Using LSTM extracted feature for trading strategy construction

COMP4971F - Independent Work

December 2023

HUANG, Luyi

Supervised by Dr. David Rossiter

Department of Computer Science and Engineering, HKUST

Abstract

In recent years, the combination of machine learning models and trading strategies has gained significant attention in stock trading. This paper explores the integration of decision tree-based models with the LSTM-AM model to determine optimal positions. Notably, our approach differs from traditional methods by manually labeling the optimal daily positions in the training set, which are then learned by the testing set. Additionally, instead of manually extracting features from past stock prices using technical indicators, we leverage the feature vector generated by the LSTM-AM model. We compare the effectiveness of this feature vector with baseline OHLCV values and close-prices-only feature vectors. Data analysis shows that our model, particularly when combined with LSTM-AM, yields higher returns and performs well on individual stocks compared to the buy-and-hold strategy. In contrast, the baseline strategy often fails to function effectively for individual stocks. This outcome suggests that our algorithm effectively incorporates stock volatility, unlike baseline methods. Furthermore, our paper demonstrates the feasibility of using machine learning algorithms to extract technical indicators, surpassing manual methods.

Contents

1 Introduction 3

2 Related Work 4
2.1 Technical Indicators 4
2.2 Technical Indicator Related Strategies 5
2.3 Tree Based Algorithm 6
2.4 LSTM 7
2.5 Attention Mechanism 8

3 Data 10
3.1 Selection of Stocks 10
3.2 Data Preprocessing 10

4 Methodology 11
4.1 Marking the Optimal Buying and Selling time 11
4.2 Extracting the Feature Vector 11
4.3 Classification Tasks 11

5 Result Analysis 12
5.1 Information Technology Sector 12
5.2 Health Care Sector 14
5.3 Consumer Discretionary Sector 16
5.4 Financials Sector 18
5.5 Materials Sector 20
5.6 Industrials Sector 22
5.7 Utilities Sector 24
5.8 Real Estate Sector 26

6 Conclusion 27

7 Future Aspects to Explore 28

8 References 29

1 Introduction

In the past few decades, there has been a growing interest in predicting stock prices and developing effective trading strategies, given their stochastic and highly volatile nature influenced by various qualitative factors. Consequently, numerous approaches have been developed to analyze stock price data and enhance trading outcomes.
Traditional methods encompass fundamental and technical analysis. Technical analysis has introduced several technical indicators, including the relative strength index, moving averages, and momentum oscillator. These indicators are used to generate trading signals based on assumptions of mean reversion (the stock price deviating from its long-term average and returning) and general stock trends (long-term bullish or bearish movements). Feature engineering, which involves extracting technical indicators, has proven to be effective. However, the manual selection of optimal parameters from a large number of indicators can be laborious. Furthermore, the effectiveness of technical indicators can vary as stocks and markets change, posing a challenge despite their wide applicability.

With the advancement of machine learning, various time series models have been adopted to address price prediction challenges. Among these models, LSTM is a widely used and classical approach. It can capture long-term dependencies and extract features from historical data, making it suitable for processing financial data. Researchers have also combined LSTM with other models, such as incorporating 1D-CNN and attention mechanism layers to improve prediction accuracy. However, LSTM-related models can suffer from overfitting to older data, limiting their ability to generalize to new data.

While price prediction models are useful, stock trend prediction models may be more suitable for trading decisions since they directly influence the decision-making process. In a study mentioned in a paper, researchers explored support vector machines, random forests, and neural networks to predict one-day movements in closing prices. Although accuracy varied across models, it mostly ranged from 52% to 58%, indicating that using predicted trends to determine the next day's positions may not be the optimal profit strategy. Additionally, transformer models were introduced in the same paper to predict stock movements over different periods. While transformers were found to be more effective at capturing long-term trends than LSTM, significant improvements in accuracy were primarily observed over longer periods, such as monthly intervals. Considering the substantial price fluctuations that can occur over extended periods, long-term investments may not align with our trading strategy. These limitations of relying solely on stock trend predictions inspire us to explore more reasonable approaches.

