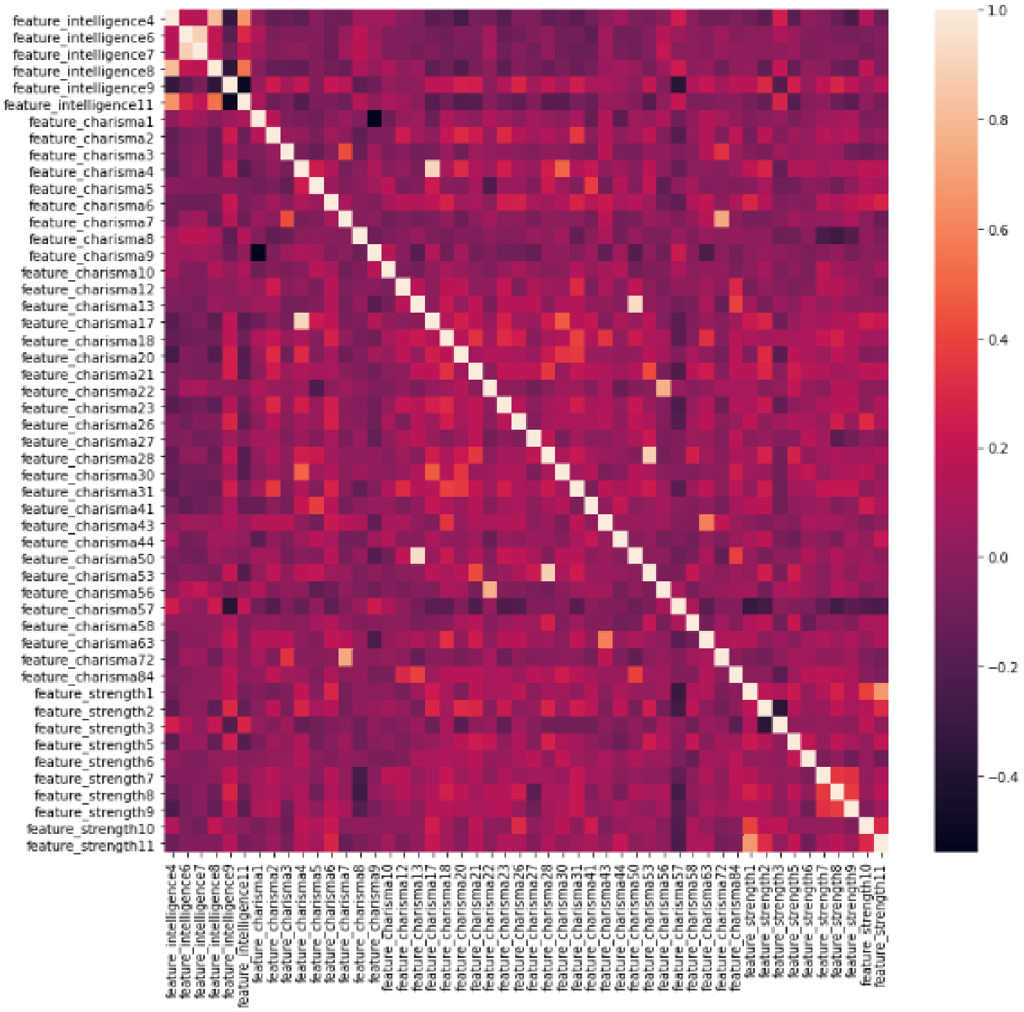


Figure 2. Heat map for correlated values less than 0.9

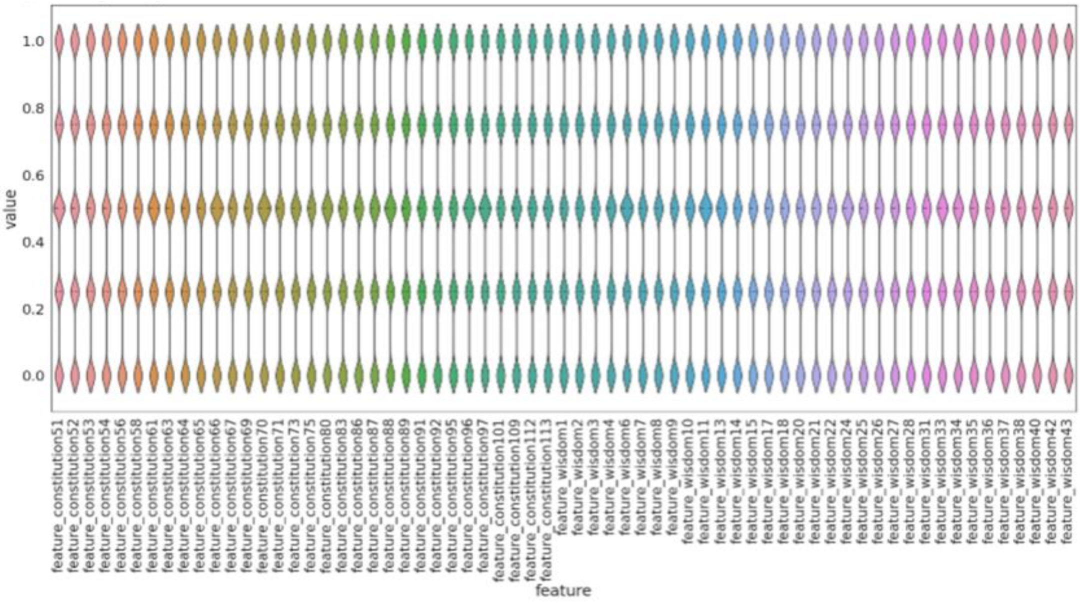


Figure 3. Visualization of selected 167 features data entries

C. Methodology

NN is to instantiate nonlinear relation to the data given as a predefined value, which makes them unique for modeling the systems called forecasting of financial exchange. In the neural network, we have used the Relu and Linear activation function. We can also use any activation function in the neural network like unipolar, bipolar, hyperbolic, and radial functions. Hidden patterns and nodes in the NN are trained for forecasting of data related to stock that has several various layers as described. Using Keras library for the neural network, we have used python language for the implementation of the model, which we can train on our training dataset for getting a good result on the test dataset. In the neural network model, we have used 400 neurons; dropout is 0.4 and batch normalization in the second layer of the neural network model. The term batch normalization is a reference for training very deep NN that standardizes the contribution to a layer for every minimum batch. It has the impact of settling the learning procedure and drastically diminishing the number of epochs used for training a deep neural network. So, with the batch normalization speed of training, the model is increased.

This how the batch normalization works and the value of x over a mini-batch is B = {x_1,……,m}, and the parameter to be learned is γ, β and the out-put will be $\{y_i = BN_{\gamma,\beta}(x_i)\}$

$$\mu_\beta \leftarrow \tfrac{1}{m}\textstyle\sum_{i=1}^{m} x_i \qquad mini - batch\ mean \tag{1}$$

