Artificial Stock Analyst: Using Machine Learning to Create a Holistic Approach to Stock Analysis

Capstone Report, 2023

Nathaniel Gleberman
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
njg7hy@virginia.edu

ABSTRACT

The stock market is a critical element in the pursuit of financial freedom, yet many Americans are discouraged from investing by a lack of expertise. To improve inclusivity, I propose combining traditional stock analysis algorithms with machine learning to better predict stock equity fluctuation, specifically in the S&P 500. The TensorFlow machine learning package would be combined with other algorithms built in Python to systematically rank the impact and prediction of different variables related to an individual company. Free historical data on the S&P 500 from Kaggle combined with a free API call from Polygon would create the dataset backing controlled by the Pandas library. A dashboard of the results would be visualized through a containerized Flask web application hosted for free on Google Firebase. The expected outcome would be a free and helpful tool that could supplement short-term stock selection to create more consistent returns and encourage investment. After building the application, results should be measured and tracked to test the longevity of the models.

1. INTRODUCTION

From February to April 2020, the stock market plummeted with the largest daily drop in the global markets reaching nearly 13%. In mere weeks, billions of dollars in savings and income seemingly disappeared as markets reached new bottoms. Events like pandemics that turn the market upside down are referred to as “Black Swan” events because they are difficult to predict. The prospect of savings rapidly disappearing is scary and might discourage investing in stock markets. Yet, by 16 August 2021, the S&P 500 had doubled from that pandemic bottom, an achievement that typically takes nearly one thousand days. Buying at the bottom could have netted investors years’ worth of income (Li & Rattner, 2021).

Understanding when to buy and sell stock is a learnable and important skill that can help people retire faster. The market is complex as stock price movement can rarely be attributed to one factor (Harper, 2022). This complication cannot be mistaken for pure chance. Algorithms can penetrate this complication and create educated estimates for price movement, and although perfect accuracy is never guaranteed, trading with an algorithm can help increase the probability of positive returns.

2. RELATED WORKS

According to Devadoss and Ligori (2013) Artificial Neural Networks are widely regarded as one of the most effective machine learning algorithms for stock prediction. The network consists of an input layer, one hidden layer, and the output layer. Inputs include price, high, low, closing price, and volume. ANNs are modeled after the cerebral cortex of the brain and are capable of effectively forecasting with data as complicated as that around a stock price. The researchers found that increasing input data allowed for better results.

My project would build upon the accomplishments of the applications of ANNs by combining them into a more holistic approach to prediction. Instead of stopping with ANNs, the networks would make up a part of the larger picture. This way, in conditions where the ANN was less effective at prediction, other strategies can be employed to account for its weaknesses.

Researchers at Siksha 'O' Anusandhan University (2016) take this strategy further. By applying a computational efficient link artificial neural network (CEFLANN), they communicate predictions as classifications using buy, hold, or sell signals. This modification to the ANN model used above turns the complex calculations derived from technical indicators into a classification.

I propose avoiding classification and opting for percent weekly change to convey a quantitative prediction regarding the price. This will better display the severity of predictions as well as the possibility of more opportune moments to buy or sell because a stock moving up 1% and a stock moving up 10% would both reflect "buy" signals, but the 10% mover would net more gain. Further, I would still utilize machine learning to create a quantitative prediction to include inside the overarching machine learning model rather than opting for pure technical indicators as inputs.

3. PROPOSAL DESIGN

The artificial stock analyst is a dynamic webpage that grants access to weekly stock movement predictions derived from a Python backend.

3.1 Review of System Architecture

The system is a publicly accessible web application available through Google Firebase, a method of web publishing with servers and a free tier. As such, the application features a front end consisting of HTML and CSS generated through Flask, a microweb framework written in Python. The backend features machine learning models from TensorFlow, a machine learning library developed at Google. In addition, technical indicators will be coded in Python to easily interact with TensorFlow and Flask. A database is stored in the backend containing publicly accessible stock data. This data is accessed, added to, and manipulated using the Pandas package. The front end will allow users to select a stock from the S&P 500 and will connect with the backend to then display predicted weekly change for that stock.

3.2 Design Specifications

The front end features a simple user interface that grants access to the predictions for each model. The back end is where the bulk of the work is done. Inside the backend are two major components: database engine and the stock predictor engine. The stock predictor engine consists of technical indicators, stock prediction machine learning model, and the overarching machine learning model.

3.2.1 User Interface

The user interface will model a simple search page such as Google with a few extra additions. The search bar will consist of a drop-down menu where users can select or search for a company in the S&P 500 by company name or ticker. The submission button will respond to user's hover and upon submission, the button will trigger the actions necessary to display the results for the selected company. Lastly, a section below the search will display top predicted winners and top predicted losers for the week in the form of company ticker connected with the estimated percent change and last week's change. Negative changes will be colored red. Positive changes will be colored green. The results will be displayed below the search and above the weekly estimates in a similar manner to the cells inside the weekly estimates table. Overall, the user interface will be simplistic, allowing for easy use and fast development while hiding the complexities behind the outputted numbers.

3.2.2 Database Creation and Interaction

Clean, accurate, and consistent data is a necessity. To accomplish that goal, data can be freely scraped from Nasdaq.com with daily stock information consisting of data, close/last, volume, open, high, and low. The data at the time of creation will be scraped going back five years. From there, it will be continuously updated on a weekly basis.

