IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Deep Learning System For Short-Term Stock Market Price Trend Prediction

[1]Darak Ankita Balaprasad, [2] Prof. Sushil Venkatesh Kulkarni

[1]Darak Ankita Balaprasad, [2] Prof. Sushil Venkatesh Kulkarni

[1]Student, [2]Project Guide
[1]Computer Science & Engineering Technology,
[1]M.B.E. Society's College of Engineering, Ambajogai, India

Abstract:

Deep learning for predicting stock market prices and trends has grown much more popular than before in the age of big data. In order to predict the price trend of stock markets, we collected two years' worth of data from the Chinese stock market and provideda thorough customisation of feature engineering and deep learning-based model. The pre-processing of the stock market dataset, the use of various feature engineering approaches, and a customized deep learning based system for stock market price trend prediction make up the suggested solution, which is complete. We thoroughly assessedseveral popular machine learning models and came to the conclusion that our proposed solution outperformed them thank s to the thorough feature engineering we developed. The method predicts stock market trends with good overall accuracy. This work adds to the stock analysis research community in both the financial and technical areas through the thorough design and evaluation of prediction term lengths, feature engineering, and data pre-processing techniques.

Index Terms – LSTM, SVR, MSE, MAE, RMSE.

I. INTRODUCTION

STOCK market research is a popular topic for academics in both the financial and technical sectors because it is one of the primary industries to which investors devote their attention. In this study, we aim to develop a cutting-edge price trend prediction model that focuses on short-term price trend prediction. We employed the support vector machine (SVM) model to short-term forecasting of stock prices. Our primary contribution is a comparison of multi-layer perceptron's (MLP) and support vector machines (SVM), which revealed that SVM outperformed MLP in most circumstances but the outcome was also influenced by various trading methods. Researchers in the financial fields were studying stock market data at the same time using traditional statistical approaches and signal processing techniques. Principal component analysis (PCA), one of the optimization techniques, was also used to predict short-term stock prices. Over the years, researchers have tried to study stock market transactions such volume burst hazards in addition to stock price-related analysis, which has broadened the research arena for stock market analysis and shown that it still has a lot of promise. Many proposed solutions attempted to merge machine learning and deep learning techniques based on prior approaches as artificial intelligence techniques advanced in recent years, and then offered new metrics that serve as training features. To examine various quantitative methods in stock markets, we suggested a convolutional neural network (CNN) and a long short-term memory (LSTM) neural network-based model. As an open-sourced data API, we created the dataset on our own using the data source. Our approach is unusual because we suggested feature engineering combined with a fine-tuned system rather than just an LSTM model.

Before training the prediction model, we reviewed the prior research, identified the gaps, and presented a solution architecture with a thorough feature engineering process. Other machine learning algorithms can now benefit from the success of the feature extension method working in tandem with recursive feature reduction methods to attain high accuracy scores for short-term price trend prediction. By introducing our unique LSTM model, we were able to further raise the prediction scores across all evaluation metrics. In similar earlier efforts, the suggested method performed better than deep learning-based models.

II. SURVEY OF RELATED WORKS

In this section, we discuss related works. We reviewed the related work in two different domains: technical and financial, respectively

A. The dataset:

This section describes the data that was taken from the open data sources and the generated final dataset. Since there are many different types of stock market-related data, we first compared the related works from the survey of financial research works in order to define the paths for data collecting. We established the dataset's data structure after data collection. Below, we provide a detailed description of the dataset, along with information about the data structure and tables for each type of data that include segment definitions.

The details of our dataset:

We will go into great detail about the dataset in this part. 3558 stocks from the Chinese stock market are included in the dataset. We gathered daily fundamental data, daily price data, and the history of suspending and resuming trading, the top 10 shareholders, and other information for each stock ID. We cite two justifications for our decision to choose 2 years as the time period for this dataset: first, most investors conduct stock market price trend analyses using data from the most recent 2 years; second, utilizing more recent data would improve the analysis's findings. We used a web-scraping technique to get information from Sina Finance web pages and the SWS Research website in addition to the open-sourced API.