$$\sigma_\beta^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_\beta)^2 \quad mini - batch\ variance \tag{2}$$

$$\widehat{x_\iota} \leftarrow \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \in}} \qquad normalize \tag{3}$$

$$y_i \leftarrow \gamma\hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i) \quad Scale\ and\ shift \tag{4}$$

We have used one passive layer with a Relu function and one end layer with a linear function. The End layer linear activation hypothesis is used because of the regression problem, and the linear activation gives a better result for the model. Then we create a wrapper for the neural network, which helps to create a bridge between Keras and scikit-learn. By using some hyperparameter tuning, we have used the Keras regressor for the regression problem with the epochs 30, batch size 400, and we set the verbosity to 0 because we do not need to see how far the network has been trained. For fitting the model, we have used the Randomized-Search-CV from sci-kit-learn, with the help of the Randomized-Search-CV, we can get better results of the model by using different hyperparameters. We have tried by putting different hyperparameters, which will work best. So, we have used 80 neurons or 90 neurons and a dropout probability of 0.1 or 0.3. This gives a parameter with a total of 4 combinations. Then we will create the instance of Randomized-Search-CV with our model, the parameter with the four combinations, a scoring function we have used MSE (Mean Squared Error), and a verbose is set to 3. In figure 4, we can see the model in which we have used the stock market prediction.