To take advantage of the benefits of the aforementioned models without incorporating their disadvantages, we introduced a novel method that would automatically extract feature vectors and self-learn optimal trading strategies using classification models.

2 Related Work

Various approaches are employed by people to predict stock trends and represent stock features. Traders use these indicators to inform their trading decisions. The subsequent section will provide a concise overview of these algorithms and indicators.

2.1 Technical Indicators

Here are examples of some basic technical indicators:

Moving Averages:

Moving averages (MA) are widely used technical indicators that help filter out short-term price fluctuations and provide a clearer picture of the underlying trend. They calculate the average price of an asset over a specific period.

There are different types of moving averages, including simple moving averages (SMA), exponential moving averages (EMA), and weighted moving averages (WMA).

The SMA is the basic form of a moving average. It calculates the average closing price of an asset over a specific number of periods and assigns equal weight to each period.

The EMA gives more weight to recent price data, making it more responsive to changes in price compared to the SMA. It applies a smoothing factor (often represented as a percentage) to the previous EMA value and the current price to calculate the new EMA. A commonly used smoothing factor is 2 / (n + 1), where 'n' represents the number of periods.

Relative Strength Index(RSI):

The Relative Strength Index (RSI) is a popular momentum oscillator used to assess the strength and speed of price movements. It compares the magnitude of recent price gains to recent price losses over a specified period, typically 14 periods. Thus, It helps traders identify overbought and oversold conditions in an asset and can provide potential buy or sell signals.

The resulting RSI value ranges from 0 to 100. Typically, an RSI reading above 70 is considered overbought, indicating a potential price reversal or correction to the downside. Conversely, an RSI reading below 30 is considered oversold, suggesting a potential price reversal or correction to the upside.

Moving Average Convergence Divergence (MACD):

The Moving Average Convergence Divergence (MACD) is a popular trend-following momentum indicator that helps traders identify potential buy and sell signals. It measures the relationship between two exponential moving averages (EMAs) of different periods

MACD consists of several components:

MACD Line = 12-Day EMA - 26-Day EMA

The MACD line represents the difference between these two EMAs and provides insights into the strength and direction of the trend.

Signal Line = 9-Day EMA of MACD Line

The signal line helps identify potential buy and sell signals. When the MACD line crosses above the signal line, it generates a bullish signal (buy). Conversely, when the MACD line crosses below the signal line, it generates a bearish signal (sell).

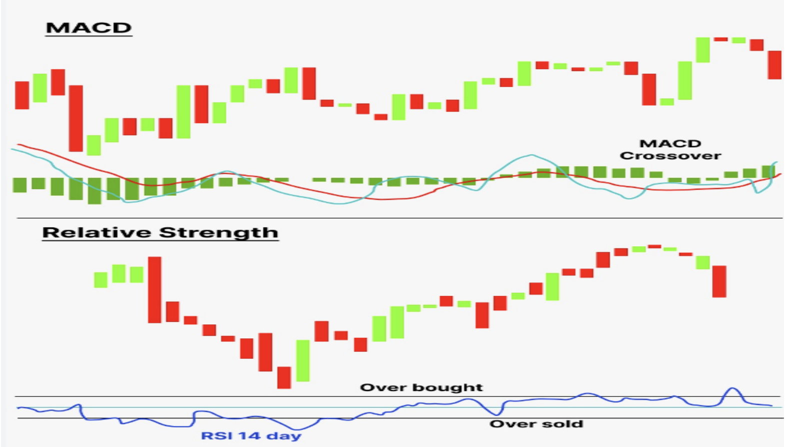

Figure 1: Implementation of MACD strategy

2.2 Technical Indicator Related Strategies

Trend Following Strategy – Bias:

The trend-following bias strategy using moving averages aims to identify and capitalize on the prevailing trend in the market. It involves using two moving averages to determine the bias and generate trading signals. Traders confirm the bias by observing the relationship between a shorter-term moving average (e.g., 50-day SMA) and a longer-term moving average (e.g., 200-day SMA). Entry signals align with the bias, such as buying on retracements in a bullish bias or selling short on retracements in a bearish bias. Proper risk management, including setting stop-loss and take-profit levels, is essential. Traders can also trail stops to protect profits as the trade progresses.