Methods:

In this section, we present the proposed methods and the design of the proposed solution. Moreover, we also introduce the architecture design as well as algorithmic and implementation details.

Problem statement:

We looked at feature engineering, financial domain expertise, and prediction algorithms to determine the optimal strategy for forecasting short-term price movements. The following three research topics were then addressed, one for each aspect, in turn: How does feature engineering improve the accuracy of model prediction? How does the construction of prediction models benefit from research in the financial domain? And which algorithm is most effective at identifying short-term price trends?

The first inquiry relates to feature engineering. We are interested in learning how the feature selection approach enhances the effectiveness of prediction models. We may infer from the quantity of earlier studies that stock price data is highly noisy and that there are correlations across features, which makes price prediction notoriously challenging.

The effectiveness of the information we gathered from the financial domain is being assessed as the second study question. In contrast to earlier research, our evaluation will focus on the efficacy of newly additional characteristics that we collected from the financial domain, in addition to the standard evaluation of data models such as the training costs and scores. We present a few elements from the financial industry. Even if we were only able to extract a few particular findings from earlier studies, the associated raw data still needs to be transformed into useful features. We integrate the characteristics with other popular technical indices after extracting related features from the financial domain in order to vote out the aspects with the greatest impact. We would be unable to discuss all of the aspects of the financial world because there are so many.

Which algorithms will we use to model our data? is the third research query. Researchers have worked hard to estimate prices accurately based on their prior study. We break the issue down into two parts: predicting the trend and then the precise figure. This essay concentrates on the initial action. Therefore, the goal has been changed to solve a binary classification problem while also figuring out how to effectively reduce the negative effects of the high amount of noise. Our strategy is to break down the

complex problem into smaller, less interdependent problems, solve each one individually, and then combine the solutions into an ensemble model to serve as a reference for investing behavior.

In this work, we will compare our approach with the outperformed machine learning models in the evaluation part and find the solution for this research question.

Proposed solution:

Our suggested solution's high-level design might be divided into three sections. To ensure that the features chosen are very effective, the feature selection process comes first. The data are examined and dimensionality reduction is done next. The creation of a prediction model for the target stocks is the final step, which is the primary contribution of our work. Different categories of stocks can be categorized in a variety of ways. Long-term investments are preferred by some investors, whereas short-term investments are more appealing to others. One of the phenomena that show the stock price prediction has no set criteria and that effective features on data are required can be seen frequently in stock-related reports that perform on average while the stock price is rising sharply.

We concentrate on predicting short-term price trends in this study. Right now, all we have is the unlabeled raw data. So labeling the data is the first step. Since our research is focused on the short term, we use the comparison of the closing price of the current trading day with the closing price of n trading days ago to identify price trends. We designate it as 1 if the price trend is upward or 0 if it is downward. To be more precise, we forecast the price trend of the nth day using the indices from the indices of the n1st day.

To help with the design of our feature expansion, we used their works as references and their guidelines as inspiration. In contrast to the financial domain, where researchers prefer to analyze a specific investment scenario, we discovered that the majority of prior works in the technical domain analyzed all stocks. To bridge this gap between the two domains, we decided to apply a feature extension based on the information we gathered from the financial domain before we started the RFE procedure.
The more features there are, the more difficult the training process will be because we intend to turn the data into time series. Therefore, we will start our proposed solution architecture by employing randomized PCA to take advantage of the dimensionality reduction.

Detailed technical design elaboration:

This section elaborates on the technological architecture in depth as a thorough solution built on the use of numerous existing data preparation, feature engineering, and deep learning approaches. We divided the content into primary procedures, and each procedure has steps that follow an algorithm. The following section goes into greater detail regarding the algorithms. The purpose of the material in this part is to provide examples of the data workflow.