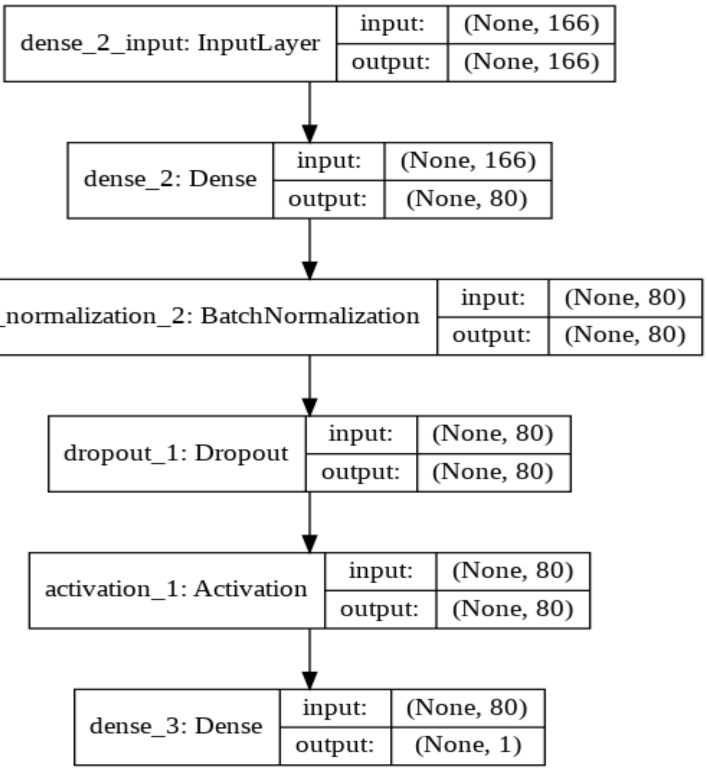


Figure 4. Proposed Neural Network Model

D. Performance Indicators

We start by some underlying arrangement of the model and predict the output dependent on some info. The predicted value contrasted, and the target label and the proportion of our model execution taken. At that point, the different parameters of the model balanced iteratively to arrive at the optimal estimation of the performance metric. In most of the things, the outputs assessed from two types: the primary is RMSE or RMSRE between real value and predicted value, the next is Mean Directional Accuracy, which suggests the proportion of correct analysis of price flow direction, as up and down movements that can immensely matter for taking any decisions. The little enhancements in prediction performance are often beneficial. For more understanding of the mathematical form, refer to table 2 in which we have explained what parameters used to find error and score of the model for regression problems.