Mean Reversion Strategy – Bollinger Brand:

Bollinger Bands are a technical analysis tool used in mean reversion strategies. They consist of three lines plotted on a price chart: a middle band (usually a 20-period moving average) and upper and lower bands based on standard deviations. Traders use Bollinger Bands to identify overbought and oversold conditions in the market. When the price touches or exceeds the upper band, it may be considered overbought, signaling a potential selling opportunity. Conversely, when the price touches or falls below the lower band, it may be considered oversold, indicating a potential buying opportunity. Bollinger Bands should be used in conjunction with other indicators and analysis techniques for making trading decisions.

It's important to note that mean reversion strategies including Bollinger Brand are not foolproof and can be subject to risks. For example, a prolonged trend or fundamental change in the market can cause prices to deviate from their historical averages for an extended period, resulting in potential losses for mean reversion traders.

Using only technical analysis may not be adequate for profitable trading, as it can lag behind the actual trend. Lagging indicators like MACD and Bollinger Bands often start reflecting the trend after a substantial portion of the movement has already occurred. Additionally, human biases can introduce variations in analysis, leading to different insights even when analyzing the same chart. Therefore, it is important to consider technical analysis as a reference rather than a definitive predictor for trader

2.3 Tree-Based Algorithms

Random Forest:

Random Forest is an ensemble learning algorithm that combines multiple decision trees to make predictions. It is used for both classification and regression tasks.

The main idea behind Random Forest is to create an ensemble of decision trees, where each tree is trained on a different subset of the training data. This is achieved through bootstrap aggregating. Random subsets, known as bootstrap samples, are created by randomly sampling the training data with replacement. Each bootstrap sample is used to train a decision tree. Random Forest introduces additional randomness by considering only a random subset of features at each node of the decision tree during the split, helping to decorrelate the trees. When making predictions, Random Forest aggregates the predictions of all the individual trees. In classification tasks, the class with the majority of votes from the trees would be chosen. Random Forest helps to mitigate overfitting by combining multiple trees and reducing the variance of the predictions. It is robust to outliers and noisy data and can handle high-dimensional datasets and large feature spaces effectively. It also provides a measure of feature importance, which can be used for feature selection.

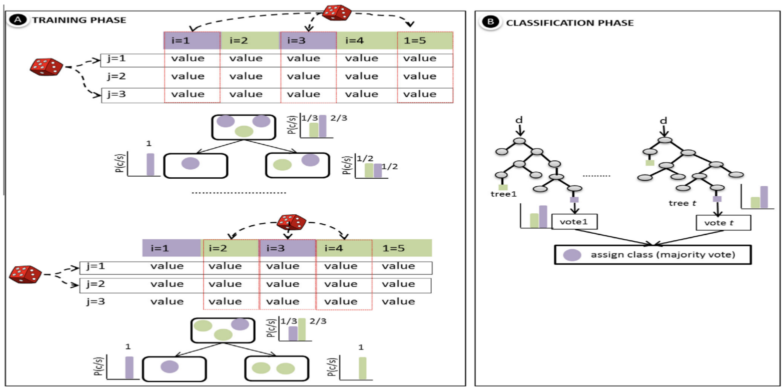

Fig. 1. Training and classification phases of Random Forest classifier: i = samples, j = variables, p = probability, c = class, s = data, t = number of trees, d = new data to be classified, and value = the different values that the variable j can have.

