JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

THE POSITIVE TRAITS RELATED TO STOCK TRADES USING MACHINE LEARNING APPROACH

[1]Nagsen Bansod, [2]Pratibha Sharma, [3]Siddharth Dabhade
[123]Assistant Professors,
[1]Dr. G Y Pathrikar College of Computer Science and IT, MGM University Aurangabad, India
[2]J. D. Birla Institute, Kolkata, India.
[3]School of Management Studies, National Forensic Sciences University, Sector 9, Gandhinagar-382007, Gujarat - India

Abstract : This study has been undertaken to investigate the determinants of stock returns in Indian Stock Exchange (ISE) using machine learning algorithms and IoT, for finding the positive trades in stock. Experimentation performed with the dataset of Kaggle, Indian and American Stock exchange. In this article we used google colab and github data repositories for better interface with API key. prediction of positive trades is possible using machine learning technology; it is shown in the figure and table of result.

***IndexTerms* - Machine learning, API, Stock, Trade, Colab.**

I. INTRODUCTION

Stock trading refers to the buying and selling of shares in a particular company. if a person owns a share or stock, he/she owns a piece of the company. The goal of stock traders is to capitalize on short-terms market events to sell stocks for a profit or buy stock at a low.When machine learning is used, a quantitative evaluation based on a large volume of data is possible, and evaluation can be conducted quickly and inexpensively, contributing to the activation of patent transactions. However, due to patent characteristics, securing the necessary training data is challenging because most patents are traded privately to prevent technical information leaks. In this study, the derived marketable value of a patent through event study is used for patent value evaluation, matching it with the semantic information from the patent calculated using latent Dirichlet allocation (LDA)-based topic modeling. In addition, an ensemble learning methodology that combines the predicted values of multiple predictive models was used to determine the prediction stability. Base learners with high predictive power for each fold were different, but the ensemble model that was trained on the base learners' predicted values exceeded the predictive power of the individual models. The Wilcoxon rank-sum test indicated that the superiority of the accuracy of the ensemble model was statistically significant at the 95% significance level(Gali Nageshwarrao et al 2022). Investing is long-term and involves lesser risk, while trading is short-term and involves high-risk. Both earn profits, but traders frequently earn more profit compared to investors when they make right decisions, and the market is performing accordingly. Thus, there is a need for a system that can analyze the positive aspects of stock trading.

