Machine Learning Prediction of S&P 500 Price Movement

Clara Beuoy[1], Isaac Teal[1], and Tamiru Workneh[1]

[1]Arizona State University, Tempe, AZ 85281, USA

May 4, 2025

1 Abstract/Executive Summary

With the rise of Data Science, particularly with Machine Learning becoming an essential tool in this field, institutional investors, economists, and researchers have attempted to leverage the power of data to predict stock price movements. The goal is to find profitable strategies for making market trades. This has been a core area of research since the 1950s [27, 37, 46]. In fact, using Google Scholar to search for "Stock Price Prediction Machine Learning", we found more than 24 thousand articles written on the topic, several of which deal directly with S&P 500 predictions [4, 11, 25, 41]. Predicting market movements is clearly a challenging and valuable application of Data Science.

The S&P 500 [23] is an index of the top 500 companies in the U.S. stock market, and serves as a benchmark that investors can use to evaluate the general health of the stock market and the economy as a whole. Although you cannot invest directly in the S&P 500, there are options available for Americans looking to invest in their retirement. Index funds that track the S&P 500 are accessible, and investors can also consider the SPY stock ticker as a way to invest in line with the S&P 500.

Swing trading is a practice seen in the financial markets wherein investors buy stock when they expect the stock to increase in value in the short term and sell stock when they expect the stock to drop in value in the short term. By making iterative trade actions, they attempt to ride the crests up to market highs and then sell and wait for the market to hit a low before reentry [5, 8]. In doing so, they profit from the volatility within the securities [18].

Our research will attempt to find exploitable data within the historical records of the S&P 500, which investors and those saving for retirement can leverage in order to swing-trade their investment or retirement portfolios at an advanced rate of growth over traditional buy-and-hold strategies.

2 Introduction & Background

2.1 Introduction

Our research plan centered on finding amplifying information both within and outside of the S&P 500 index history to feed into machine learning methods to predict price swings that we could generate a swing trading strategy to take advantage of. Our initial plan was to look at a variety of macroeconomic variables [1, 26, 34, 35] such as the price of eggs, oil [24, 42], gas, etc., to explore media sentiment analysis [13, 29, 33], and to extract useful technical analysis indicators from the index movements.

With a plan established, we examined the work completed in these areas to gather additional insights into what held predictive power and what did not.

2.2 Literature Review

In 1986, Chen et al. [10] examined the effects of Inflation, Treasury Bill Rates, Long-term Government Bond returns, Industrial production, Low-grade bonds, Equally weighted equities, Value-weighted equities, Consumption, and Oil prices on the stock market as a whole. Of note, they found that the oil price coefficient they regressed did not provide a meaningful impact on the accuracy of their predictions, but they noted that the risk associated was not priced in well during the period of their study. Considering the significant changes to the geopolitical landscape and the effects of events such as the Russian Oil Boycott after the invasion of Ukraine, this metric is worth revisiting.

Sakaria (2024) highlights the potential of artificial neural networks and ensemble learning techniques and considers a series of AI and ML models in S&P 500 movement forecasting [40]. Ahmed (2024) illustrates the efficiency of deep learning models compared to traditional statistical techniques by contrasting a series of machine learning models with a perspective to determine their performance in short-term forecasting [2]

Their article shows the strengths and limitations of various machine learning models, some of which excel at recognizing patterns, but others of which are particularly good at making sense of macroeconomic data. Whether future price action can be predicted from historical S&P 500 prices is one of the simplest problems in predictive finance. Zhong (2021) predicts price movements in the S&P 500 based on technical, fundamental, and text data. Accuracy of prediction assignment, their result indicates, is greatly improved by the inclusion of financial indicators, movement averages, and macroeconomic factors [48]. This establishes that historical data continues to be a solid component of financial modeling.

In addition to this, the study demonstrates that precision in the model is differentially impacted by various sources of data. Technical variables like average movements and momentum oscillators are more suited to short-term forecasting, whereas fundamental variables like earnings reports have long-term forecasting ability. Technical Indicators across momentum, trend, volatility, and volume categories significantly enhanced the accuracy of daily S&P 500 prediction [31]. Economic Indicators and Sectoral Behavior continues to be a significant component of stock market research, especially during financial crises. integrating macroeconomic indicators—such as oil prices, economic policy uncertainty, and geopolitical risk—improves the performance of forecasting models over extended periods [26].

The response of the S&P 500's various economic sectors to crises is addressed by Sudah (2024), who also addresses sectoral vulnerability and resilience to macroeconomic shocks. By recognizing the sectors that dominate or lag when considering the general market performance, sector-specific patterns can be incorporated in machine learning models to improve their accuracy. When economic recessions, for example, are encountered, stocks in the technology and healthcare industries will have dissimilar properties from finance and industry stocks. [43]

Much effort has been spent over the past few years on using sentiment research to aid in predicting the stock market. Yu (2024) discusses sentiment analysis methods used in financial news outlets and social media to improve S&P 500 trend forecasting. Sentiment analysis and other conventional financial factors might improve model stability, especially during times of market instability, based on their research [47]. According to this, Zhong (2021) emphasizes the significance of text data by showing how investor sentiment in the form of earnings announcements and news stories can be used to be a good indicator of the direction of the market [48].

2.3 Unresolved Questions

Even though ML models turned out to be highly predictive, there remains a set of issues and debates that are associated with them. To what extent ML models can generalize across different market conditions is a debatable topic. Sudah (2024) suggests that market crises can induce structural changes that can disrupt forecasting models and necessitate nimble approaches [43]. There is also debate regarding the morals of doing business using AI, including the possible effects on equity and market manipulation. ML models can also make market inefficiencies worse and create feedback loops that increase volatility if they are too good at anticipating stocks' direction. To what extent different sources of data, such as changes in luxury and basic good prices, would improve predictive accuracy is still open. While the behavior of consumers determines the stock performance, the relationship between S&P 500 price variation and buying patterns is not yet known. Empirical testing may subsequently be applied to check for correlation.

In addition, interest lies in whether it is possible to bring together world economic data and cross-market relationships. Because of the interconnection between financial markets, S&P 500 predictions can be improved when world economic tendencies are added. Finally, discussion exists concerning whether or not sentiment analysis works. Sentiment data can be used in predictions, according to research like Yu (2024) and Zhong (2021), but its usefulness depends on model calibration and data quality [47, 48].

Finally, the question of whether a variety of data sources could be utilized in conjunction with one another to provide a synergistic effect to prediction accuracy remains open. The literature we reviewed addresses various factors in isolation, which is good practice for measuring predictive power, but leveraging them in tandem may bridge gaps in earlier research to further improve trading strategies.

Continued innovation in NLP techniques and dataset optimization is necessary to overcome these limitations. Application of reinforcement learning to forecasting financial movements should be explored further through future research because it has the capability of building self-refining models that will learn how to adjust to changing market trends.

3 Methods

3.1 Data Processing

Technical Indicators

The core dataset for the project comes from the historical data section on Yahoo Finance [15]. The S&P 500 is listed under the ticker GSPC. We utilized the quantmod package in R [39] to webscrape the full historical records for the GSPC ticker. We then checked for missing values. Out of the columns 'Date', 'Close', 'Open', 'High', 'Low', 'Adjusted', and 'Volume', the only feature that was not full throughout the entire 1927 to 2025 record was 'Volume', which began in 1952. To account for this, we planned to keep the entire dataset, but drop the column with volume data when cultivating a model using the full record. When testing model creation with volume as a used feature, we would drop the 1927 to 1951 instances from a copy of the full DataFrame.

To build out the dataset to contain additional useful features, we utilized the TTR package in R [45] to extract the following technical indicators [9] from the stock data.

- Exponential Moving Average (EMA): This is an indicator that gives higher weight to the most recent data points. It provides a weighted average of a security's price over a set period. The period utilized was 20 days.

- Simple Moving Average (SMA): This indicator provides the average price over a set period. The period utilized was 20 days.

- Bollinger Bands (BB): These bands are a set of volatility indicators which encompass a Simple Moving Average (SMA), and the +/- two standard deviations around the 20-day SMA..

- Momentum (M): This is a measure of the momentum of a security that gives the price difference over n days (we used n=2).

- Price Rate of Change (ROC): This is an indicator that quantifies the momentum of a security by measuring the percent change between the current price and n days prior (we used n=2).

- Moving Average Convergence/Divergence (MACD): This is a signal indicator for buying or selling created by subtracting the 26-day EMA from the 12-day EMA

- Relative Strength Index (RSI): This is an oscillator that measures momentum based on the speed and magnitude of recent price changes. It is used to detect overbought or oversold conditions.

We also engineered two sets of target variables focusing on the S&P 500 price trends over the following 'n' days to allow for a better prediction goal from the model. Through our literature review, we had seen a similar approach from Basak et al [7], where they utilized trading windows of 3, 5, 10, 15, 30, 60, and 90 days. We pared down this list in part based on their success. 90 days presented far too much noise in the model due to complex market dynamics, making long-term predictions difficult.

- Price in n Days (PN): This is the value of the S&P 500 n days after each record. We used 10, 20, 30, 45, and 60-day values for n to produce 5 feature columns. Since this feature is forward-looking, this represents a potential target variable for more generalized prediction of positive or negative price movement.
- Percent Change in n Days (PCN): This is the percentage increase or decrease over the next n days. We used 10, 20, 30, 45, and 60-day values for n to produce 5 feature columns. Since this feature is forward-looking, this represents a potential target variable for more generalized prediction of positive or negative price movement.

Following our feature and target engineering, we dropped the NA values that had been incurred in the beginning and end of the dataset as a result of the technical indicators requiring some prior day data to construct and the target variables requiring some trailing day data to construct.

Macroeconomic Trends

Macroeconomic time series data were collected through the Federal Reserve Economic Data (FRED) API to support an analysis of key economic trends in the United States [6]. The dataset included a diverse set of indicators:

- Consumer Price Index
- Unemployment Rate
- Federal Funds Rate
- West Texas Intermediate Crude
- Average price: eggs
- U.S. Unemployment Rate
- Real GDP (percent change)
- Treasury Yield (10-Year)
- Gold Price (Daily from IMF)
- U.S. Dollar Index (Nominal Broad)

These variables were selected to represent a wide spectrum of economic activity and policy, spanning inflation, labor markets, monetary policy, commodity pricing, and financial markets.

Each time series varied in terms of historical coverage. For example, the CPI dataset contained monthly observations beginning in 1913, offering a long-term view of inflation trends in the U.S. In contrast, other series—particularly commodity prices like eggs and WTI crude oil—were only available beginning in the mid-2000s. This variation was taken into account during the data cleaning and pre-processing phase to ensure consistent alignment across periods used in the analysis.

Then, the dataset was assessed for missing values, gaps in temporal coverage, and inconsistencies across indicators. Descriptive statistics were computed for each series to better understand the range, central tendencies, and potential outliers. During the modeling phase in our investigation, additional features were engineered for each variable, including percent change (to capture relative growth or decline) and a 3-month rolling average (to smooth out short-term fluctuations and highlight underlying trends). These transformations provided a better basis for time series visualization and modeling tasks.

Missing data were handled using a forward-fill imputation method, which assumes the most recent known value remains valid until updated. This approach maintained continuity in the time series without introducing artificial trends. Overall, the resulting dataset offered a rich foundation for exploratory analysis, correlation studies, and potential predictive modeling of macroeconomic behavior.