The most popular technical indices are chosen based on the literature study, and these are then fed into the feature extension technique to produce an extended feature set. From the extended feature set, we will pick the features that are the most useful. The PCA technique will then be used to divide the dimension into j features by feeding the data along with I chosen features into it.

Our suggested technique is innovative in that we will implement feature extensions that are popular with stock market participants in addition to applying the technical method to the raw data. The next subsection provides more information on feature extension. While designing and customising feature engineering and deep learning solutions in this study, lessons learned from implementing and optimising deep learning-based solutions were taken into consideration.

Applying feature extension:

The feature extension is the first main procedure. The most popular technical indices determined from similar research serve as the input data for this block. The three feature extension techniques are maximum-minimum scaling, polarisation, and fluctuation percentage calculation. The three feature extension methods do not all apply to all technical indices; instead, this procedure exclusively applies the useful extension methods to technical indices. We look at the indices calculation and select

meaningful extension methods. The extended features will be merged with the most popular technical indices following the feature extension technique, i.e., input data and output data, and fed into the RFE block as input data in the following phase.

Applying recursive feature elimination:

Following the aforementioned feature extension, we investigate the top I features using the Recursive Feature Elimination (RFE) algorithm. We calculate the coefficient and feature importance to estimate all the features. Additionally, we keep all pertinent features while limiting the number of characteristics we remove from the pool to one every stage. Then, the PCA-related next step's input will be the output of the RFE block.

Applying principal component analysis (PCA):

Prior to using PCA, feature pre-processing comes first. Because the output from RFE is in multiple units and some of the features following RFE are percentage data and others are very huge values. The outcome of the major component extraction will be impacted. Consequently, a feature pre-processing is required prior to feeding the data into the PCA method. In the "Results" section, we also compare methods and their efficacy.

Following feature pre-processing, the processed data with chosen I features are sent into the PCA algorithm, which reduces the feature matrix scale into j features. The goal of this stage is to keep as many useful features as feasible while reducing the computational burden of training the model. This study assesses the optimal i and j pairing, which reduces computing consumption while having comparatively superior prediction accuracy. The outcome is also available in the "Results" section. The system will receive a reshaped matrix with j columns after the PCA step.

Long short-term memory (LSTM) model:

While data pre-processing is necessary before supplying the data to the LSTM layer, PCA lowered the dimensionality of the input data. The principle component input matrix lacks time steps, which is why the data preprocessing step was added before the LSTM model. The quantity of time steps is one of the crucial factors in LSTM training. As a result, for both the training and testing datasets, we must model the matrix into matching time steps.

The final step after completing the data pre-processing is to feed training data into the LSTM and assess performance using testing data. Even with just one LSTM layer, the NN structure, as an RNN variation, is still a deep neural network because it can process sequential data and remembers its hidden states across time. One or more LSTM units make up an LSTM layer, and each LSTM unit performs classification and prediction based on time series data using its cells and gates. There are two layers that make up the LSTM structure. After the PCA process, j determines the input dimension. The input LSTM layer is the first layer, while the output LSTM layer is the second layer.

Algorithm elaboration:

This section contains in-depth information on the algorithms we created by combining and modifying various current methodologies. Information on the terms, parameters, and optimizers. The algorithm stages are shown as octagons in this section titled "Algorithm elaboration" by the legend on the right. Here is a quick introduction to data pretreatment before delving into the algorithm phases in further detail: We must programme the ground truth because we will use supervised learning methods. The closing price of the current trading date is compared to the closing price of the prior trading date that the users want to compare with to determine the research's ground truth. Put a 1 next to the price increase, otherwise, 0 will be assigned as the ground truth. The ground truth processing is done in accordance with a variety of trading days because this research effort is not only concentrated on predicting the price trend of a certain time period but also short-term generally. We can think of the prediction term length as a parameter even if the algorithms won't vary with it.