Table 2. Performance indicators of Regression

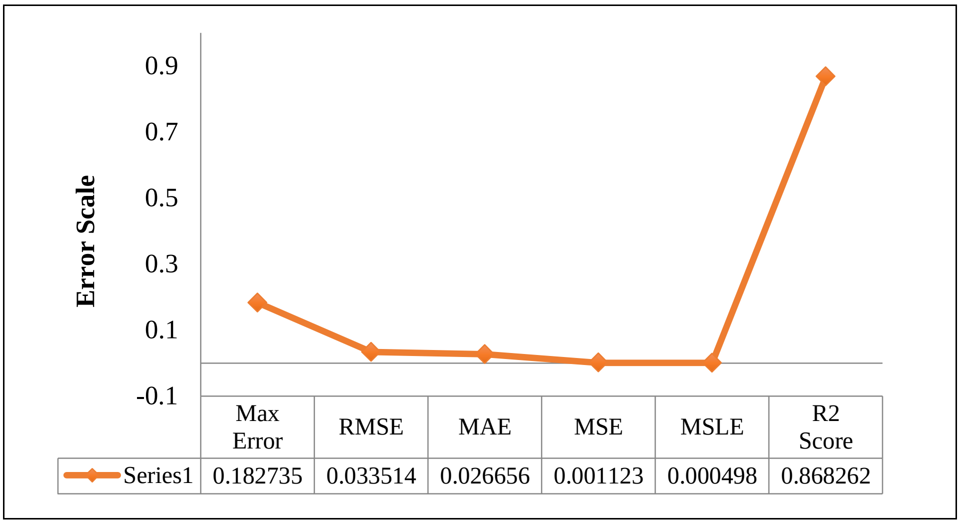
Hyperparameter	Explanation
R^2 score	It calculates the determination and it is for regression score function. $R^2(y,y') = 1 - \frac{\sum_{i=n}^{n}\left(y_i - y'_i\right)^2}{\sum_{i=n}^{n}(y_i - y')^2}$
MAE	It finds out MAE, a parameter corresponding to the absolute error loss or l1-norm loss. $MAE(y,y') = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples-1}} \lvert y_i - y'_i \rvert$
MSLE	The parameter is identified for the squared logarithmic error or loss function. $MSLE(y,y') = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples-1}} \left(log_e(1+y_i) - log_e\left(1+y'_i\right)\right)^2$
MSE	It computes parameter corresponding to the value of the squared error or loss. $MSE(y,y') = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples-1}} \left(y_i - y'_i\right)^2$
RMSE	It measures the standard deviation of the mistakes which occurs when a prediction is made on a dataset. $RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Predicted_i - Actual_i)^2}{N}}$
Max Error	It identifies the between forecasted and original value error. $Max\ Error(y,y') = \max\left(\lvert y_i - y'_i \rvert\right)$

Experimental Result and Discussion

In the Numerai competition, we can get the results by uploading our prediction on the Numerai tournament. Numerai measures Performance-based on the correlation of rank and predictions and actual targets. By correlation matrix, we can show the heatmap for the features, as shown in figure 1. By this, we can quickly check the data and some of the features which are related to others.