Sentiment Analysis

To quantify and exploit the sentiment encoded in News Headlines, we utilized the Comprehensive Financial News Dataset in Time Series Description from Dong et al [12], a pre-curated dataset containing News Headlines from the financial sector going back to 2009.

The entire FNSPID dataset was downloaded from Hugging Face [20] to a local machine for analysis. An initial exploration involved taking a random sample of 10,000 records for exploratory data analysis (EDA). From the full local dataset, a new CSV file was created by extracting only those records with a `Stock_symbol` in the following list, which represent stocks which either are comprised of an investment mix mimicking the S&P 500 or are stocks which are within the mix of 500 companies comprising the index itself:

SPYU & SPLG & IVV & SPYG & SPYV
VOO & SPYD & VOOG & VOOV & XVV
SPXE & SPXN & SPXT & SPXV & SPY
EFIV & SNPE & SPMV & QVML & SPMO
SPVU & SNPG & SNPV & SPHQ & DSPY
SPDN & RSP & SCHD & IWD & VYM
JEPI & IVE & AAPL & MSFT & NVDA
AMZN & META & BRK.B & GOOGL & JPM
TSLA & SPXU & UPRO & SH & PSLD
SPHB & BAC & WFC & C & GS
MS & XOM & CVX & CAT & BA
JNJ & UNH & PFE & MRK & WMT
HD & COST & MCD & DIS & ORCL
ADBE & CRM & INTC & AVGO & SUSA
SPLV & VTI

Both the sample and the filtered subset were then processed through a tokenizer, with padding and truncation enabled. The tokenized headlines were subsequently evaluated using the FinBERT model [3], loaded from Hugging Face's ProsusAI/FinBERT repository. The model was configured to classify each headline into one of three categories—0: 'negative,' 1: 'neutral,' and 2: 'positive'—with a label encoder used to add a corresponding numeric label column. After classification, the sentiment scores were grouped daily, and the percentage distribution of sentiment labels for each day was calculated. For example, if three headlines appeared on a given day with two labeled negative and one positive, the daily sentiment percentages would be 66.7% negative and 33.3% positive. Additionally, the mean sentiment score per day was computed. To capture short-term sentiment trends, 3-day and 7-day rolling averages of the daily sentiment percentages for negative, neutral, and positive headlines were also calculated. The final feature columns generated from this process were:

- `sentiment_encoded`
- `positive_pct`
- `neutral_pct`
- `negative_pct`
- `positive_pct_3d`
- `neutral_pct_3d`
- `negative_pct_3d`
- `positive_pct_7d`
- `neutral_pct_7d`
- `negative_pct_7d`

3.2 Exploratory Data Analysis & Visualizations

Exploratory Data Analysis (EDA) was conducted on the various data sources separately to understand their distributions and tendencies to prepare the machine learning pipelines for full integration later.

Technical Indicators & S&P 500 Historical Data

After Data Preparation, the dataset was loaded into a Google Colab Jupyter Notebook for data exploration. Having just extracted the financial indicators and cleaned the dataset of instances containing missing values, we needed to see what patterns were present in the data to ascertain if the feature engineering had the potential to be predictive.

To begin with, we reviewed the S&P 500 history on a graph to view the overall movement of the index.

Additionally, we wanted to gain information on how the S&P 500 price movements lined up with the oscillations of the financial indicators derived from the data. To accomplish this, we initially graphed the two series in tandem with each other for the entire historical record available to us. The problem that we found with this approach was that due to the extensive history, the data was not directly viewable due to the oscillations being too compacted on a time-series view.

To separate the data enough for oscillations to be apparent, we looked at the same graphing approach, but only over the course of a single year. This allowed for much better investigation of the volatility of the index next to the oscillations of the financial indicators.

We also looked at a correlation matrix of all of the feature columns present in the dataset. To avoid overfitting, it was imperative to limit the correlated features [19, 30]. Especially since a number of the features extracted were momentum oscillators, we needed to know if some of those needed to be dropped before we began the modeling phase.

Finally, we looked at a variety of distributions over both the feature variables and the target variables. We needed to first decide on a target prediction period, such as 3 days versus 45 days. We also needed to know what a good threshold increase was to exploit in our swing trade strategy, such that there would be enough instances to make profitable trades at a frequent-enough basis to end up meaningfully profitable, but not so many as to inundate ourselves with endless buy and sell signals.

The reason for avoiding endless buy and sell signals is due to the fund transfer limitations present in retirement accounts. For example, in the U.S. government worker retirement plan, the Thrift Savings Plan, fund transfers are limited to no more than 2 per month. Additionally, micro-trades in funds that have transfer fees might not be profitable if the expected value from the trade is below the fee for making the transaction.

Macroeconomic Variables

This exploratory data analysis focused on examining the Consumer Price Index (CPI) in addition to key macroeconomic indicators, specifically the S&P 500 Index and U.S. labor force statistics. The analysis began by importing three primary datasets: CPI and purchasing power data from the Federal Reserve Economic Data (FRED), daily S&P 500 index values from Macrotrends, and employment and labor force data from the U.S. Bureau of Labor Statistics.

To align the datasets temporally, the daily S&P 500 index was resampled to a monthly frequency using the mean of daily closing values. A monthly period index was created for both the CPI and S&P 500 data to support merging operations. The S&P 500 index was also inflation-adjusted using CPI values, normalized to the first observation in the series, in order to evaluate real versus nominal asset performance over time.

The datasets were then merged based on their common monthly periods. The resulting combined dataset included nominal and real S&P 500 values alongside CPI figures. Labor force data were processed separately and included metrics such as total employment, unemployment, and population not in the labor force, with additional breakdowns by industry sector.

Data visualization was performed using time series plots. CPI and S&P 500 trends were presented together using dual-axis charts to show both nominal and inflation-adjusted index values over time. Labor force statistics were visualized across decades to highlight employment patterns and structural changes in workforce composition. All data handling and visualizations were implemented using Python libraries, including pandas, numpy, matplotlib, and seaborn.

News Sentiment Analysis

This analysis used a sample of financial news headlines from the FNSPID dataset to examine sentiment trends over time. An initial random sample of 10,000 unique headlines was extracted to ensure computational efficiency and remove duplicate entries. The dataset was cleaned and prepared using Python's `pandas` library, and any necessary type conversions, such as parsing dates, were performed to enable temporal analysis.

Sentiment labels were assigned to each headline using a pre-trained sentiment analysis model designed for financial text. These predictions were appended to the dataset for further exploration. The distribution of sentiment classes was then visualized to assess overall class balance through a bar chart.

To analyze temporal patterns, the dataset was grouped by week, and sentiment counts were aggregated over time. This allowed for the visualization of weekly sentiment trends within the financial news sample.

To examine the model's output confidence and score distributions, the raw logits from the FinBERT model were passed through a softmax activation function to obtain normalized probability scores across the three sentiment classes: negative, neutral, and positive. These scores were recorded for each input headline alongside the model's final predicted sentiment.

The resulting class probabilities were reshaped into a long-format structure, where each row captured the predicted sentiment, the sentiment class being scored, and the associated probability. This format allowed for comparison of how the model scored each sentiment class regardless of the final prediction. The data were visualized using a violin plot to illustrate the distribution and concentration of class probabilities grouped by predicted sentiment and score type.

In addition to full probability distributions, the model's confidence was assessed by isolating the maximum softmax score for each headline, representing the highest probability assigned to any class. These maximum scores were grouped by predicted sentiment and analyzed to evaluate how confident the model was across different prediction types. The distribution of these confidence values was visualized using a histogram with overlaid kernel density estimation.

4 Statistical & Machine Learning Methods

We aimed to explore a variety of Regression-based models as well as Classifier-based models. Among those tested were Linear, Logistic, Lasso, and Ridge Regressions, as well as Decision Trees, K-Nearest Neighbor, and XGBoost Classifiers.

For a swing trading strategy that is repeatable, we needed reasonably readable models. To that end, we did not employ a Standard Scaler in the model design phase due to the Scale Factor being difficult to employ on a daily basis moving forward.

Additionally, we needed to generate both 'Buy' and 'Sell' Models, as the target variable for each would be different. In reality, while Buy and Hold seem like a binary variable, the full picture is Buy, Sell, or Hold, which adds an additional dimension of complexity.

Because of the added complexity, each phase of model creation was centered around making a dual-model ensemble that would generate separate predictions for whether the target variable would indicate a profitable buy or sell action.

To evaluate the models, we would initially review the test-set confusion matrices for Classifier-based models or the Root Mean Squared Error (RMSE) and R^2 (R-Squared) variables for Regression-based models.

Following a review of the accuracy of the models, we set up swing trading simulations to employ the derived models on the test set. The goal of this was to see if the model created had a positive or negative expected value, and to quantify the improvement over a buy-and-hold approach, which was the baseline by which we judged the predictive power.

While prior research had concentrated purely on accuracy metrics to evaluate stock price prediction, we recognized that a profitable swing trading strategy would rely less on calculated accuracy and more on profitable predictions. If, for example, a stock was predicted to grow 2% over the next 30 days but in reality it would go down by 1%, the trade would result in a loss despite the fairly low root error of the instance. Likewise, confusion matrices don't tell the full story of the profit or loss. In generating a buy and sell signal, only the True Positive and False Positive Rates drive your profit or loss, respectively. Furthermore, we eventually set our target variable for classifier problems to instances greater than 3.5% or less than -3.5% growth. In this case, a confusion matrix tells even less of a story, because False Positives may include still-profitable trade actions that just failed to classify a 3% growth as a negative entry.

To truly capture the effect of the models we created, we set up the aforementioned simulations to test the profit or loss of the model. Our simulation would begin at the start of the dataset or test set, and each day, the predictions from the ensemble model would be examined. The simulation would begin with exactly 1 stock in the account, and attempt to swing trade it to increase the amount of stock in the account. Conversely, the baseline comparison would begin with 1 stock and simply allow it to grow. In so doing, the ending stock balance gives a multiplier for the efficacy of the trained model. The result of the signals encoded is shown below:

- Buy Signal is True & Sell Signal is True: If both signals are positive, we did not make any transactions, as that indicated an instance poorly represented by the model.
- Buy Signal is False & Sell Signal is False: This event is a very clear 'Hold' signal, and as such, no transaction was taken.
- Buy Signal is True & Sell Signal is False: In this event, if cash was on hand from a prior sale of stock, all cash was converted to stock at the current value of the index. If no cash was on hand, no action was taken.
- Sell Signal is True & Buy Signal is False: In this event, if stock was on hand from a prior purchase of stock, all stock was sold at the current value. If no stock was on hand, no action was taken.

To further refine the simulation and make it more relevant to retirement investors, additional testing was done wherein the current month and a trade counter were tracked. Each time a trade was made, the counter would be increased by 1. Each time the month changed, the counter would be reset to 0. If the counter reached 2, no further trades were permitted until the counter was reset.

5 Results

5.1 Exploratory Data Analysis & Visualizations

Technical Indicators

To get an understanding of the distribution of our potential target variables, we utilize the violin plot method from pyplot to display the distribution of percentage growth over four groups of periods, n=3, 10, 30, and 45. This should give us an idea of how the spread of outcomes changes over time. We are looking for a significant enough spread away from the 0% baseline to reasonably expect the model to have meaningful classifications and for the transactions to be profitable enough to make up for either inter-fund transfer fees or monthly limits.