Figure 2: Implementation of Random Forest

XGBoost:

XGBoost, short for eXtreme Gradient Boosting, falls under the gradient boosting framework. At its core, XGBoost leverages the concept of gradient boosting, which involves creating an ensemble of weak prediction models, typically decision trees, in a sequential manner. Each subsequent model is trained to correct the mistakes made by the previous models, leading to a stronger and more accurate final model. One of the distinguishing features of XGBoost is its ability to handle regularization effectively. By adding a regularization term to the loss function, XGBoost penalizes complex models, resulting in better model performance on unseen data. XGBoost also employs an optimized gradient optimization algorithm that efficiently computes the gradients and updates the model parameters. This algorithmic optimization enables XGBoost to train models faster and scale well even with large datasets. Feature importance analysis is another valuable aspect of XGBoost. It provides insights into the relative importance of each feature, helping to identify the most influential predictors. Furthermore, XGBoost has built-in capabilities to handle missing values in the input data. It can automatically learn how to handle missing values, reducing the need for preprocessing. With its versatility and wide range of applications, XGBoost has achieved state-of-the-art performance in various domains. It is commonly used for classification, regression, ranking, and recommendation tasks, providing accurate predictions and valuable insights from complex datasets.

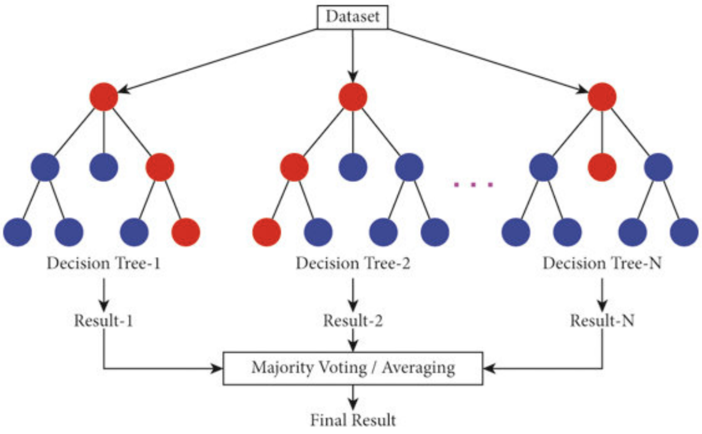

Figure 3: Implementation of XGBoost

2.4 LSTM

LSTM (Long Short-Term Memory) is a type of recurrent neural network (RNN) architecture that is widely used for sequence modeling tasks, particularly in natural language processing (NLP) and time series analysis.

The main advantage of LSTM over traditional RNNs is its ability to capture and retain long-term dependencies. Traditional RNNs suffer from the vanishing gradient problem. LSTM addresses this problem by introducing a memory cell and several gating mechanisms.

Memory Cell: The memory cell would store information over multiple time steps. It has a self-connected recurrent connection, allowing it to retain information over long sequences.

Input Gate: The input gate determines how much new information should be stored in the memory cell at each time step. It uses a sigmoid activation function to control the flow of information.

Forget Gate: The forget gate decides which information should be discarded from the memory cell. It uses a sigmoid activation function to selectively erase information that is no longer relevant.

Output Gate: The output gate regulates how much information from the memory cell should be used to produce the output at each time step. It uses a sigmoid activation function and a tanh activation function to control the flow of information.