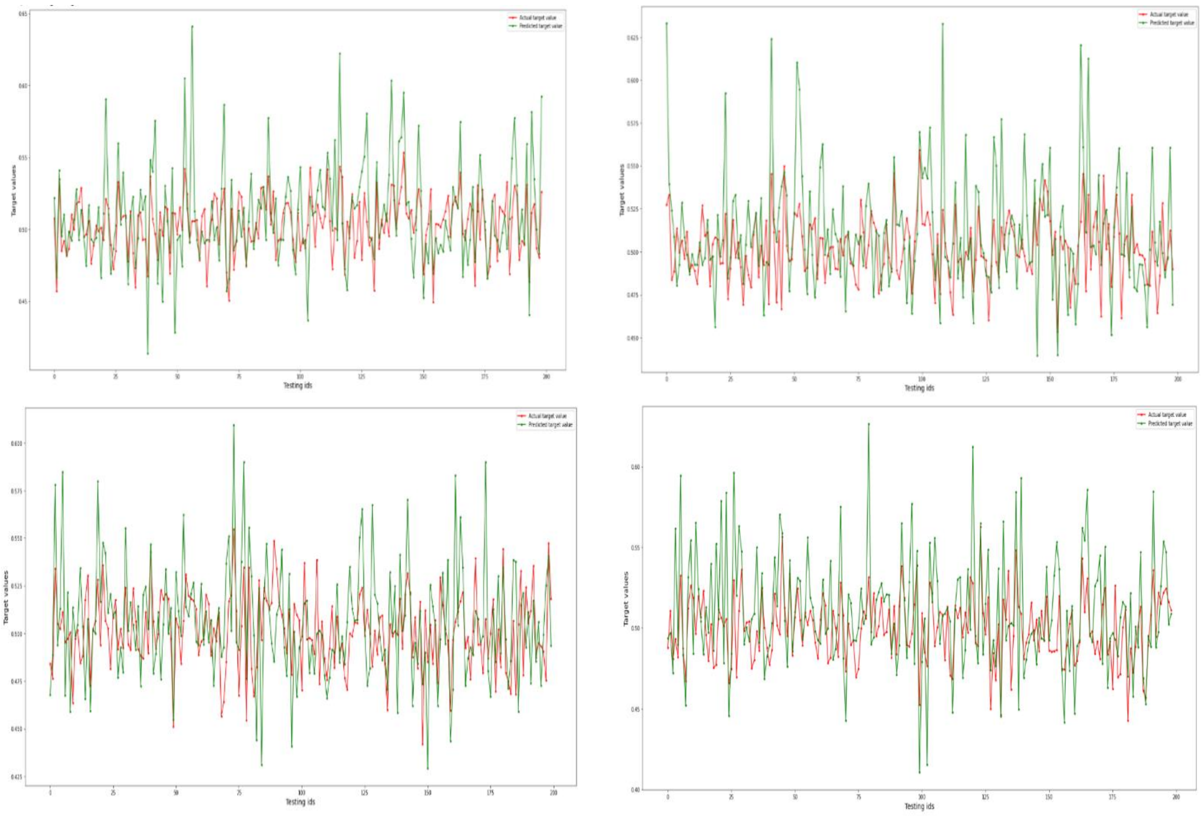
A. Hyperparameter Analysis

In figure 5 graph, we show the values of error and the score for the model. In the regression problem, we use the MSE, MAE, Max Error, RMSE, and MSLE for checking the error we get for the predictions. We got the R2 score as 0.8683. With the R2 score, we can check the model is appropriately fit. If the value is less than 0.5, we can assume that the model is poorly fit. So, as the R2 score is close to 1, the model is appropriately fitted. Next, we compared the prediction with the actual prediction to check that they are near to each other. The prediction depends upon the R2 score; as the R2 score is near to 1, the values will help us right. So, with the help of figure 6, which has five graphs, the five graphs tell the actual prediction vs. predicted prediction comparison. The predictions are so much which we cannot visualize accurately. So, we have used the values in the range of 1 to 200, 200 to 400, 400 to 600, 600 to 800, and the last graph are with the values 1 to 5000, we can visualize predicted values against actual values.



	Max Error	RMSE	MAE	MSE	MSLE	R2 Score
Series1	0.182735	0.033514	0.026656	0.001123	0.000498	0.868262