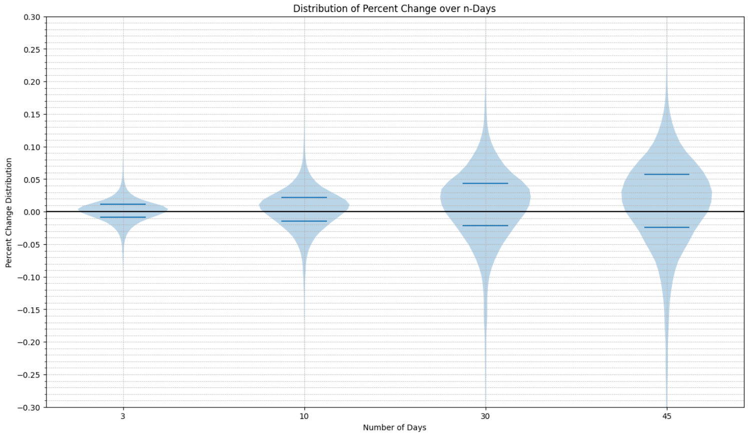

Figure 1: Distribution of Percent Change in S&P 500 over Multiple Time Horizons. Violin plots show return distributions over 3, 10, 30, and 45 days.

As seen from the figure, nearly all of the outcomes of the 3 and 10-day distributions are within ±5%, which limits the ability of the model to highlight outlying, profitable transaction opportunities. The 30-day distribution has significantly more spread. The 45-day has even more wide of a distribution, but as seen in past studies [7], longer-term predictions must deal with significant noise from complex market dynamics and fluctuating sentiment over extended periods.

With a target variable time-period decided upon, n=30 days, we must begin to attend to the feature variables to ascertain their value in our modeling efforts. To begin with, we graphed the exponential moving average (EMA) and the moving average convergence divergence (MACD) financial indicators alongside the GSPC Closing Price to look for whether the signals commonly used in the investment community predict price movements.

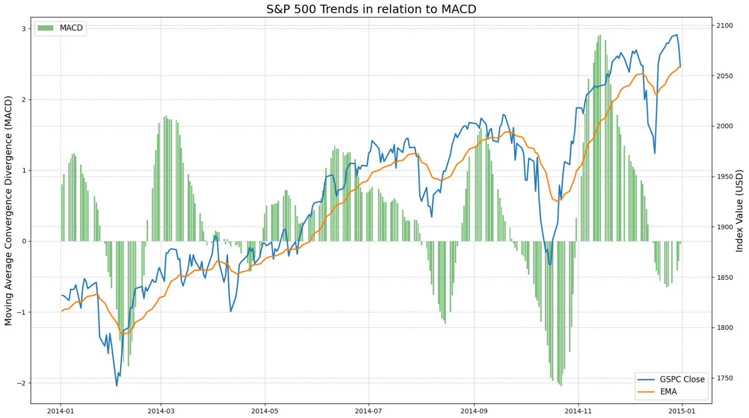

Figure 2: This chart shows the S&P 500 (in blue) alongside the Exponential Moving Average (in orange) overlayed over the Moving Average Convergence Divergence indicator (in green).

We chose a time period of 2014 to allow for a reasonable amount of recovery time from the 2008 housing crisis to be representative of typical behavior, a recent enough observation year to be useful in the context of current generalization of the model, and due to the year occuring in the middle of a presidential term, where speculation on changing market dynamics will be at a minimum. In short, we were looking for a boring market year to see standard indicator performance.

Looking at the MACD (in green) buy and sell signals, where the MACD crosses the 0 line, we note that the GSPC Closing Price (in blue) begins it's price correction just before the buy or sell signal, but that there is significant further correction occurring after those signals are seen.

Looking at the EMA (in orange) buy and sell signals, where the Closing Price goes from below the EMA to above it (and vice versa), we note that the signal happens a day or two prior to the MACD signal on average. That said, the EMA and MACD both appear to be predictive of further price corrections which should make them viable candidates for machine learning model exploitation.

To quantify the lift provided by the EMA and MACD signals, we run our first test simulations, wherein we do a sample trading strategy based solely on first the EMA and then the MACD to make stock purchases or sale when presented with a signal of coming price swings. No monthly limits were imposed on this strategy.

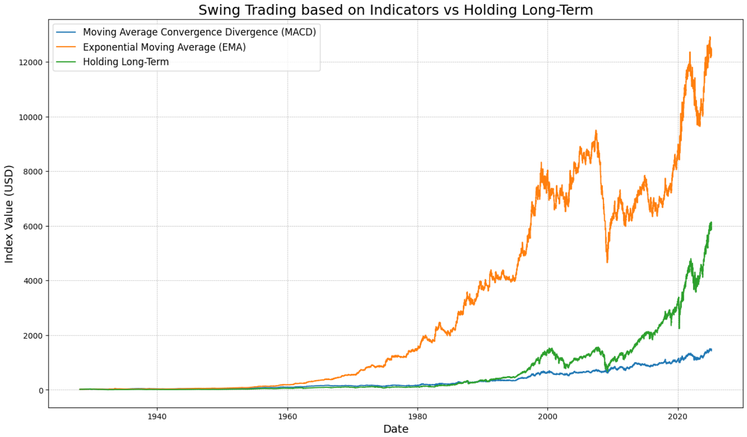

Figure 3: This chart shows the results of simulations of trading action based on single financial indicators.

The exponential moving average swing trading system is shown in orange, the buy-and-hold strategy is shown in green, and the moving average convergence divergence is displayed in blue. Note that the EMA strategy doubles the investment roughly over the course of 1927 to 2025. This gives support to the predictive power in the EMA signal. One downside to this strategy is the significant loss in 2008 from the housing crisis.

The MACD line, on the other hand, suffers to maintain value and in fact ends at a fraction of the starting stock number. This may be due to the lag in signal identification seen on the prior graph showing the 2014 trends.

Based on this EDA, the EMA is assessed to be a higher fidelity predictor of price movements in a timely manner than the MACD line.

Following this analysis, we explored the Relative Strength Index (RSI) in the same graphical paradigm as the EMA/MACD graph before it. Signal lines used by traders to signal overbought or oversold conditions in the Relative Strength Index are the red horizontal lines at 70 and 30, respectively. The orange line remains the EMA, which proved predictive in the recent test.

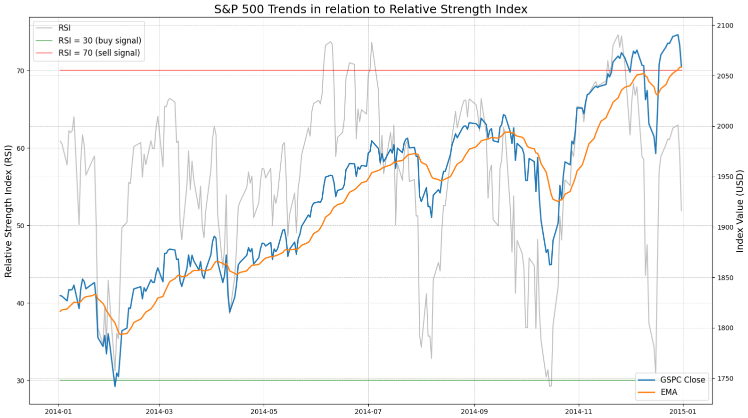

Figure 4: This graph shows the S&P 500 price movements over the Relative Strength Index (RSI) indicator in grey. Bounding lines at 30 and 70 RSI represent the buy and sell signals, respectively.

We note the relative infrequency of the signal lines being met, but that when they are met a price correction follows as indicated by the financial theory.

We aim to keep the RSI as a parameter in the final model. An additional feature is the 2 day Price Rate of Change (ROC). To explore this feature, we first wanted to look at the distribution therein to gain an understanding of what constitutes a significant value. The ROC gives a measure of the momentum of the equity.

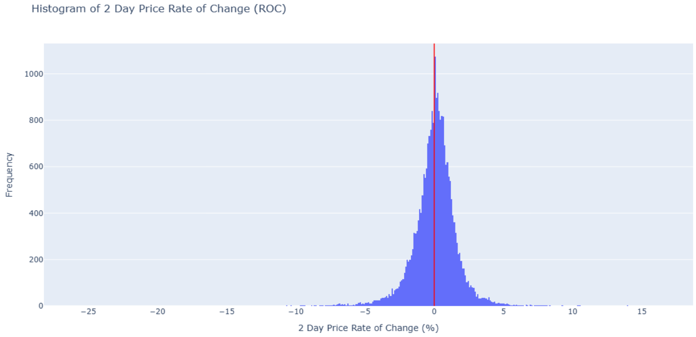

Figure 5: This histogram shows the distribution of 2-day price changes as a percentage of the value 2 days prior.

The 2-day ROC is largely constrained within the ±5% bounds. Viewing the ROC as a good measure of whether or not price movement is too sharp to continue long-term, we engineer an additional four feature variables. These combine RSI and ROC into whether or not the RSI signal has been met, and whether the ROC is positive or negative.

- FallingROC70RSI
- FallingROC30RSI
- RisingROC70RSI
- RisingROC30RSI

By labelling these specific situations in a series of binary variables, we aim to give the model a way to explicitly call on these situations to better classify situations in which we should buy or sell stock.

Finding correlations with our feature variables is vital to ensure a robust model capable of generalization that is resistant to overfitting. To accomplish this, we reviewed the correlation matrix for the features thus far described within the S&P 500 dataset to look for offending variables.

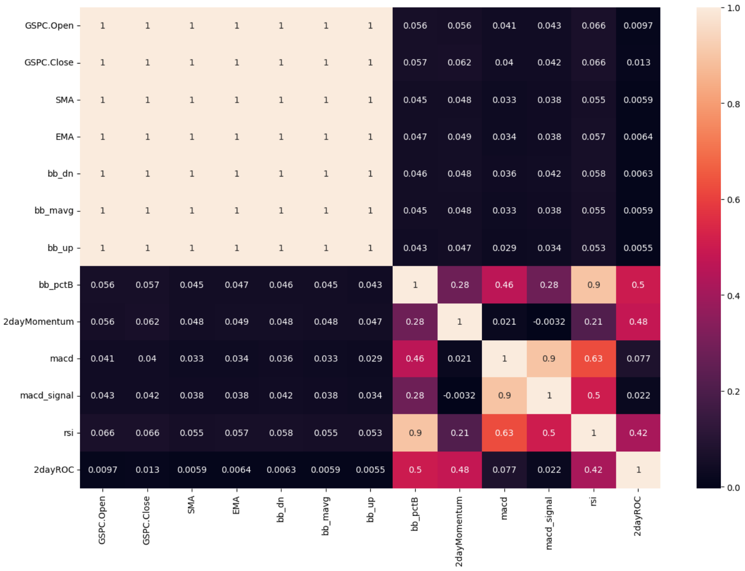

Figure 6: This heatmap shows the correlations between the various feature variables present in the dataset.

Reviewing this data, we found Bollinger Bands, a financial indicator not previously tested, were 90% correlated with RSI, which was found to be highly predictive in our exploration of the data. To avoid overfit problems, we dropped the Bollinger Band columns from the dataset, instead opting to retain the relative strength index.

We also found the MACD signal variable, a secondary variable extracted using the TTR package in R [45], to be overly correlated with the MACD variable itself. We kept the base MACD variable and removed the signal variable.