II. LITERATURE SURVEY

The evolution of the Internet of Things (IoT) has promoted the prevalence of the financial industry as a variety of stock prediction models have been able to accurately predict various IoT-based financial services.(Lili Chen et al 2021) In practice, it is crucial to obtain relatively accurate stock trading signals. Considering various factors, finding profitable stock trading signals is very attractive to investors, but it is also not easy. In the past, researchers have been devoted to the study of trading signals. A genetic algorithm (GA) is often used to find the optimal solution. In this study, a long short-term (LSTM) memory neural network is used to study stock price fluctuations, and then, genetic algorithms are used to obtain appropriate trading signals. A genetic algorithm is a search algorithm that solves optimization. In this paper, the optimal threshold is found to determine the trading signal. In addition to trading signals, a suitable trading strategy is also crucial. In addition, this research uses the Kelly criterion for fund management; that is, the Kelly criterion is used to calculate the optimal investment score. Effective capital management can not only help investors increase their returns but also help investors reduce their losses.(Thiago Christiano Silva et al 2019) model investor behavior by training machine learning techniques with financial data comprising more than 13,000 investors of a large bank in Brazil over 2016 to 2018. We take high-frequency data on every sell or buy operation of these investors on a daily basis, allowing us to fully track these investment decisions over time. then analyze whether these investment changes correlate with the IBOVESPA index. We find that investors decide their investment strategies using recent past price changes. There is some degree of heterogeneity in investment decisions. Overall, we find evidence of mean-reverting investment strategies. We also find evidence that female investors with higher academic degrees have a less pronounced mean-reverting strategy behavior compared to male investors and those with lower academic degrees. Finally, this paper provides a general methodological approach to mitigate potential biases arising from *ad-hoc* design decisions of discarding or introducing variables in empirical econometrics. For that, we use feature selection techniques from machine learning to identify relevant variables in an objective and concise way. (Lin Y et al 2021) PRML, a novel candlestick pattern recognition model using machine learning methods, is proposed to improve stock trading decisions. Four popular machine learning methods and 11 different features types are applied to all possible combinations of daily patterns to start the pattern recognition schedule. Different time windows from one to ten days are used to detect the prediction effect at different periods. An investment strategy is constructed according to the identified candlestick patterns and suitable time window. We deploy PRML for the forecast of all Chinese market stocks from Jan 1, 2000 until Oct 30, 2020. Among them, the data from Jan 1, 2000 to Dec 31, 2014 is used as the training data set, and the data set from Jan 1, 2015 to Oct 30, 2020 is used to verify the forecasting effect. Empirical results show that the two-day candlestick patterns after filtering have the best prediction effect when forecasting one day ahead; these patterns obtain an average annual return, an annual Sharpe ratio, and an information ratio as high as 36.73%, 0.81, and 2.37, respectively. After screening, three-day candlestick patterns also present a beneficial effect when forecasting one day ahead in that these patterns show stable characteristics. Two other popular machine learning methods, multilayer perceptron network and long short-term memory neural networks, are applied to the pattern recognition framework to evaluate the dependency of the prediction model. A transaction cost of 0.2% is considered on the two-day patterns predicting one day ahead, thus confirming the profitability. Empirical results show that applying different machine learning methods to two-day and three-day patterns for one-day-ahead forecasts can be profitable.(Prasad A et al 2021) Machine learning algorithms with varied levels of input variables and found that though the performance of models measured by root-mean-square error (RMSE) for regression and accuracy score for classification models varied greatly, long short-term memory (LSTM) model displayed higher accuracy amongst the machine and deep learning models reviewed. However, reinforcement learning algorithm performance measured by profitability and Sharpe ratio outperformed all. In general, traders can maximize their profits by using machine learning instead of using technical analysis. Technical analysis is very easy to implement, but the profit based on it can vanish too soon or making a profit using technical analysis is almost difficult because of its simplicity. Hence, studying machine, deep and reinforcement learning algorithms is vital for traders and investors. These findings were based on the literature review consolidated in the result section. Nowadays, the most significant challenge in the stock market is to predict the stock prices. The stock price data represents a financial time series data which becomes more difficult to predict due to its characteristics and dynamic nature

III. METHODOLOGY

Based on other relevant research mentioned earlier, I would expect a machine learning model to be able to extract a more meaningful signal from stock-specific social media analysis, small-cap and microcap stocks, and improve the overall accuracy with a deep neural network. data from the stock market, particularly some technology stocks. We will learn how to use pandas to get stock information, visualize different aspects of it, and finally we will look at a few ways of analyzing the risk of a stock, based on its previous performance history. We will also be predicting future stock prices through a Long Short Term Memory (LSTM) method.

We'll be answering the following questions along the way:

1.) What was the change in price of the stock over time?
2.) What was the daily return of the stock on average?
3.) What was the moving average of the various stocks?
4.) What was the correlation between different stocks'?
5.) How much value do we put at risk by investing in a particular stock?
6.) How can we attempt to predict future stock behavior? (Predicting the closing price stock price of APPLE inc using LSTM)

IV. EXPERIMENTAL SETUP AND RESULTS

Here for experimental setup we have used data sources data_source='Yahoo'',start='2012-01-01',end='2019-12-17') and he end-to-end data science lifecycle of developing a predictive model for stock price movements with Alpha Vantage APIs and a powerful machine learning algorithm called Long Short-Term Memory (LSTM). By completing this project, you will learn the key concepts of machine learning / deep learning and build a fully functional predictive model for the stock market, all in a single Python file.

```
import math
import pandas_datareader as web
import numpy as np
import pandas as pd
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
```

```
from keras.layers import Dense,LSTM
import matplotlib.pyplot as plt
plt.style.use('fivethirtyeight')
```