Figure 5. Hyperparameter Analysis of Proposed Model



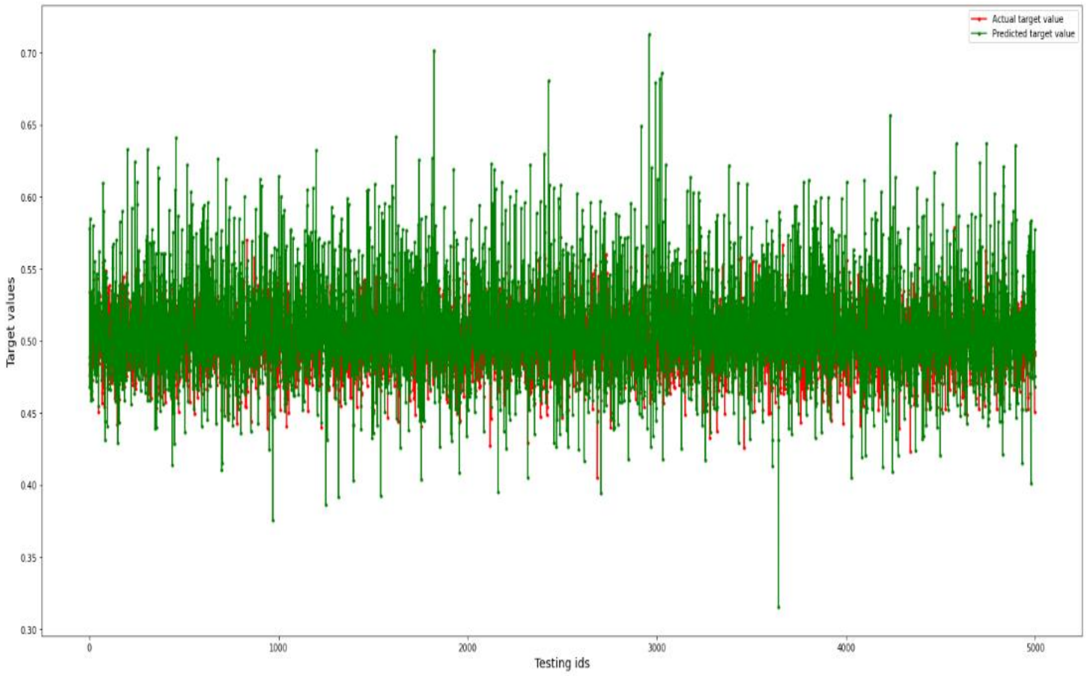


Figure 6. All 5 graph shows the comparision of actual values and predicted values at the differnet count of ids

B. Comparative Analysis

In this section, we compare our model with the existing techniques like Two-stream Gated Recurrent Unit Network (TGRU), Hybrid time series predictive neural network (HTPNN), Long short-term memory (LSTM), Multilayer Perceptron (MLP) and Decision Tree (DT). We have generated the comparation graph in figure 7 which shows the accuracy of the performance of our model. The comparison graph clearly shows our proposed method accuracy is high compared to all the previous works.

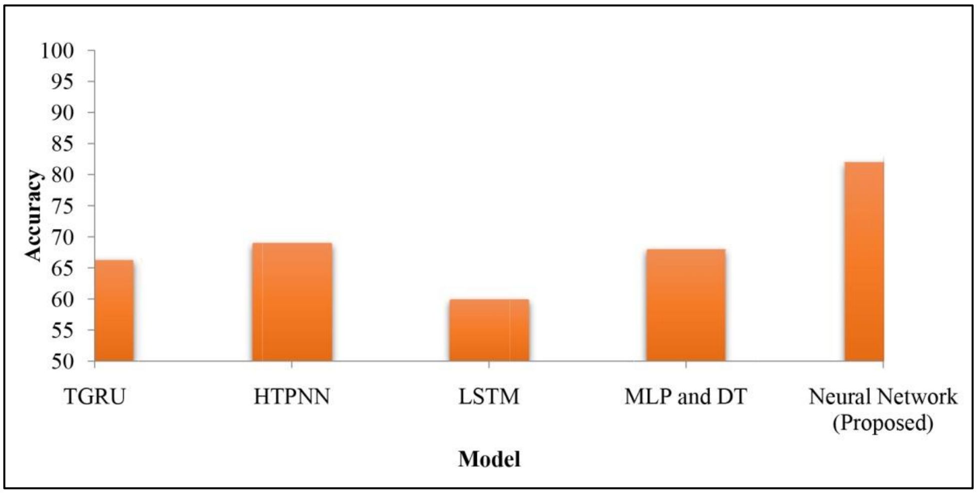


Figure 7. Accuracy performance comparison between other models and proposed model

Conclusions

In our proposed work, we used a customized Neural Network model for the prediction of the live stock market. The performance of the experimental method of the proposed neural network gave excellent predictions for the targets, and the consistency of the model is also right. The adaptive nature of the Neural Network accelerates the connection between input and output values in a way that the obtained network can predict the stock market for the future. Hence, we conclude that NN is an efficient model for the prediction of financial data, and it might be done on real-time values. The proposed neural network model experimental result shows the high stock market prediction accuracy that is 86% accuracy on the training set as well as the 14% loss on the testing set compared with all the existing models. Our future work is to implement an efficient neural network model to improve the performance of the conjecture of financial exchange.