Other than that, the variables in our dataset were now disagregated to a reasonable degree and we could proceed with model curation, pending our additional exploratory data analysis of CPI and Sentiment data.

Consumer Price Indices

After pre-processing the consumer price indices data, we moved into an exploration of the trends underlying the dataset. We begin with looking at the purchasing power of the U.S. Dollar, shown below.

The loss of purchasing power, exponential since 1940, has wide-reaching implications for consumers, businesses, investors, and financial market indices such as the S&P 500.

Next, steps were taken to align the datasets in terms of structure and frequency. Since some data sources were reported monthly while others, like the S&P 500, were daily, we needed to resample and synchronize the timeframes for accurate comparison. Following the timeframe alignment, we plotted the following, showing the relationship between the S&P 500 and the Consumer Price Index (CPI) from 1950 to 2025, highlighting distinct behavioral patterns between the two. Its movement resembles a simple moving average (SMA), particularly a long-term one such as a 12-month or 36-month SMA, as it smooths out short-term fluctuations and shows the overall inflation trajectory over time.

We followed this step by examining the relationship between the GSPC Closing Price and the Consumer Price Index over time. The CPI began rising in the early 1970s, while S&P 500 shows a continuous rise,

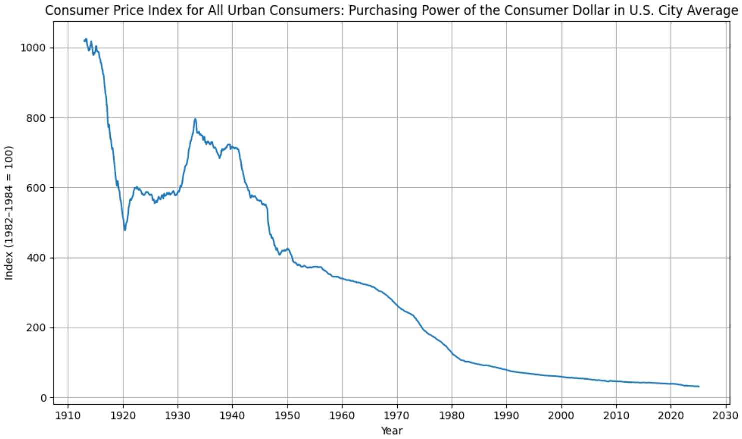

Figure 7: The decrease in consumer price index for all urban consumers measured by the purchasing power of the consumer dollar.

especially since the 1980s. The graph shows that, while the S&P 500's value has continuously risen, the inflation has also continuously risen, beginning early and maintaining a near-linear trend compared to the S&P.

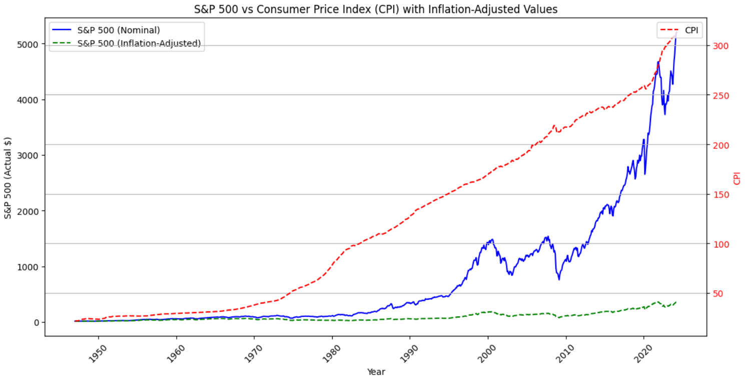

Figure 8: Comparative trends of the S&P 500 and U.S. Consumer Price index (1950-2024) showing market performance and inflation trends.

Attending the green line at the bottom, we see that while the raw price of the S&P 500 has grown nearly exponentially in recent years, the inflation-adjusted price rises at a much more gradual, and linear rate. This highlights part of the challenge of forecasting stock prices over lengthy time scales, and lends credence to our decision to pursue a percentage growth approach rather than a raw price approach.

Next, we look at the unemployment statistics from the U.S. Bureau of Labor Statistics [36] to look for any trends which might be predictive for the dataset.

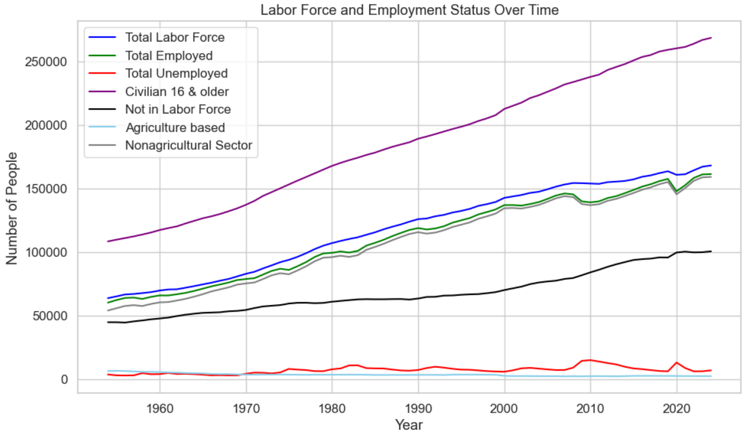

Figure 9: The line graph showing labor force and employment status over time.

While there are certainly some fluctuations, especially around the 2008 housing crisis and the 2020 coronavirus outbreak, the data is largely gradual and linear.

We also investigated other Economic Indicators, such as the price of Oil, to look for interesting fluctuations. We found oil to be particularly volatile, as seen in the image below:

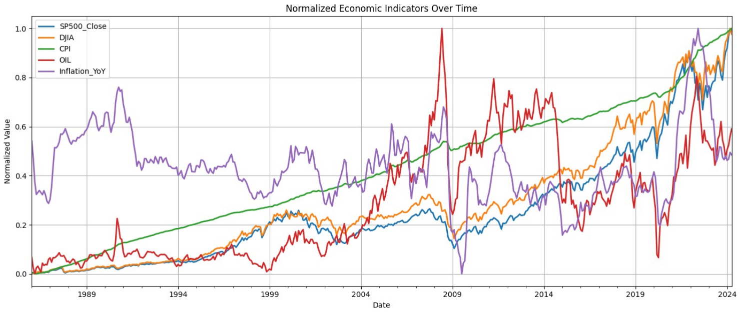

Figure 10: This graph shows the S&P 500 across similar time-scales as other important Economic Indicators. Oil, in red, is particularly volatile.

Note the severe price increase of oil during the 2008 housing crisis [24]. This could be a good benchmark for a solid sell signal, as increasing oil prices increase consumer cost of living as well as cost of doing business for the corporations leading the market and being represented by the S&P 500 Index.

Likewise, the drop in oil pricing in the late 2010's precedes notable market growth. Looking at these fluctuations, it is clear that oil pricing has a tangible effect on market behavior and could be a great feature to include in predictive modeling.

This prediction is further supported by performing a Linear Regression on S&P 500 returns and highlighting the most important features. For this model, only macroeconomic and S&P 500 features were included, and Oil ranks as number 5 overall with an importance of 15%.

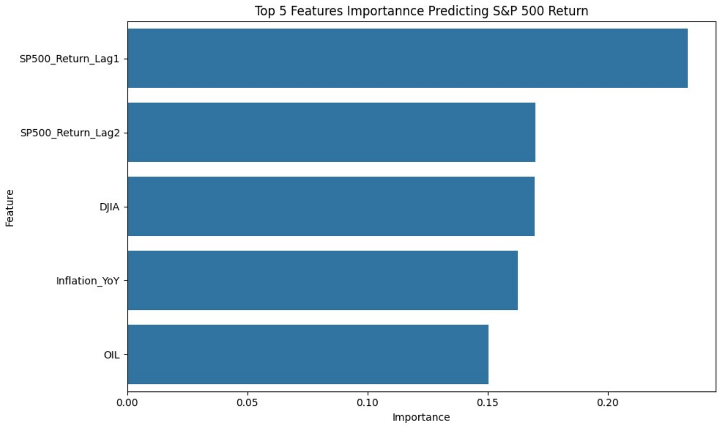

Figure 11: This shows the most important macroeconomic features impacting the S&P 500 returns.

News Headlines & Sentiment Analysis

Once pre-processed and classified using FinBERT [3], we began exploring the FNSPID dataset by looking at the overall distribution of predicted sentiments. A bar plot, shown below, reveals the proportion of headlines categorized as positive, neutral, and negative.

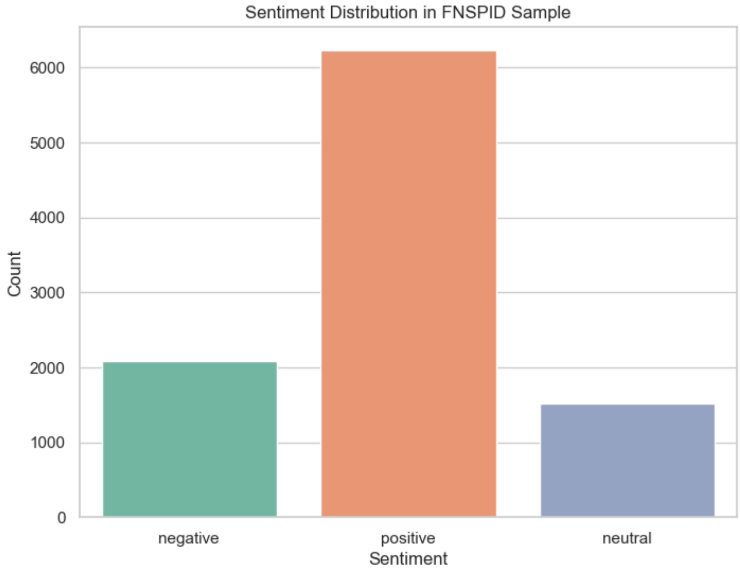

Figure 12: Bar chart showing the distribution of predicted sentiments of FNSPID sample data

Overall, we found that the dataset was skewed towards more positive sentiments, with over 6,000 positive headlines compared to around 2,000 negative and 1,500 neutral entries. This indicates a general optimism or bullish tone in financial reporting. It may also reflect FinBERT's known tendency to assign higher probabilities to positive sentiment, especially in weakly polarized contexts [3, 17].

Next, we explored how sentiment trends have evolved. This is a very important area to look at, as changes in media behavior may follow what is likely to get the most attention, rather than what classifies the news most accurately.

To do this, we aggregated the sentiment on a weekly basis and plotted the results to identify fluctuations in the prevalence of each class. These trends not only reveals how public sentiment has varied during key periods, but also if there are media-centric behavioral changes that may affect the generalization-ability of the resulting models we engineer.

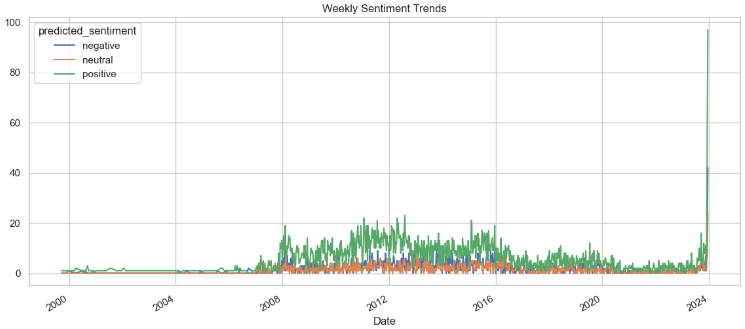

Figure 13: Weekly sentiment trend lines derived from FinBERT predictions on sampled FNSPID headlines.