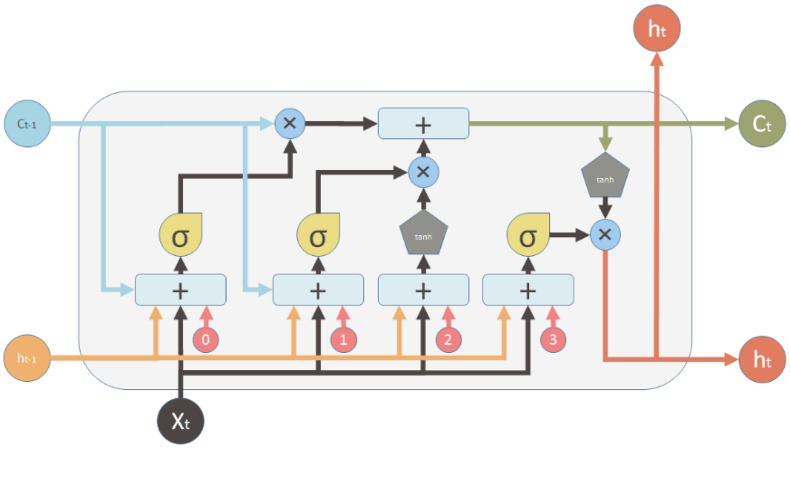

Figure 4: Implementation of LSTM

2.5 Attention Mechanism

The attention mechanism is a key concept in deep learning that allows models to focus on specific parts of input data when making predictions or generating output. It has been particularly influential in the field of natural language processing (NLP) and has found applications in various tasks such as machine translation, text summarization, and image captioning.
The attention mechanism helps models to assign different weights or importance to different parts of the input data, allowing them to pay attention to the most relevant information. Rather than relying on fixed-length feature vectors, attention enables models to dynamically weigh the importance of different elements of the input during the prediction process.

Input Representation: The attention mechanism begins with representing the input data, such as words in a sentence or pixels in an image, as a set of features or embeddings.

Query, Key, and Value: The attention mechanism defines three components: query, key, and value. These components can be viewed as transformations of the input data that help determine the attention weights.

Query: It represents the current state or context of the model, typically derived from the output of a previous layer or time step.

Key: It represents the input data and is used to compute the similarity or compatibility between the query and the input.

Value: It contains the actual information or content associated with the input data.

Attention Scores: The attention mechanism calculates a similarity metric between the query and each key to generate attention scores. These scores indicate the relevance or importance of each key with respect to the query.

Attention Weights: The attention scores are often normalized using a softmax function to obtain attention weights. These weights reflect the contribution of each input element to the final prediction or output.

Weighted Sum: The attention weights are used to compute a weighted sum of the corresponding values. This weighted sum represents the context or attended representation, which is then used for further processing, such as making predictions or generating output.

The attention mechanism allows models to selectively focus on different parts of the input, emphasizing the most relevant information while suppressing irrelevant or noisy data. This enables the model to capture dependencies, context, and long-range dependencies effectively, leading to improved performance in various tasks.
One popular variant of the attention mechanism is called "self-attention" or "scaled dot-product attention." In self-attention, the query, key, and value are derived from the same input, enabling the model to attend to different positions within the input sequence. This variant has been particularly successful in transformer models, which have achieved state-of-the-art results in various NLP tasks.

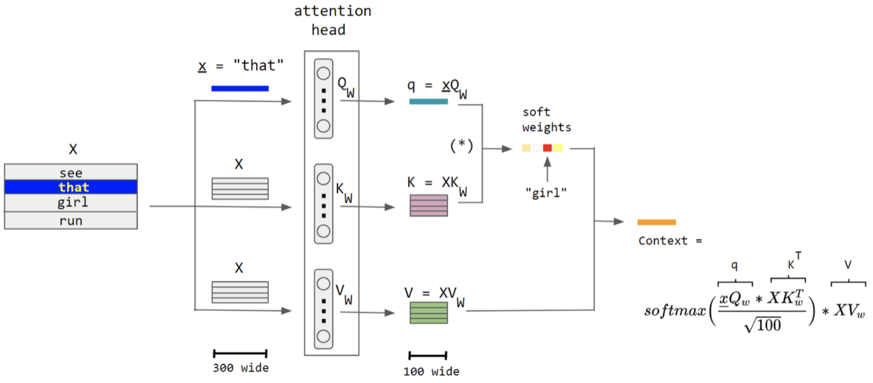

Figure 5: Implementation of Attention Mechanism

3 Data

The datasets used in this paper are the historical prices of the top 8 individual stocks and the indexes of the top 7 industries in the United States. This would help verify the trading strategy's applicability in different situations. The OHLCV of the daily stock prices are collected from Yahoo Finance, and the time range is from 2013.12.01 to 2023.12.01. This prolonged period would help to guarantee the generality of the industry and stocks involved.

3.1 Selection of Stocks

Industry indexes are calculated as the weighted average of the stocks within a specific industry. (for example, S&P500) As a result, these indexes serve as indicators of the overall market trend within that industry and tend to be more stable and predictable than individual stocks' prices. Conversely, the prices of individual stocks are generally more volatile. They exhibit larger standard deviations on average and are prone to experiencing sudden jumps or declines. This volatility presents both potential risks and opportunities for profit.