Table 4.1 Trade Open-close and Adjacent Close

	High	Low	Open	Close	Volume	Adj Close
Date						
2012-01-03	58.92857	58.42857	58.485714	58.747143	75555200	51.115936
2012-01-04	59.240002	58.468571	58.57143	59.062859	65005500	51.390648
2012-01-05	59.792858	58.952858	59.278572	59.718571	67817400	51.961189
2012-01-06	60.392857	59.888573	59.967144	60.342857	79573200	52.504375
2012-01-09	61.107143	60.192856	60.785713	60.247143	98506100	52.421093
...	...	...	...	...	...	...
2019-12-11	271.100006	268.5	268.809998	270.769989	19689200	270.769989
2019-12-12	272.559998	267.320007	267.779999	271.459991	34327600	271.459991
2019-12-13	275.299988	270.929993	271.459991	275.149994	33396900	275.149994
2019-12-16	280.790009	276.980011	277	279.859985	32046500	279.859985
2019-12-17	281.769989	278.799988	279.570007	280.410004	28539600	280.410004

```
plt.figure(figsize=(16,8))
plt.title('Close Price History')
plt.plot(df['Close'])
plt.xlabel('Date',fontsize=18)
plt.ylabel('Close Price USD($)',fontsize=18)
plt.show()
```

Figure 4.1 Trade close trade price History

```
# build LSTM model
model=Sequential()
model.add(LSTM(50,return_sequences=True,input_shape=(x_train.shape[1],1
  )))
model.add(LSTM(50,return_sequences=False))
model.add(Dense(25))
model.add(Dense(1))
train=data[:training_data_len]
valid=data[training_data_len:]
valid['Predictions']=predictions

#visualize the data
plt.figure(figsize=(16,8))
plt.title('Model')
plt.xlabel('Date',fontsize=18)
plt.ylabel('Close Price USD($)',fontsize=18)
plt.plot(train['Close'])
plt.plot(valid[['Close','Predictions']])
plt.legend(['Train','Val','Predictions'],loc='lower right')
plt.show()
```

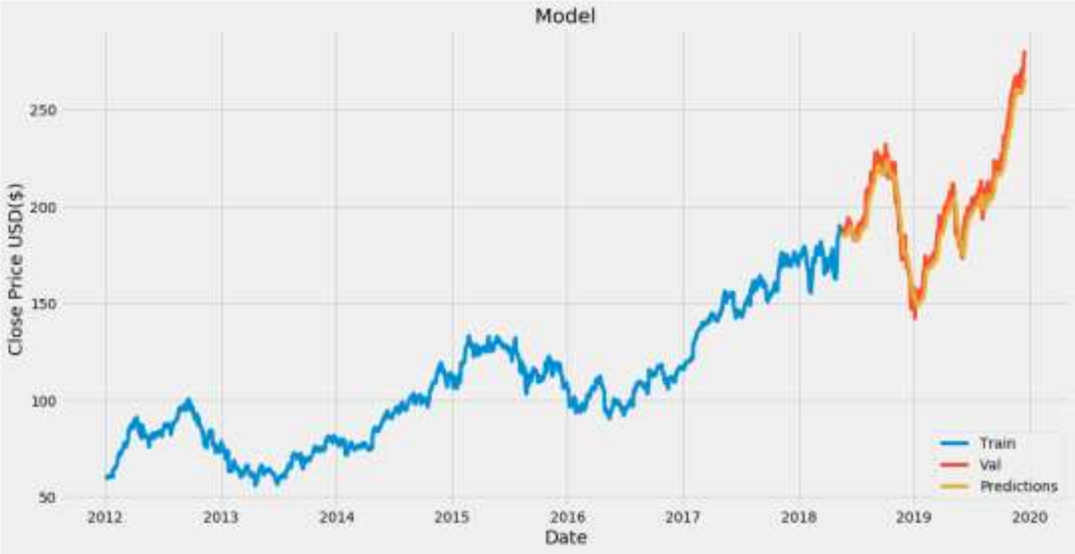

Figure 4.2 shows shows the valid and predicted prices

	Close	Predictions
Date		
2018-05-17	186.990005	184.572235
2018-05-18	186.309998	184.71109
2018-05-21	187.630005	184.603256
2018-05-22	187.160004	184.561096
2018-05-23	188.360001	184.495834
2019-12-11	270.769989	260.616028
2019-12-12	271.459991	261.560303
2019-12-13	275.149994	262.555115
2019-12-16	279.859985	263.92334
2019-12-17	280.410004	265.879852