The time series plot reveals temporal patterns in sentiment distribution:

- Positive sentiment was consistently dominant from 2010 to 2013, possibly reflecting post-recession recovery and general market optimism. It also showed a notable spike during mid to late 2024, which may correspond to recent bullish trends or positive macroeconomic events.
- Neutral sentiment remained relatively stable throughout the period, with no major fluctuations, indicating a consistent presence of non-polarized reporting.
- Negative sentiment exhibited a slight increase in 2024, suggesting a rise in market concerns, risk-oriented reporting, or volatility-triggered pessimism during that time frame.

To assess FinBERT's confidence in its sentiment classifications, we examined the full class probability vectors output by the model. These include probabilities for each of the three classes—positive, neutral, and negative—for every headline. This provides insight into not only the model's prediction, but also how confident it was in that prediction.

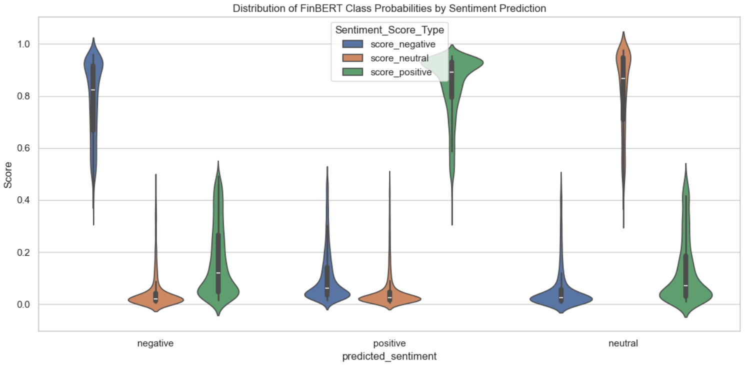

Figure 14: Violin plot of FinBERT's class probability distributions by predicted sentiment label.

Figure 14 presents a violin plot of these class probabilities, disaggregated by the final predicted sentiment label. This allows us to see the distribution of probabilities associated with each label and assess how "decisive" the model tends to be. For instance, predictions labeled as positive are often accompanied by high positive class probabilities, but there is some degree of overlap between sentiments. The violin plots illustrate that when a particular sentiment is predicted, the corresponding probability score is typically high while the scores for other sentiments remain low. Positive sentiment predictions again stand out for their sharp and high-centered distribution, suggesting FinBERT is more decisive when classifying a headline as positive.

In addition to analyzing the full probability distributions, we also computed the maximum class probability for each prediction, representing FinBERT's confidence level.

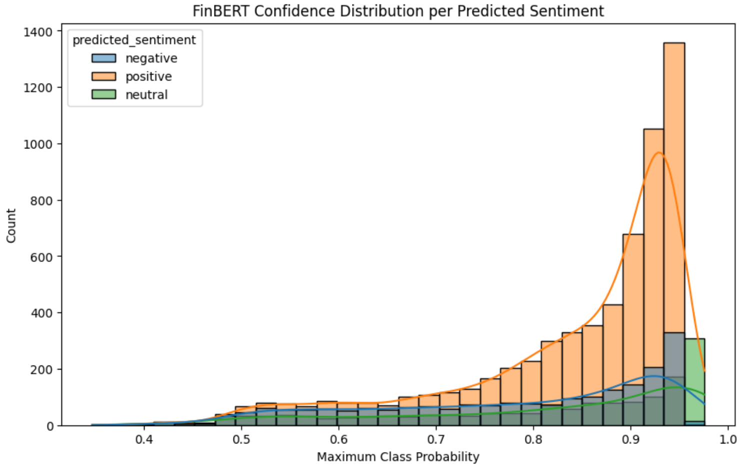

Figure 15: Histogram of maximum class probabilities (confidence scores) by predicted sentiment.

Figure 15 presents a histogram of these confidence values. This plot provides a clearer picture of how confident the model is across different sentiment categories. High confidence (scores closer to 1.0) suggests the model is more decisive, while a broader distribution suggests some predictions were more ambiguous. The distribution of maximum class probabilities shows that positive sentiment predictions have notably higher confidence scores, often clustering around 0.9 to 1.0. Meanwhile, confidence scores for negative and neutral predictions are more widely distributed and generally lower, highlighting areas of uncertainty in sentiment classification, particularly for non-positive classes.

5.2 Analysis, Statistical, & Machine Learning

At the outset of the study, several machine learning models were tested on the dataset, including Linear Regression, Logistic Regression, Lasso, Ridge, XGBoost, Decision Tree, and k-NN to determine which algorithm performed best. Almost all of the regression-based models demonstrated low accuracy and poor overall performance. While the Decision Tree initially appeared promising, it experienced significant overfitting as a result of a lack of proper model constraints. XGBoost also showed encouraging early results. The overfitting issue with Decision Tree became apparent when training accuracy reached 96%, while test accuracy lagged at 76%. After setting a maximum depth of 5, the accuracy for both training and test slices dropped to 82%. The ability of Decision Trees to model non-linear relationships with minimal preprocessing makes them robust for modeling and prediction in financial market trends [2], however, the performance after correcting the overfitting still left something to be desired. XGBoost, on the other hand, resisted the overfitting problem without model tweaks and reported 87% accuracy for both training and test sets. XGBoost is also recommended for long-term forecasting, given its capacity to handle complex relationships within the data, and has been used in prior studies of financial prediction learning [47].

Both Decision Tree and XGBoost achieved an accuracy score of about 85%. However, a significant issue emerged upon closer examination because both models exhibited a low True Positive Rate (TTR) when predicting up and down trends. After extensive testing, it was found that setting the threshold too low led to excessive classification of signals which diluted precision and hindered the model's ability to identify truly exploitable opportunities.

Random Train/Test Split: A Time-Series Shortfall

Initially, we wanted to ensure that our model did not rely overmuch on any particular era of governmental policy or market behavior. Due to the nature in which regulations shape company strategies, over time these policies can shift, resulting in market dynamics that are no longer representative of present-day. Furthermore, for a model to be useful long-term, it should be resistant to influences of this type.

To tackle this problem, we first decided to do an 80/20 random Train/Test split. We could still measure accuracy of predictions, we reasoned, on a test set randomly selected just as well as we could a test set chronologically sampled.

To begin the modeling phase, after extracting the features mentioned previously in this paper, we created a new dataset of only those features we wished to create a model on, and initiated our split. The accuracy of prediction in regression-based models was poor, as we expected, given the noise-filled nature of such a dynamic system. Accuracy of classifier predictions was also notably low in a positive or negative outcome classification at the 30-day mark, which prompted us to create threshold target variables where the classification would attempt to answer not whether the result was growth or loss, but whether the result was growth or loss over ±3.5%. This metric was created upon rigorous analysis of distributions in percentage growth at the 30-day mark and is a hyperparameter of our model creation. Many values between 2 and 5% were tested, and 3.5% gave the most predictions with the fewest False Positives.

Upon re-orientation of the classification problem, our model, utilizing XGBoost Classifier and K-Nearest Neighbors, was able to deliver an accuracy of 86% on the Test Set. Looking at the accuracy on the Train Set, the accuracy was 87%. This gave us high confidence that our model was not overfitting, and we moved on to the simulation phase in our plan.

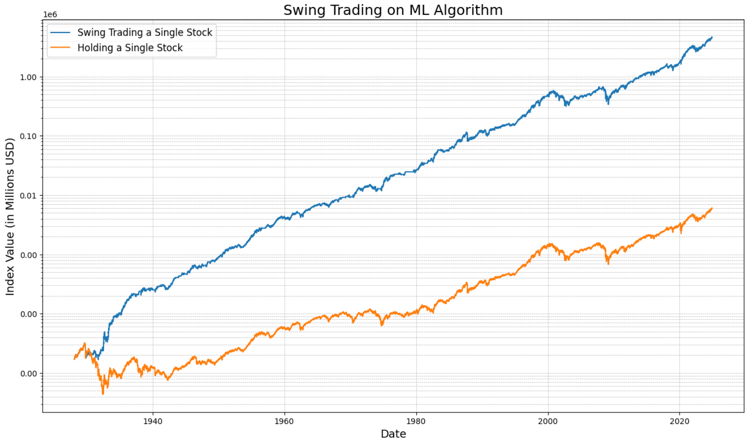

Figure 16: Simulation of Trading Actions based on Machine Learning modeling since 1927. Results in a logarithmic scale for ease of examination.

After a basic simulation, even with limiting trades to 2 per month to ensure a representative performance, we were getting millions of dollars out of the initial investment of $17.04, compared to the Index Fund, which rose to only $6K.

This result was way beyond what we expected to see, and this prompted a thorough examination of the model, our simulation code, and in the end: our Train/Test split methodology.

We realized rather quickly that there was no way for us to rigorously test the model result in a random Train/Test split using our designed experiment simulation. Profitable instances in the train set could be overwhelmingly poor or even lost from the test set predictions. To truly ascertain value, we needed to rethink our design of the experiment to allow for rigorous testing. This brought us back to our previously dismissed Chronological approach of splitting the data. The authors caution against random splits in a Time-Series-centric project as a result of this false positive result.

Chronological Train/Test Split for Rigorous Quantification of Results

In keeping with the idea of an 80/20 split of the dataset, we calculated from the dataset's length the 80% mark, which was 2003, and set up the training set to be from 1927 to 2003, with the test set comprised of 2004 through 2025.

We integrated the same modeling pipeline as before, and we had a very similar accuracy score from the Classifiers. Once again, the Regression methods failed to deliver worthwhile performance.

For this phase of testing, both XGBoost and K-Nearest Neighbors were tested for predictive power while maintaining the same feature variables. The confusion matrices showed a high False Positive Rate for both modeling types, which indicates high potential for loss, or at the very least, suboptimal profit below the threshold. To gather more data on the expected value, we compared the 3.5% predictions to a positive or negative outcome column labeled 'nDaysPos' and 'nDaysNeg'. While the False Positive Rate was still fairly high for 'nDaysNeg' in relation to the Sell Signal, the 'nDaysPos' in relation to the Buy signal showed significantly lower False Positive Rate and a huge jump in True Positive Rate. These confusion matrices are shown below.

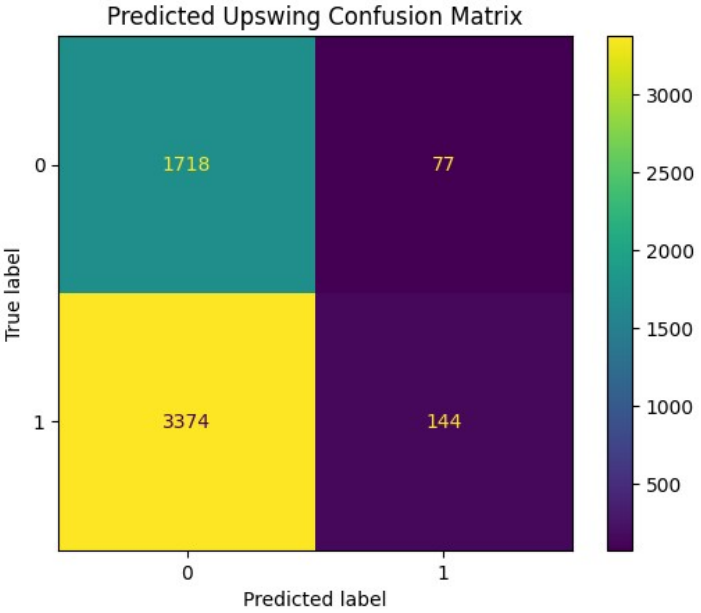

Figure 17: Confusion Matrix displaying the outcome of the Buy Signal, predicted on a positive percentage movement of 3.5% compared to a simple positive change. True Positive Rate indicates positive profit while False Positive Rate indicates negative value. False Negative Rate indicates missed opportunities for swing trades, while True Negative Rate indicates the profitable decision not to buy, or re-enter, the position.