The sample URL follows the path market-activity/stocks/ticker/historical where ticker can be changed for each company in the S&P to automate and streamline the scraping process. Duplicate entries will not be allowed. This table will be referred to as *technical_indicators.db*. A script can be programmed to scan the past ten days and append to the database while removing duplicates to help automate updating the database.

The next database needed is for the overarching machine learning model. This database will consist of the outcomes of each technical indicator for each week along with economic data. The economic data will be derived from Federal Reserve Economic Data, read in as a CSV, and added to the database as an ETL pipeline (Lewis, Mertens, & Stock, 2008). The data grabbed is the start of week data and the weekly economic index which is "an index of ten daily and weekly indicators of real economic activity, scaled to align with the four-quarter GDP growth rate" (*Weekly Economic Index*, n.d.). This data will be merged with the technical indicators for the week on date.

3.2.3 Technical Indicators

The technical indicators will consist of the following tools: artificial neural network (ANN) stock predictor, Bollinger Bands, Moving Average Convergence/Divergence (MACD), Relative Strength Index (RSI), Exponential Moving Average (EMA), and Simple Moving Average (SMA). All technical indicators will access the *technical_indicators* database. Programmed in Python, the technical indicators will follow their mathematical formulas to output quantifiable results that can be stored and run through the overarching machine learning model.

3.2.4 Overarching Machine Learning Model

The overarching machine learning algorithm needs to solve a regression problem as a supervised learner by outputting a percent weekly change created from all the technical indicators charted against certain economic variables and the date. To accomplish this goal, a Random Forest model will be used as an ensemble learning method that constructs multiple decision trees. The Random Forest will allow for an easy understanding of the most effective variables with a provided visual in the form of a decision tree while serving as a robust model to properly determine the necessary action. Moreover, if the model is overanalyzing, Random Forest can allow for easy pivoting to a Decision Tree model to accomplish the same outcome in a less robust manner that could be less prone to over analysis.

3.3 Challenges

The key challenges with this project stems from keeping the tools free. Firebase allows a free method of hosting a website; however, this tier limits daily users and server power. With no backend server running consistently, calling APIs, updating the database, and continuously training the model are difficult. In addition, outside dependencies such as data scraping lead to vulnerabilities if one of the scraped websites were to change one day. Another key challenge is adjusting the technical indicators to signify a weekly percent change. Though quantifiable, this output will take some adjusting and learning to properly equate the technical indicator outcomes to weekly values. Additional machine learning could be used to adjust rates which could further strain the free servers.

4. ANTICIPATED RESULTS

Anticipated results would be general stock prediction accuracy. General is applied here because perfect accuracy cannot be guaranteed. By limiting the artificial analyst to the S&P 500, the assistant would look to, at the very least, replicate market return. The primary concern with the results lies in over engineering. By using many technical indicators, the intelligent predictor risks "analysis paralysis," a term used to describe over analyzing a stock price (Quantified Strategies, 2023).

5. CONCLUSION

Learning effective trading can be a time-consuming and arduous challenge for many people. According to a 2019 study, 52.6% of families in the United States directly own stock, yet most wealthier Americans tend to directly own stock (USA Facts, 2021). This statistic serves as a strong indicator that to reach the wealthier quartiles, owning stock is a necessity especially while balancing family life. By putting free and powerful tools into the hands of people looking to diversify their savings, trading can become a manageable and encouraging pursuit for anyone, regardless of wealth level.

6. FUTURE WORK

This application can be expanded through optional donations, a small paywall or advertisements that could generate more consistent funding. Financial backing for the artificial analyst would allow access to better economic data hidden behind paywalls in addition to a stronger server backing that could continuously update and manage more users. The server would also be capable of a more robust self-analysis better allowing the model to track its own history, records, and accuracy to improve performance. As the model is improved and data can be grabbed real time, different time frames of investment could be predicted to enhance personalization of the platform for users.

REFERENCES

Dash, R., & Dash, P. K. (2016). A hybrid stock trading framework integrating technical analysis with machine learning techniques. *The Journal of Finance and Data Science*, *2*(1), 42–57. https://doi.org/10.1016/j.jfds.2016.03.002

Devadoss, A., Antony, T., & Ligori, A. (2013). Stock Prediction Using Artificial Neural Networks. *International Journal of Data Mining Techniques and Applications*, *02*, 283–291. https://doi.org/10.1016/j.eswa.2021.115537

Harper, D. R. (2022, July 22). *What Drives the Stock Market?* Investopedia. https://www.investopedia.com/articles/basics/04/100804.asp

Lewis, Daniel J., Mertens, Karel, & Stock, James H. (2008, January 5). *Weekly Economic Index (Lewis-Mertens-Stock)*. FRED, Federal Reserve Bank of St. Louis. https://fred.stlouisfed.org/series/WEI

Li, Y., & Rattner, N. (2021, August 16). *S&P 500 doubles from its pandemic bottom, marking the fastest bull market rally since WWII*. CNBC. https://www.cnbc.com/2021/08/16/sp-500-doubles-from-its-pandemic-bottom-marking-the-fastest-bull-market-rally-since-wwii.html

New York Fed. *Weekly Economic Index (WEI)*. (n.d.). www.newyorkfed.org. https://www.newyorkfed.org/research/policy/weekly-economic-index#/interactive

Quantified Strategies. (2023, October 2). *Quantitative Indicators (Trading) | Quantified Strategies.* https://www.quantifiedstrategies.com/quantitative-indicators/

USA Facts. *What percentage of Americans own stock?* (2021, March 2). https://usafacts.org/articles/what-percentage-of-americans-own-stock/