Table 4.2 shows shows the valid and predicted prices

4.1 Data preparation: acquiring financial market data from Alpha Vantage

4.1.1 Alpha VantageStock API

Extracting stock data manually from websites is a tedious task to perform. It is even more difficult to find reliable data. This situation is a hectic one to deal with for beginners. The solution to this problem is to automate the process of extracting the data. How can we do this? The answer is simple, with the help of Stock API.

4.1.2 Need of Stock.

A Stock API is a database hosted in a cloud that offers real-time stock updates, intraday data, historical data, and much more. In recent days, almost every financial institution is using stock APIs for trading and research purposes as it helps in cutting the expense of buying stock data directly from the exchanges (which costs a hefty amount of money). Secondly, with the help of programming, it is easy to interact with Stock APIs to obtain the desired information. Finally, we have access to varied types of data with highly customizable features. Alpha Vantage provides free stock APIs through which users can access a wide range of data like real-time updates, and historical data on equities, currencies, and cryptocurrencies.

In this article, we are going to interact with the stock API provided by Alpha Vantage with python to extract three types of equity data: intraday data, historical data, and the latest updates or information of stocks. Before moving on to the coding part, the user must create a developer account on Alpha Vantage (https://www.alphavantage.co/support/#api-key), only then, the API key (a vital part of an API) can be accessed to pull data.

4.2 Importing Packages

We are going to use only two packages in this article which are Pandas, and Requests. The Pandas package is used to carry out an extensive amount of data manipulations and processing and the Requests package provides functions to pull data from an API.

4.2.1 Table Intraday data of TSLA

	open	high	low	close	volume
2021-04-20 16:59:00	714.82	714.82	714.82	714.82	959.0
2021-04-20 17:01:00	714.56	714.56	714.56	714.56	426.0
2021-04-20 17:02:00	715.00	715.00	715.00	715.00	1337.0
2021-04-20 17:03:00	714.75	714.75	714.00	714.00	2523.0
2021-04-20 17:04:00	714.20	714.20	714.20	714.20	304.0
...	...	...	...	...	...
2021-04-20 19:56:00	713.21	713.21	713.10	713.10	454.0
2021-04-20 19:57:00	713.11	713.11	713.05	713.10	1088.0
2021-04-20 19:58:00	712.77	712.77	712.62	712.62	1044.0
2021-04-20 19:59:00	712.98	712.98	712.17	712.17	2676.0
2021-04-20 20:00:00	712.11	712.39	712.00	712.39	6826.0

4.3 Extracting Historical Data

(Nikhil Aditya et al 2022) This part is to specifically extract the historical data of the given stock using the stock API provided by Alpha Vantage

Figure 4.2 Historical Data of MSFT

	open	high	low	close	adj close	volume
2020-01-02	158.780	160.73	158.3300	160.62	158.572113	22634546.0
2020-01-03	158.320	159.95	158.0600	158.62	156.597613	21121681.0
2020-01-06	157.080	159.10	156.5100	159.03	157.002385	20826702.0
2020-01-07	159.320	159.67	157.3200	157.58	155.570873	21881740.0
2020-01-08	158.930	160.80	157.9500	160.09	158.048871	27762026.0
...	...	...	...	...	...	...
2021-04-14	257.475	258.83	255.1600	255.59	255.590000	23070938.0
2021-04-15	257.931	259.93	257.7300	259.50	259.500000	25627481.0
2021-04-16	259.470	261.00	257.6014	260.74	260.740000	24878582.0
2021-04-19	260.190	261.48	257.8210	258.74	258.740000	23209260.0
2021-04-20	257.820	260.20	256.8400	258.26	258.260000	19722875.0