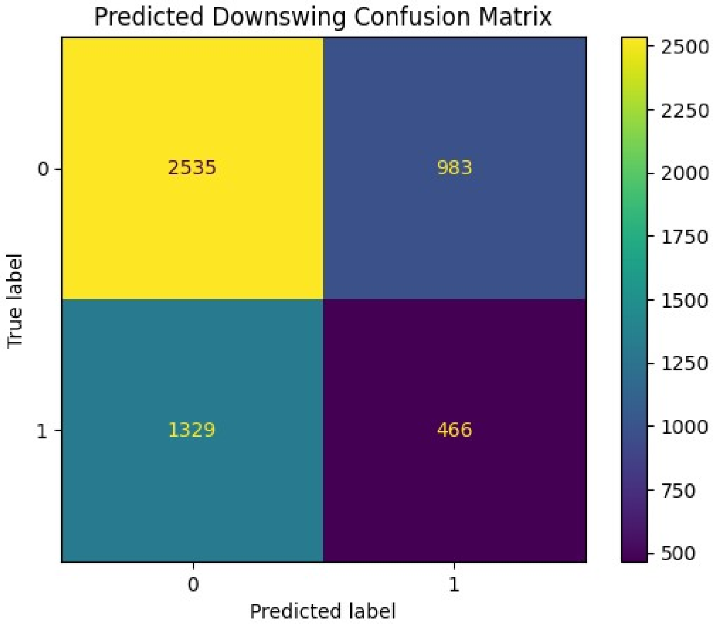

Figure 18: Confusion Matrix displaying the outcome of the Sell Signal, predicted on a negative percentage movement of 3.5% compared to a simple negative change.

Viewing the Upswing confusion matrix in Figure 17, we note the high True Positive Rate to False Positive Rate, which is a good sign for profitable trading actions. The Downswing confusion matrix in Figure 18, on the other hand, has nearly double the False Positives as True Positives.

As mentioned previously, the True and False Positives represent the profit and loss for the model, while the True Negative represents a correct holding decision, and the False Negative represents lost-profit opportunities.

Due to our classification being at a threshold of ±3.5%, we hoped that the False Positives might be mostly still profitable, just to a more marginal extent than if the threshold set were reached.

After reviewing the performance on the test set and considering how our experiment was set up to ensure robustness, we put the model through a simulation once again. The resulting growth chart is shown below.

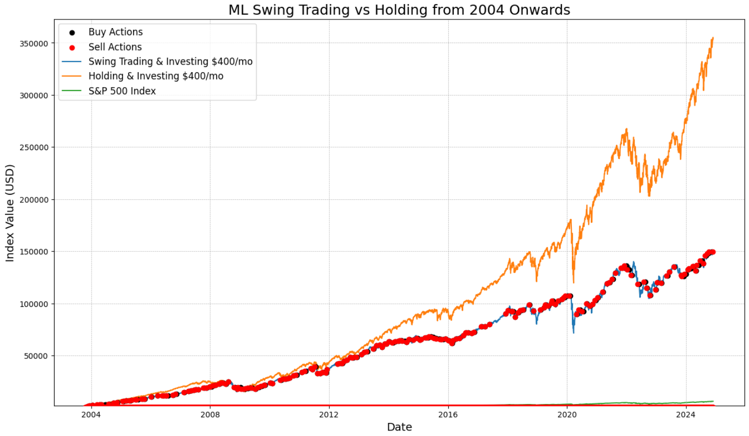

Figure 19: Simulation of Trading Actions based on Machine Learning modeling since 2004. Results in a linear scale. Red and Black dots along the blue line indicate taken buy and sell signals.

Not only did the results not live up to the earlier random split outcome, but our swing trading model ended up losing a significant amount of stock value over the 21 years. This told us two things: (1) the previous result was a result of training set positives outperforming test set negatives, resulting in unrealistic outcomes, and (2) technical indicators would not be enough to produce a usable, profitable, long-term swing trading strategy.

Sentiment & Macroeconomic Integration

To enhance the model's predictive capabilities beyond purely technical analysis, additional macroeconomic and sentiment features were incorporated. The goal was to capture broader economic forces and psychological factors [28, 29, 33] influencing financial markets, thereby improving the robustness and generalization capability of the trading strategies.

Recognizing that financial markets are influenced not only by fundamental economic realities but also by collective investor psychology, sentiment analysis was used to capture shifts in market mood that might precede price movements, especially during uncertain or turbulent periods. Market movements are dependent upon various factors ranging from political events, firms' policies, economic background, commodity prices, exchange rates, movements of other stock markets, to the psychology of investors [21]. Daily financial news headlines were scraped and processed through established sentiment scoring libraries, producing measures that categorized headlines as positive, negative, or neutral. Aggregated daily sentiment scores were constructed, and additional engineered features such as 3-day and 7-day rolling averages of sentiment were computed to detect emerging trends in investor sentiment.

A baseline simulation focusing solely on technical indicators from 2004 onward revealed that swing trading, without sentiment integration, performed significantly worse than simply investing $400 monthly in a buy-and-hold strategy (Figure 19). This was likely due to market dynamics, which shift with time, resulting in a non-representative training set, but it was necessary to be used as an accurate comparison to the following models. Furthermore, governmental policies affecting market dynamics will continue to bring challenges to the task of stock prediction, and any machine-learned swing trading strategy must be resistant to debilitation from such news. To address this, sentiment features were incrementally introduced into the model. To begin testing this regime, we reverted to the original target variables temporarily to assess a comparative performance between the original target and the re-engineered one. The addition of a daily aggregated sentiment score improved predictive performance, with swing trading achieving a final value of $12,462 compared to $14,377 for buy-and-hold (Figure 20). Further improvements were seen when incorporating the 3-day and 7-day sentiment rolling averages (Figure 21). Under this configuration, swing trading ended with a final value of $13,616, narrowing the performance gap with the passive strategy.

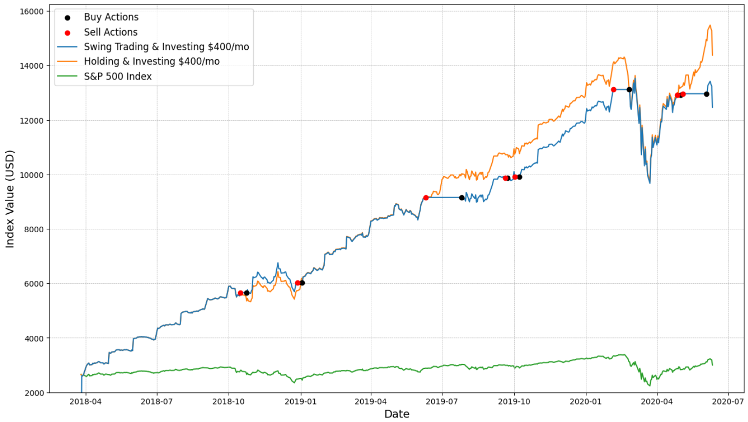

Figure 20: ML Swing Trading vs. Holding with Monthly Contributions (2018–2020): During this sentiment regime, the holding strategy begins to outperform the machine learning swing trader, which misses some upside following conservative sell signals.

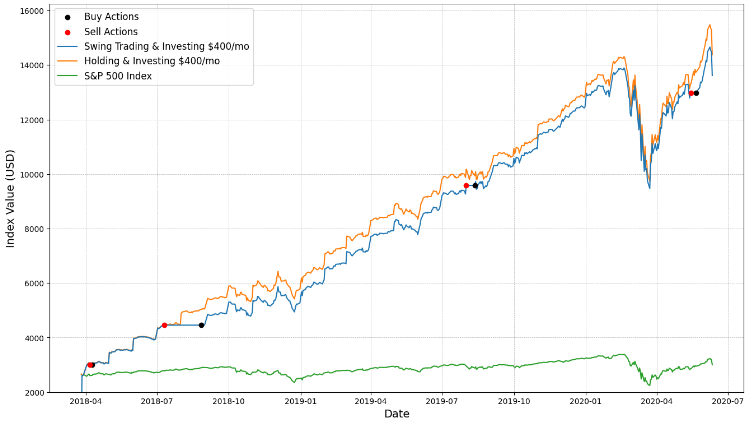

Figure 21: Lagged Sentiment Scenario (2018–2020): Here, the ML swing trading model stays more actively engaged in the market, resulting in performance that tracks more closely with the buy-and-hold strategy, though still trailing slightly.

Additionally, alternative problem formulations were explored to leverage sentiment data more effectively. The "v2" version of the target variable was utilized now, redefining the successful signal as a rise or fall of more than ±3.5% at any point within the next 30 days, rather than over fixed shorter windows (Figure 22). Under this revised target definition, swing trading achieved a final value of $13,705, slightly closer to the $14,377 value for buy-and-hold but still not exceeding it.

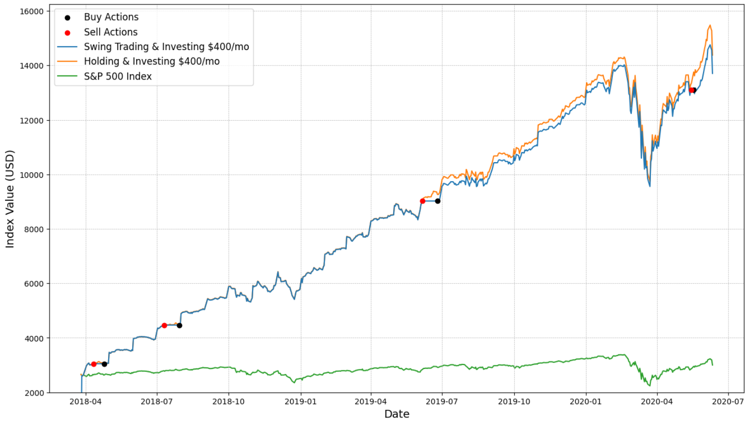

Figure 22: ML Swing Trading vs. Holding Strategy (2004 Onwards): A closer look at model performance during alternative target conditions, showing occasional underperformance of swing trading.

These results are significantly better than the technical indicators alone, but to go further, we needed to incorporate the macroeconomic variables to provide our growing model with the broadest picture of the economy and environment of the S&P 500 as we could.

Historical data was obtained using the Federal Reserve Economic Data (FRED) API [6]. Several key indicators were selected based on their established relevance to market performance. These included the Consumer Price Index (CPI), the U.S. Unemployment Rate, the Federal Funds Effective Rate, Crude Oil Prices (WTI), Gold Spot Prices, the U.S. Dollar Index (DXY), and the 10-year Treasury Yield. Rather than using these indicators in their raw form, the project constructed meaningful derivative features aimed at highlighting the underlying trends and volatilities. Percent changes were calculated over various timeframes, such as monthly and quarterly periods, to capture the momentum and directionality of macroeconomic conditions. Rolling averages were engineered to track trend shifts more smoothly, while rolling volatilities were computed to detect periods of economic stability or instability that might precede market turbulence.