Specifically, the first method is the hybrid feature engineering component for creating high-quality training and testing data. It correlates to the RFE, PCA, and Feature extension blocks. The LSTM method block, which includes time-series data preprocessing, NN construction, training, and testing, is the second algorithm.

III. RESULTS:

Data selection:

The data selection is the process of selecting the data for predicting the stock.

- The dataset was collected from dataset repository like UCI.
- The dataset is in the format like '.csv'.
- In this system, the time series dataset is used for predicting the stock.
- The dataset which contains the information about the high, low, open and close price.
- With the help of panda's package, we can read or load our input dataset.

```
----------------------------------------
============== Input data ============
----------------------------------------
   Symbol Series        Date  ...  Low Price  Last Price  Close Price
0    SBIN      EQ  02-Jan-17  ...     242.60      243.55       243.60
1    SBIN      EQ  03-Jan-17  ...     241.10      244.90       244.90
2    SBIN      EQ  04-Jan-17  ...     242.20      243.20       242.90
3    SBIN      EQ  05-Jan-17  ...     243.70      245.50       245.35
4    SBIN      EQ  06-Jan-17  ...     245.50      246.05       245.90
5    SBIN      EQ  09-Jan-17  ...     246.00      246.70       247.05
6    SBIN      EQ  10-Jan-17  ...     246.40      248.90       248.30
7    SBIN      EQ  11-Jan-17  ...     249.00      251.70       252.15
8    SBIN      EQ  12-Jan-17  ...     250.55      251.15       251.25
9    SBIN      EQ  13-Jan-17  ...     249.10      251.00       250.90
10   SBIN      EQ  16-Jan-17  ...     250.70      256.40       255.75
11   SBIN      EQ  17-Jan-17  ...     254.40      255.90       256.00
12   SBIN      EQ  18-Jan-17  ...     256.60      258.30       258.35
13   SBIN      EQ  19-Jan-17  ...     256.50      258.05       258.40
14   SBIN      EQ  20-Jan-17  ...     250.30      251.20       251.05
15   SBIN      EQ  23-Jan-17  ...     250.10      254.80       254.15
16   SBIN      EQ  24-Jan-17  ...     254.00      254.60       254.90
17   SBIN      EQ  25-Jan-17  ...     254.20      259.85       259.20
18   SBIN      EQ  27-Jan-17  ...     259.50      266.50       266.45
19   SBIN      EQ  30-Jan-17  ...     263.40      263.85       263.95

[20 rows x 9 columns]
```

Preprocessing:

- Data pre-processing is the process of removing the unwanted data from the dataset.
- Data pre-processing allows for the removal of unwanted data with the use of data cleansing, this allows the user to have a dataset to contain more valuable information after the pre-processing stage for data manipulation later in the data mining process.
- Missing data removal: In this process, the null values such as missing values and Nan values are replaced by 0.
- Encoding Categorical data: That categorical data is defined as variables with a finite set of label values.

```
------------------------------------------------
============== Data Preprocessing ============
------------------------------------------------
Symbol          0
Series          0
Date            0
Prev Close      0
Open Price      0
High Price      0
Low Price       0
Last Price      0
Close Price     0
dtype: int64
```

Data splitting:

- Data splitting is the act of partitioning available data into two portions, usually for cross-validator purposes.
- One Portion of the data is used to develop a predictive model and the other to evaluate the model's performance.
- Separating data into training and testing sets is an important part of evaluating data mining models.
- Typically, when you separate a data set into a training set and testing set, most of the data is used for training, and a smaller portion of the data is used for testing.

```
======================================================
------------------ Data Splitting ------------------
======================================================

Total number of data's in input         : 248

Total number of data's in training part : 173

Total number of data's in testing part : 75
```

Feature Extraction:

```
======================================================
-------- Principle component Analysis --------------
======================================================

 The original features is : 4

 The reduced feature is  : 3
```

Performance analysis:

- MSE: The mean squared error (MSE) is a common way to measure the prediction accuracy of a model. It is calculated as:

$$MSE = (1/n) * \Sigma (actual - prediction)^2$$

- Where:
- Σ – a fancy symbol that means “sum”
- n – sample size
- actual – the actual data value
- forecast – the predicted data value

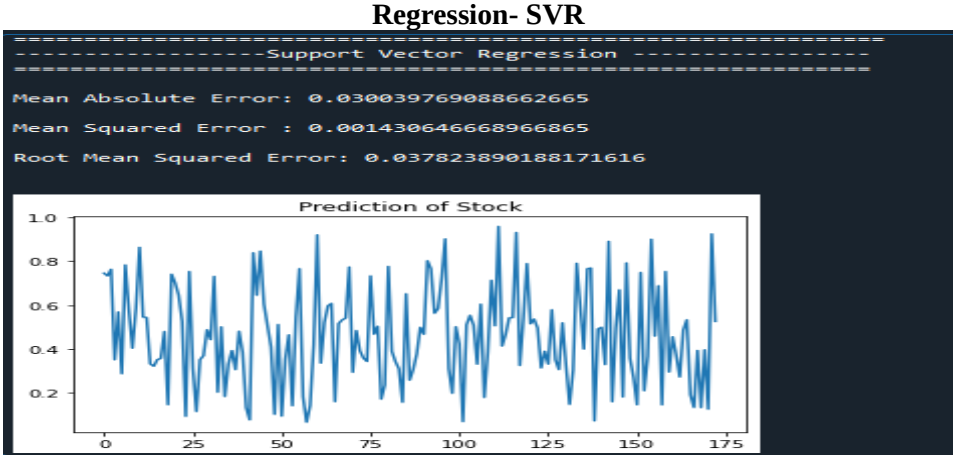
```
============================================================
---------------- Prediction (Stock Price ) -----------------
============================================================

============================
[0] The stock price = 0.7458777851409175
============================
[1] The stock price = 0.7330977094903861
============================
[2] The stock price = 0.763509943823862
============================
[3] The stock price = 0.35045483843753783
============================
[4] The stock price = 0.5686894763814878
============================
[5] The stock price = 0.28658908848099013
============================
[6] The stock price = 0.78339551023489
============================
[7] The stock price = 0.5764799094462199
```

Regression:

- In our process, we have to implement the machine and deep learning algorithm such as Support vector regression and LSTM.
- Support Vector Regression is a supervised learning algorithm that is used to predict discrete values.
- Support Vector Regression uses the same principle as the SVMs. The basic idea behind SVR is to find the best fit line. In SVR, the best fit line is the hyper plane that has the maximum number of points.
- Long Short-Term Memory (LSTM) networks are a type of recurrent neural network capable of learning order dependence in sequence prediction problems.

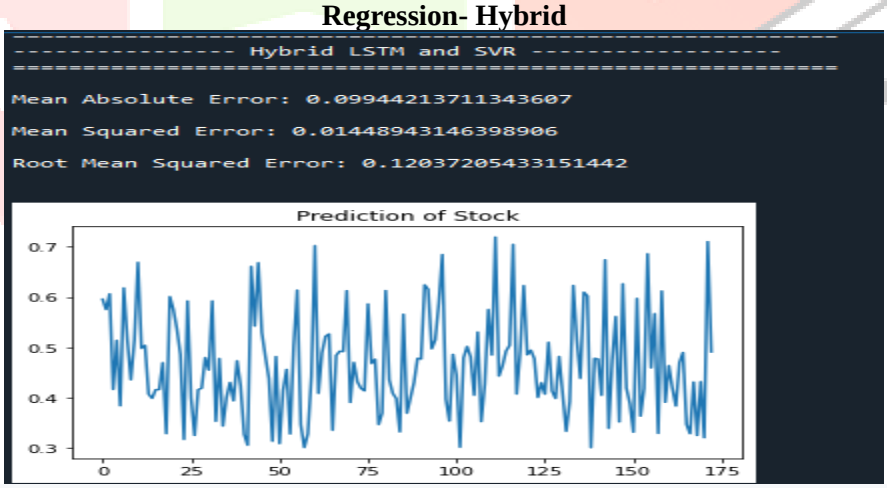
- This is a behavior required in complex problem domains like machine translation, speech recognition, and more. LSTMs are a complex area of deep learning.