Table 4.3.1 Historical Data of MSFT

The first thing we did is to define a function named 'get_historic_data' that takes the stock's symbol ('symbol') as a required parameter and the starting date of the historical data ('start_date') as an optional parameter. Like we did in the previous function, we are defining the API key and the URL and stored them into their respective variable. Next, we are extracting the historical data in JSON format using the 'get' function and stored it into the 'raw_df' variable. After doing some processes to clean and format the raw JSON data, we are returning it in the form of a clean Pandas dataframe. Finally, we are calling the created function to pull the historic data of Microsoft from the starting of 2020 and stored it into the ‘msft_hist’ variable

```python
def get_live_updates(symbol):
    api_key = open(r'api_key.txt')
    api_url = f'https://www.alphavantage.co/query?function=GLOBAL_QUOTE&symbol={symbol}&apikey={api_key}'
    raw_df = requests.get(api_url).json()
    attributes = {'attributes':['symbol', 'open', 'high', 'low', 'price', 'volume', 'latest trading day', 'previous close', 'change', 'change percent']}
    attributes_df = pd.DataFrame(attributes)
    values = []
    for i in list(raw_df['Global Quote']):
        values.append(raw_df['Global Quote'][i])
    values_dict = {'values':values}
    values_df = pd.DataFrame(values).rename(columns = {0:'values'})
    frames = [attributes_df, values_df]
    df = pd.concat(frames, axis = 1, join = 'inner').set_index('attributes')
    return df

ibm_updates = get_live_updates('IBM')
ibm_updates
```

Figure 4.3.2 IBM Updates

attributes	values
symbol	IBM
open	137.0700
high	139.7700
low	136.7000
price	138.1600
volume	15480579
latest trading day	2021-04-20
previous close	133.1200
change	5.0400
change percent	3.7861%

Table 4.3.2 IBM Update

The structure or format of the code for this function is almost similar to the previous functions we created earlier but, the URL of the API changes. The steps involving data processing are slightly extensive in this function as the raw JSON data which is being pulled is comparatively messy. From the output dataframe being represented, we can observe that almost every fundamental information of a stock is revealed.Here we learned to pull intraday data, historical data, and the latest information of a stock using the stock APIs provided by Alpha Vantage. We just explored a pinch of Alpha Vantage’s huge collection of stock APIs to carry out a varied amount of tasks. Also, we retained the default API parameters while defining the URL of the API but, there are a lot of flexible and customizable parameters that come along with the API. So, it is highly recommended to experiment with the function we created with different API parameters. That’s it! We finally automated one of the stressful tasks to perform in finance using free stock APIs.

V. Conclusion

We have found various machine learning algorithms which are already implemented in this research article using python, googlecom laboratories. we used a dataset using machine learning we can predict stock and found 95 percent accuracy. The Machine learning approach to analyze the positive traits related to stock trading on studying the positive aspects of stock trading since they have many negative attributes as well. The trains a machine learning model and implements using algorithms of classification and prediction.

VI. Acknowledgment

We would like to thank the author for providing the library data dataset, their research works also cited by us. thank to google colab for providing us with experimental support.

References

[1] Lili Chen, Lingyun Sun, Chien-Ming Chen, Mu-En Wu, Jimmy Ming-Tai Wu, "Stock Trading System Based on Machine Learning and Kelly Criterion in Internet of Things", *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 7632052, 9 pages, 2021.

[2] Thiago Christiano Silva, Benjamin Miranda Tabak, Idamar Magalhães Ferreira, "Modeling Investor Behavior Using Machine Learning: Mean-Reversion and Momentum Trading Strategies", *Complexity*, vol. 2019, Article ID 4325125, 14 pages, 2019.

[3] Lin Y, Liu S, Yang H, Wu H, Jiang B (2021) Improving stock trading decisions based on pattern recognition using machine learning technology. PLoS ONE 16(8): e0255558.

[4] Prasad A, Seetharaman A. Importance of Machine Learning in Making Investment Decisions in the Stock Market. *Vikalpa*. 2021;46(4):209-222.

[5] Selvamuthu, D., Kumar, V. & Mishra, A. Indian stock market prediction using artificial neural networks on tick data. *Financ Innov* 5, 16 (2019).