Initial simulations using only raw macroeconomic variables showed limited improvement over purely technical models. Swing trading using raw macroeconomic inputs achieved a final value of only $8,566, compared to $14,377 for a simple buy-and-hold strategy (Figure 23). However, after incorporating the engineered features, such as monthly percent changes and 3-month moving averages, the performance of the swing trading strategy improved significantly. With these enhanced macroeconomic features, swing trading's final value rose to $18,904, although it still trailed the buy-and-hold value of $30,502 (Figure 24). This result demonstrated that thoughtful feature engineering was essential to extract meaningful predictive power from macroeconomic data.

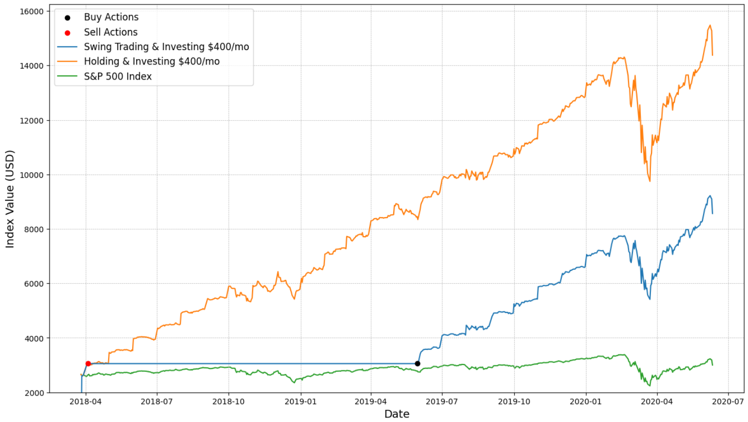

Figure 23: Raw Macroeconomic Data (2018–2020): The ML swing trading model, trained solely on contemporaneous macroeconomic indicators, underperforms the holding strategy.

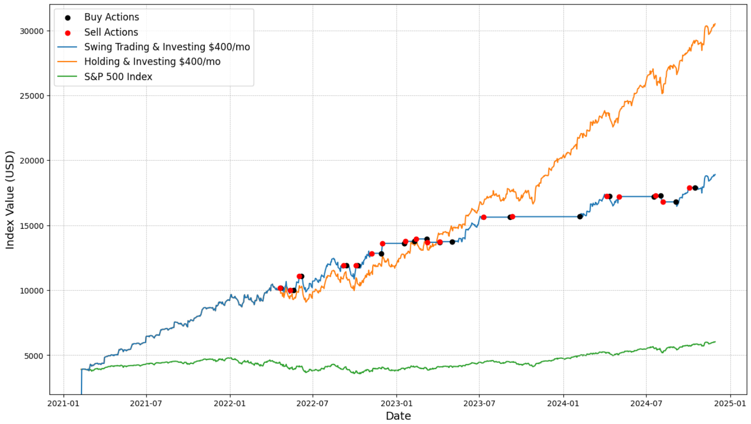

Figure 24: Lagged Macroeconomic Features (2021–2025): Incorporating lagged macroeconomic variables improves model timing, but the swing trading strategy still underperforms the holding approach.

Full Integration

After extensive feature engineering across technical, sentiment, and macroeconomic domains, the final model aimed to integrate these elements into a unified predictive framework. The final dataset incorporated technical indicators such as moving average crossovers (EMA and MACD), momentum measures like two-day Rate of Change (ROC), and Relative Strength Index (RSI) thresholds coupled with ROC signals. Sentiment features were based on daily news headline scoring, generating daily sentiment encodings and three-day and seven-day rolling averages of positive, neutral, and negative sentiment percentages. Macroeconomic data from sources such as the Federal Reserve Economic Data (FRED) were processed into monthly percentage changes and three-month trend indicators for variables including CPI, Unemployment Rate, Federal Funds Rate, Treasury yields, oil prices, gold prices, and the U.S. Dollar Index.

The dataset was partitioned into training and testing sets, with approximately 80% of observations (covering roughly 3000 data points) allocated for model training and the remainder reserved for evaluation.

Initially, the machine learning models were trained to predict binary target variables based on short-term market movements. Specifically, nDaysDown and nDaysUp outcomes were defined based on whether the S&P 500's closing price experienced a directional move over the following 30 days. If the index showed a positive or negative change across the full 30-day window, this was recorded as an upward or downward move, respectively. The K-Nearest Neighbors (KNN) Classifier was first applied. On the training data, the KNN Classifier produced relatively high accuracy for predicting nDaysDown, but testing results showed limitations, particularly in detecting true positive cases. In contrast, when the XGBoost Classifier was deployed, predictive performance improved substantially. On the test set, the XGBoost model correctly classified a majority of downtrend cases (nDaysDown), achieving a strong balance between precision and recall. For nDaysUp predictions, the model captured a meaningful portion of upward market movements but faced more challenges, reflecting the noisier nature of positive sentiment in financial markets. Trading simulations were then conducted based on these model predictions (Figure 25). Following a constrained swing trading regime—limiting trades to no more than two per month and assuming monthly $400 contributions—swing trading achieved a final portfolio value of approximately $28,634. In comparison, a buy-and-hold investor over the same period would have ended with approximately $28,133. Thus, even under realistic trading constraints, the machine learning-guided swing trading strategy very slightly outperformed simple passive investing. Furthermore, the swing trading portfolio demonstrated lower maximum drawdowns during major market corrections, reflecting a degree of built-in defensive positioning driven by model sell signals.

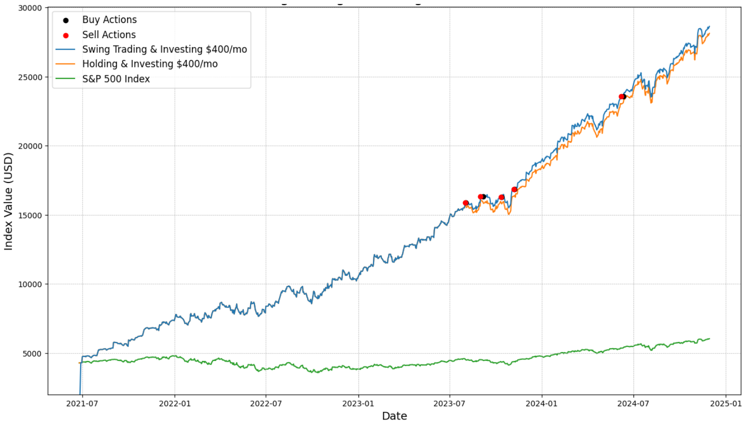

Figure 25: Comprehensive Model (2021–2025): The ML swing trading model demonstrates stable growth with reduced drawdown exposure but ultimately lags behind the buy-and-hold strategy during an extended upward trend.

Recognizing that market movements often include significant short-term fluctuations rather than clean directional trends over fixed windows, a refined target structure was proposed. In this version, referred to as “v2,” the target variable defined success as the market experiencing a rise or fall exceeding ±3.5% at any point within the subsequent 30 days, rather than based on the final closing value after 30 days. This approach better reflected the opportunities and risks faced by active traders who must react to intraperiod volatility. Models were retrained using the same feature set but substituting the nDaysUp_v2 outcome for the original nDaysUp variable. Again, both KNN and XGBoost classifiers were applied. Training accuracies remained high, but true positive rates became more balanced as the v2 target allowed the model to detect meaningful price swings more flexibly. When trading simulations were conducted using the “v2“ target variable-derived models, the results were less impressive than the original target variable when all features were integrated into the model. The final swing trading portfolio value reached approximately $22,880, noticeably under-performing the $28,133 buy-and-hold benchmark and the original target variable-derived model.

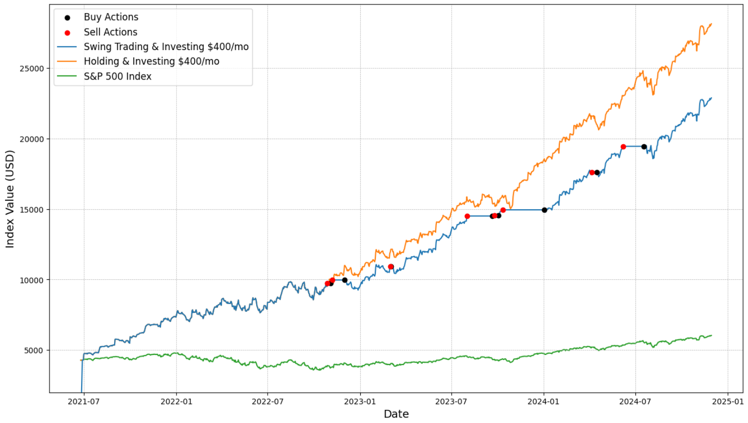

Figure 26: Comprehensive Model Using Alternative Target Variables (2021–2025): This version of the ML swing trading strategy, trained on alternative target definitions, exhibits slower growth compared to the holding strategy.

6 Discussion

While the incremental integration of technical indicators, sentiment analysis, and macroeconomic data improved model performance, several concerns remain. The main concern among them is the model's inability to significantly outperform a basic buy-and-hold strategy. Although combined technical indicators with sentiment data showed better returns than using sentiment data alone, overall results still lagged behind the passive investment benchmark [49]. It is often said in financial circles that time in the market is better than timing the market, and this so far has rung true in our testing.

One of the most significant challenges in this study was designing a robust training and testing methodology across nearly a century of historical market data. Due to the time-dependent nature of financial markets, we had to abandon random data splitting in favor of a time-series aware approach. As the feature set grew in complexity, especially because of the addition of macroeconomic and sentiment data, the risk of model instability increased. Additionally, the inclusion of numerous variables increases the risk of multicollinearity and computational challenges, necessitating careful feature selection and dimensionality reduction techniques [14].

Another challenge in this study was identifying features that could reliably capture underlying market trends. Effective feature selection is important for real-time stock trading, and the unpredictability and volatility of the stock market render it challenging to make a substantial profit using any generalized scheme [44]. Despite feature engineering efforts, model performance in correctly classifying buy and sell signals remained inconsistent. Additional testing involved introducing more complex logic in the simulation, such as restricting trades based on recent sell or buy prices, or initiating trades based on price movements exceeding defined thresholds even in the absence of clear model signals.

Another main concern is the model's ability to generalize across different asset classes. Since all of the models in this study were trained based on historical data from the S&P 500 index, it remains uncertain whether the current feature would yield comparable performance when applied to other asset categories such as bonds, commodities, and international equities.

Additionally, the update frequency of macroeconomic data and the limited historical availability of sentiment data present a fundamental limitation. Many economic indicators are reported monthly, quarterly, or annually, whereas financial markets operate daily. For instance, unemployment data was originally reported yearly, dating back to 1954 by the U.S. Bureau of Labor Statistics, and the same annual figure was repeated across all months within each year to match the S&P 500's monthly frequency. As a result, the statistical inference of unemployment status and S&P 500 shows no relationship-assuming that unemployment alone does not explain the movements of the S&P 500. On the other hand, since sentiment features have only been available from 2004 onward, the potential for long-term back testing of fully enriched models is inherently limited. This misalignment poses challenges for forecasting models, as low-frequency macroeconomic releases struggle to keep pace with the high-frequency nature of financial market volatility and investor decision-making [16]. Although the model partially addressed this mismatch through the use of rolling averages and derivative features, it remains difficult to fully eliminate the lag effect inherent in macroeconomic data and limitations posed by the limited historical availability of sentiment indicators.