Regression- SVR



Regression- LSTM

Regression- Hybrid



IV. CONCLUSION

This project consists of three parts: feature engineering, stock price trend prediction model based on long short-term memory, and data extraction and pre-processing of the Chinese stock market dataset (LSTM). We gathered, purified, and organised data from the Chinese stock market for two years. We looked at many strategies frequently employed by real-world investors, created a fresh algorithm component we called a feature extension, and it worked well. To create a feature engineering technique that is both successful and efficient, we used the feature expansion (FE) approaches with recursive feature elimination (RFE), followed by principal component analysis (PCA). The system is made to order by combining the feature engineering process with an LSTM prediction model that outperforms the industry standard models.

We also conducted a thorough examination of this work. We draw several heuristic conclusions that could be future research problems in both the technical and financial search domains by contrasting the most widely used machine learning models with our suggested LSTM model under the feature engineering portion of our proposed system.

In contrast to earlier works, our proposed solution is distinctively customised because, rather than simply recommending yet another cutting-edge LSTM model, we recommended a fine-tuned and personalised deep learning prediction system along with the use of extensive feature engineering and combined it with LSTM to perform prediction. We bridge the gap between investors and researchers by analysing the findings from earlier studies and suggesting a feature extension algorithm before recursive feature elimination.

ACKNOWLEDGMENT

I pay thanks to our project guide Prof. Sushil Venkatesh Kulkarni for assistance and guidance especially related to technicalities and also who encouraged and motivated us.

REFERENCES

[1] Short-term stock market price trend prediction using a comprehensive deep learning system, Shen and Shafq J Big Data (2020) 7:66 https://doi.org/10.1186/s40537-020-00333-6**.**

[2] Ayo CK. Stock price prediction using the ARIMA model. In: 2014 UKSim-AMSS 16th international conference on computer modelling and simulation. 2014. https://doi.org/10.1109/UKSim.2014.67.

[3] Atsalakis GS, Valavanis KP. Forecasting stock market short-term trends using a neuro-fuzzy based methodology. Expert Syst Appl. 2009;36(7):10696–707.

[4] Brownlee J. Deep learning for time series forecasting: predict the future with MLPs, CNNs and LSTMs in Python. Machine Learning Mastery. 2018. https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networkspython-keras/

[5] Eapen J, Bein D, Verma A. Novel deep learning model with CNN and bi-directional LSTM for improved stock market index prediction. In: 2019 IEEE 9th annual computing and communication workshop and conference (CCWC). 2019. pp. 264–70. https://doi.org/10.1109/CCWC.2019.8666592.

[6] Fischer T, Krauss C. Deep learning with long short-term memory networks for financial market predictions. Eur J Oper Res. 2018;270(2):654–69. https://doi.org/10.1016/j.ejor.2017.11.054.

[7] Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classifcation using support vector machines. Mach Learn 2002;46:389–422.

[8] Hafezi R, Shahrabi J, Hadavandi E. A bat-neural network multi-agent system (BNNMAS) for stock price prediction: case study of DAX stock price. Appl Soft Comput J. 2015;29:196–210. https://doi.org/10.1016/j.asoc.2014.12.028.

[9] Halko N, Martinsson PG, Tropp JA. Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. SIAM Rev. 2001;53(2):217–88.

[10] Hochreiter S, Schmidhuber J. Long short-term memory. J Neural Comput. 1997;9(8):1735–80. https://doi.