Acknowledgment

I would like to express my special thanks to the Numerai for providing the live dataset for the stock market prediction.

Compliance with Ethical Standards

Conflict of interest on behalf of all authors, the corresponding author states that there is no conflict of interest.

References

Al-Hmouz, R., Pedrycz, W., & Balamash, A. (2015). Description and prediction of time series: A general framework of Granular Computing. Expert Systems with Applications, 42(10), 4830-4839. doi:10.1016/j.eswa.2015.01.060

Babu, M., N.Geethanjali, & B.Satyanarayana, P. (2012, January 02). Clustering Approach to Stock Market Prediction. Retrieved from http://paper.researchbib.com/view/paper/59204

Bagheri, A., Peyhani, H. M., & Akbari, M. (2014). Financial forecasting using ANFIS networks with Quantum-behaved Particle Swarm Optimization. Expert Systems with Applications, 41(14), 6235-6250. doi:10.1016/j.eswa.2014.04.003

Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. Plos One, 12(7). doi:10.1371/journal.pone.0180944

Bengio, Y. (2009). Learning Deep Architectures for AI. Foundations and Trends® in Machine Learning, 2(1), 1-127. doi:10.1561/2200000006

Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. Expert Systems with Applications, 83, 187-205. doi:10.1016/j.eswa.2017.04.030

Ghosh, P., Neufeld, A., & Sahoo, J. K. (2020, April 21). Forecasting directional movements of stock prices for intraday trading using LSTM and random forests. Retrieved from https://arxiv.org/abs/2004.10178

Gunduz, H., Cataltepe, Z., & Yaslan, Y. (2017). Stock market direction prediction using deep neural networks. 2017 25th Signal Processing and Communications Applications Conference (SIU). doi:10.1109/siu.2017.7960512

Hiransha M, Dr. E. A. Gopalakrishnan, Vijay Krishna Menon, Dr. Soman K. P, (2018). NSE Stock Market Prediction Using Deep-Learning Models. Procedia Computer Science, 132, 1351-1362. doi:10.1016/j.procs.2018.05.050

Idrees, S. M., Alam, M. A., & Agarwal, P. (2019). A Prediction Approach for Stock Market Volatility Based on Time Series Data. IEEE Access, 7, 17287-17298. doi:10.1109/access.2019.2895252

Khan, W., Ghazanfar, M. A., Azam, M. A., Karami, A., Alyoubi, K. H., & Alfakeeh, A. S. (2020). Stock market prediction using machine learning classifiers and social media, news. Journal of Ambient Intelligence and Humanized Computing. doi:10.1007/s12652-020-01839-w

Khan, W., Malik, U., Ghazanfar, M. A., Azam, M. A., Alyoubi, K. H., & Alfakeeh, A. S. (2019). Predicting stock market trends using machine learning algorithms via public sentiment and political situation analysis. Soft Computing, 24(15), 11019-11043. doi:10.1007/s00500-019-04347-y

Kimoto, T., Asakawa, K., Yoda, M., & Takeoka, M. (1990). Stock market prediction system with modular neural networks. 1990 IJCNN International Joint Conference on Neural Networks. doi:10.1109/ijcnn.1990.137535

Kusuma, R. M., Ho, T., Kao, W., Ou, Y., & Hua, K. (2019, February 26). Using Deep Learning Neural Networks and Candlestick Chart Representation to Predict Stock Market. Retrieved from https://arxiv.org/abs/1903.12258

Lathuiliere, S., Mesejo, P., Alameda-Pineda, X., & Horaud, R. (2020). A Comprehensive Analysis of Deep Regression. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1-1. doi:10.1109/tpami.2019.2910523

Lee, J., Kim, R., Koh, Y., & Kang, J. (2019). Global Stock Market Prediction Based on Stock Chart Images Using Deep Q-Network. IEEE Access, 7, 167260-167277. doi:10.1109/access.2019.2953542