Operational and transaction costs were not modeled within the trading simulations, representing another area of caution. Although trading frequency was constrained, real-world implementation could still, depending upon the institution, entail brokerage fees, taxes, and slippage, all of which could meaningfully reduce net performance relative to theoretical in-simulation results. Transaction costs can substantially impact trading performance, especially when considering multiple assets with varying costs and volatilities [32].

Finally, while the model outperformed a buy-and-hold strategy during known periods of market stress-such as the 2008 Financial Crisis-it remains vulnerable to true black swan events: extreme and unpredictable shocks that fundamentally disrupt market dynamics. Machine learning models, by their nature, are built on historical patterns, and their ability to respond effectively to unprecedented situations remains a significant open question. Analytical models often fail to provide expected fidelity during statistically unlikely events, with errors increasing by several hundred percent [22].

7 Conclusions

Preliminary analysis of the S&P 500 dataset using financial indicators has demonstrated encouraging predictive potential. Among these, the Exponential Moving Average (EMA) demonstrated strong potential, indicating that further application of ML techniques and feature engineering could effectively exploit its buy/sell signals for improved forecasting accuracy. However, Moving Average Convergence/Divergence (MACD) performed relatively poorly. The exploration of macroeconomic data revealed certain patterns, though many appeared smoothed due to the reporting frequency and lagging nature of key indicators. For instance, consumer price indices are often reported weekly or monthly, and their impact on labor and broader economic conditions tends to unfold gradually. The largest oscillations in labor data occurred during the major economic downturns, such as the 2008 financial crisis and the 2020 COVID-19 pandemic. The unemployment rate rose from a low of 4.4 percent in May 2007 to a high of 10.0 percent in October 2009 [38], and significant downward trends in the S&P 500, reflecting the broader economic impact of the 2008 financial crisis. In addition, changes in the Consumer Price Index (CPI), which also reflect inflation levels, significantly influenced medium-term trends in S&P 500. As a result, both the Consumer Price Index (CPI) and S&P 500 exhibited an upward trajectory over time, with an inflation-adjusted value of the index also showing a gradual and linear increase.

News and sentiment analysis provided additional ways for forecasting large-scale market collapses, such as the 2008 financial crisis. Preliminary analysis of the sentiment dataset revealed significant positive skew. When plotted on a time series, the data appears to have spikes around U.S. Presidential Elections. 2008 saw a large spike in positive sentiment despite the housing crisis that was kicking off. While sentiment alone may not be highly predictive, there is potential value in applying machine learning techniques to the tokenized text data. Specific keywords or phrases may prove to be useful indicators of future market movements, particularly if they consistently precede negative or positive price action. Overall, the dataset exhibited a strong skew toward positive sentiment, suggesting a general tone of optimism in financial news reporting. In general, sentiment extracted from financial news articles may contribute meaningfully to understanding trends in the S&P 500, but with limitations.

Overall, based on the results, a machine learning model that integrates technical indicators, sentiment, and CPI data has the potential to meaningfully predict S&P 500 price movements, though challenges remain. Of all models tested, XGBoost showed promising accuracy (around 87%) and performed well without overfitting. This suggests that integrating multiple data sources, including technical indicators, sentiment, and the Consumer Price Index (CPI), can be useful in predicting S&P 500 price movements, but the model must be carefully calibrated to address challenges like noise, overfitting, class imbalance, and the sequential nature of market behavior.

References

[1] M. Abelson. Options traders bet on more s&p 500 turbulence after cpi data. *Bloomberg*, 2025. Accessed April 5, 2025.

[2] M. Ahmed and I. Mohammed. How do different machine learning models compare in their ability to predict short-term movements in the s&p 500 index?, 2024.

[3] D. Araci. Finbert: Financial sentiment analysis with pre-trained language models, 2019.

[4] B. Bah. Predicting the movement of the s&p500 index using machine learning. *Jouranl of Financial Analysis*, 10(2):100–120, 2023.

[5] S. Banik, N. Sharma, M. Mangla, S. N. Mohanty, et al. Lstm based decision support system for swing trading in stock market. *Knowledge-Based Systems*, 239:107994, 2022.

[6] F. R. Bank. Consumer price index for all urban consumers: All items in u.s. city average, 2025.

[7] S. Basak, S. Kar, S. Saha, L. Khaidem, and S. R. Dey. Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and Finance*, 47:552–567, 2019.

[8] O. Bassal. *Swing trading for dummies.* John Wiley & Sons, 2019.

[9] J. Chen. Technical indicator: Definition, analyst uses, types and examples, 2021. A series on Technical Indicators and their uses.

[10] N.-F. Chen, R. Roll, and S. A. Ross. Economic forces and the stock market. *Journal of business*, pages 383–403, 1986.

[11] C. Chia-Cheng, C. Chun-Hung, and L. Ting-Yin. Investment performance of machine learning: analysis of s&p 500 index. *International Journal of Economics and Financial Issues*, 10(1):59, 2020.

[12] Z. Dong, X. Fan, and Z. Peng. Fnspid: A comprehensive financial news dataset in time series. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4918–4927, 2024.

[13] K. Du, F. Xing, R. Mao, and E. Cambria. Financial sentiment analysis: Techniques and applications. *ACM Comput. Surv.*, 56(9), Apr. 2024.

[14] J. Fan and R. Li. Statistical challenges with high dimensionality: Feature selection in knowledge discovery. In *Proceedings of the international Congress of Mathematicians*, volume 3, pages 595–622. European Mathematical Society Zurich, 2006.

[15] Y. Finance. S&P 500 (GSPC) historical data, 2025. Historical Records of SP 500 Price.

[16] E. Ghysels, P. Santa-Clara, and R. Valkanov. Predicting volatility: getting the most out of return data sampled at different frequencies. *Journal of Econometrics*, 131(1-2):59–95, 2006.

[17] I. Goncharov. Financial sentiment analysis on stock market headlines with finbert huggingface, Sep 2021.

[18] L. Harris. S&p 500 cash stock price volatilities. *The Journal of Finance*, 44(5):1155–1175, 1989.

[19] D. M. Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

[20] S. M. Jain. Hugging face. In *Introduction to transformers for NLP: With the hugging face library and models to solve problems*, pages 51–67. Springer, 2022.

[21] Y. Jiao and J. Jakubowicz. Predicting stock movement direction with machine learning: An extensive study on s&p 500 stocks. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 4705–4713. IEEE, 2017.

[22] E. Kaminski. *The Limits of Analytics During Black Swan Events A Case Study of the Covid-19 Global Pandemic*. PhD thesis, Massachusetts Institute of Technology, 2021.

[23] W. Kenton. S&p 500 index: What it's for and why it's important in investing, Jun 2024.

[24] L. Kilian and C. Park. The impact of oil price shocks on the us stock market. *International economic review*, 50(4):1267–1287, 2009.

[25] C. Krauss, X. A. Do, and N. Huck. Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the s&p 500. *European Journal of Operational Research*, 259(2):689–702, 2017.

[26] S. Latif, F. Aslam, P. Ferreira, and S. Iqbal. Integrating macroeconomic and technical indicators into forecasting the stock market: A data-driven approach. *Economies*, 13(1):6, 2024.

[27] S. B. Meeradevi and B. Swetha. Evaluating the machine learning models based on natural language processing tasks. *Int J Artif Intell ISSN*, 2252:8938, 1955.

[28] A. Mittal and A. Goel. Stock prediction using twitter sentiment analysis. *Standford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis. pdf)*, 15:2352, 2012.

[29] S. Mohan, S. Mullapudi, S. Sammeta, P. Vijayvergia, and D. C. Anastasiu. Stock price prediction using news sentiment analysis. In *2019 IEEE fifth international conference on big data computing service and applications (BigDataService)*, pages 205–208. IEEE, 2019.

[30] O. A. Montesinos López, A. Montesinos López, and J. Crossa. Overfitting, model tuning, and evaluation of prediction performance. In *Multivariate statistical machine learning methods for genomic prediction*, pages 109–139. Springer, 2022.

[31] S. M. Mostafavi and A. R. Hooman. Key technical indicators for stock market prediction. *Machine Learning with Applications*, 20:100631, 2025.

[32] S. Murthy and J. K. Wald. Optimal trading with transaction costs and short-term predictability. *Quantitative Finance*, 23(7-8):1115–1127, 2023.

[33] T. H. Nguyen, K. Shirai, and J. Velcin. Sentiment analysis on social media for stock movement prediction. *Expert Systems with Applications*, 42(24):9603–9611, 2015.

[34] Noble Gold Investments. Cpi vs wage growth: What the data tells us. https://noblegoldinvestments.com/cpi-vs-wage-growth/, 2024. Accessed April 5, 2025.

[35] A. Odell. Luxury no longer means quality: Consumers weigh in on the slowdown. *Vogue Business*, 2023. Accessed April 5, 2025.

[36] U. B. of Labor Statistics. Databases, tables calculators by subject, 2025. Data on Labor Statistics by Category.

[37] M. F. Osborne. Brownian motion in the stock market. *Operations research*, 7(2):145–173, 1959.

[38] J. Rothstein. The great recession and its aftermath: What role for structural changes? *RSF: The Russell Sage Foundation Journal of the Social Sciences*, 3(3):22–49, 2017.

[39] J. A. Ryan and J. M. Ulrich. quantmod: Quantitative financial modelling framework, 2025. R package version 0.4.27.1.

[40] S. Sakaria. *Predicting directions of S&P 500 using AI & ML models.* PhD thesis, Brunel University London, 2024.

[41] A. F. Sheta, S. E. M. Ahmed, and H. Faris. A comparison between regression, artificial neural networks and support vector machines for predicting stock market index. *Soft Computing*, 7(8):2, 2015.

[42] N. Sim and H. Zhou. Oil prices, us stock return, and the dependence between their quantiles. *Journal of Banking & Finance*, 55:1–8, 2015.

[43] R. Sudah. Sectoral behavior to crises: an analysis of the s&p 500 economic sectors. Master's thesis, Itä-Suomen yliopisto, 2024.

[44] A. Ullah. Effective feature selection for real-time stock trading in variable time-frames and multi criteria decision theory based efficient stock portfolio management. 2021.

[45] J. M. Ulrich. Ttr: Technical trading rules, 2025. R package version 0.24.4.

[46] P. R. Winters. Forecasting sales by exponentially weighted moving averages. *Management science*, 6(3):324–342, 1960.

[47] S. Yu, Q. Zhang, and Y. Zhao. S&p 500 trend prediction. *arXiv preprint arXiv:2412.11462*, 2024.

[48] S. Zhong and D. B. Hitchcock. S&p 500 stock price prediction using technical, fundamental and text data. *arXiv preprint arXiv:2108.10826*, 2021.

[49] H. Zhou. Profits of trading strategies based on market sentiments and technical analysis. *Global Business & Finance Review*, 14(2):104, 2009.

8 Code Appendix

Code Repository: https://github.com/IsaacTeal2025/DAT490-Capstone