Liu, G., & Wang, X. (2019). A Numerical-Based Attention Method for Stock Market Prediction With Dual Information. IEEE Access, 7, 7357-7367. doi:10.1109/access.2018.2886367

Minh, D. L., Sadeghi-Niaraki, A., Huy, H. D., Min, K., & Moon, H. (2018). Deep Learning Approach for Short-Term Stock Trends Prediction Based on Two-Stream Gated Recurrent Unit Network. IEEE Access, 6, 55392-55404. doi:10.1109/access.2018.2868970

Nguyen, T., & Yoon, S. (2019). A Novel Approach to Short-Term Stock Price Movement Prediction using Transfer Learning. Applied Sciences, 9(22), 4745. doi:10.3390/app9224745

Parmar, I., Agarwal, N., Saxena, S., Arora, R., Gupta, S., Dhiman, H., & Chouhan, L. (2018). Stock Market Prediction Using Machine Learning. 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC). doi:10.1109/icsccc.2018.8703332

Parmar, I., Agarwal, N., Saxena, S., Arora, R., Gupta, S., Dhiman, H., & Chouhan, L. (2018). Stock Market Prediction Using Machine Learning. 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC). doi:10.1109/icsccc.2018.8703332

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques. Expert Systems with Applications, 42(1), 259-268. doi:10.1016/j.eswa.2014.07.040

Qiu, J., Wang, B., & Zhou, C. (2020, January 03). Forecasting stock prices with long-short term memory neural network based on attention mechanism. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/31899770

Ren, R., Wu, D. D., & Liu, T. (2019). Forecasting Stock Market Movement Direction Using Sentiment Analysis and Support Vector Machine. IEEE Systems Journal, 13(1), 760-770. doi:10.1109/jsyst.2018.2794462

S Abdulsalam Sulaiman Olaniyi, Adewole, Kayode S. Jimoh, R. G. Stock Trend Prediction Using Regression Analysis – A Data Mining Approach, ARPN Journal of Systems and Software, Volume 1 No. 4, JULY 2011

Selvamuthu, D., Kumar, V., & Mishra, A. (2019). Indian stock market prediction using artificial neural networks on tick data. Financial Innovation, 5(1). doi:10.1186/s40854-019-0131-7

Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017). Stock price prediction using LSTM, RNN and CNN-sliding window model. 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI). doi:10.1109/icacci.2017.8126078

Son, Y., Noh, D., & Lee, J. (2012). Forecasting trends of high-frequency KOSPI200 index data using learning classifiers. Expert Systems with Applications, 39(14), 11607-11615. doi:10.1016/j.eswa.2012.04.015

Stoean, C., Paja, W., Stoean, R., & Sandita, A. (2019). Deep architectures for long-term stock price prediction with a heuristic-based strategy for trading simulations. Plos One, 14(10). doi:10.1371/journal.pone.0223593

Wang, Y., Liu, H., Guo, Q., Xie, S., & Zhang, X. (1970). Stock Volatility Prediction by Hybrid Neural Network: Semantic Scholar. Retrieved from https://www.semanticscholar.org/paper/Stock-Volatility-Prediction-by-Hybrid-Neural-Wang-Liu/310b54f1913ac93cb2817e810c62e92e6a65e326

Yuan, X., Yuan, J., Jiang, T., & Ain, Q. U. (2020). Integrated Long-Term Stock Selection Models Based on Feature Selection and Machine Learning Algorithms for China Stock Market. IEEE Access, 8, 22672-22685. doi:10.1109/access.2020.2969293

Bibliographic information of this paper for citing:

Singh, Karanveer; Tiwari, Rahul; Johri, Prashant & Elngar, Ahmed A. (2020). Feature Selection and Hyper-parameter Tuning Technique using Neural Network for Stock Market Prediction. *Journal of Information Technology Management*, Special Issue, 89-108.