3.2 Data Preprocessing

It is important to consider the varying ranges of the input values. These differences in value ranges can lead to oscillations in gradients within the neural network, ultimately impacting the performance of the model. This can result in suboptimal actions and lower rewards. Data normalization can address this issue by standardizing the data through mean removal and scaling to unit variance. In this project, the min max standardization method is utilized for data normalization. This normalization process improves gradient flow, making network training easier, and enhances rewards by enabling optimal actions in each state.

4 Methodology

4.1 Marking the Optimal Buying and Selling Time

To address the limitations of the previous method, we adopt a new approach that involves identifying the optimal buying and selling points within the training set, guided by the principle of "buy low, sell high." Specifically, we designate the points where the price consistently increases over the next three days as the optimal buying points, while the points where the price consistently decreases over the next three days are marked as the optimal selling points. The optimal buying points are assigned a label of 1, indicating the use of all available funds to purchase stocks. Conversely, the optimal selling points are labeled 0, indicating the absence of any stocks in possession. Data points that do not qualify as optimal buying or selling points are determined based on the decisions made by preceding data points.

4.2 Extracting the Feature Vector

To extract the feature vector produced by the LSTM-AM model, we first feed the data into the LSTM-AM model trained for stock price prediction. After that, we would extract the feature vector produced after the attention mechanism layer for the classification task. The feature vector for the baseline model, however, would only be the closing stock price for the past 5 days.

4.3 Classification Task

The random forest classification model and the XGBoost are used to figure out the correlation between the feature vectors and the optimal trading decisions. The classification result on the testing set would be used to construct a trading strategy, and these strategies would be compared with the baseline “buy and hold”. Its result in trading would then be compared to the “buy and hold” strategy.

5 Result Analysis

5.1 Information Technology Sector

During a comprehensive study of historical stock prices, it became evident that many companies in the Information Technology sector experienced drawdowns at the beginning of the testing period. However, the effectiveness of the algorithm varied across different stocks.

Specifically, when analyzing Alphabet and NVDA, it was notable that the strategy based on the close price encountered challenges, as both stocks exhibited significantly different patterns in their training and testing sets.

Alphabet's stock demonstrated a steady increase until the beginning of 2021, after which it experienced an unprecedented drawdown. On the other hand, NVDA's price witnessed a significant drop before eventually rebounding to reach a historical high. These observations suggest that the LSTM algorithm successfully captured features within the stock prices, making the model more resilient to changes in the stock's pattern.

Furthermore, it is apparent that XGBoost outperformed random forest in classification tasks within this particular case. This implies that XGBoost provided a distinct advantage in accurately classifying stock patterns compared to random forest.

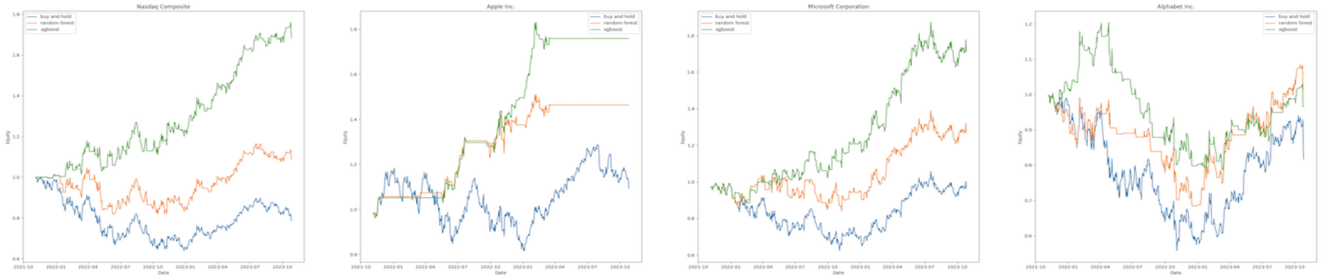

Figure 6: The equity curve of the first 4 Information Technology stocks with close price

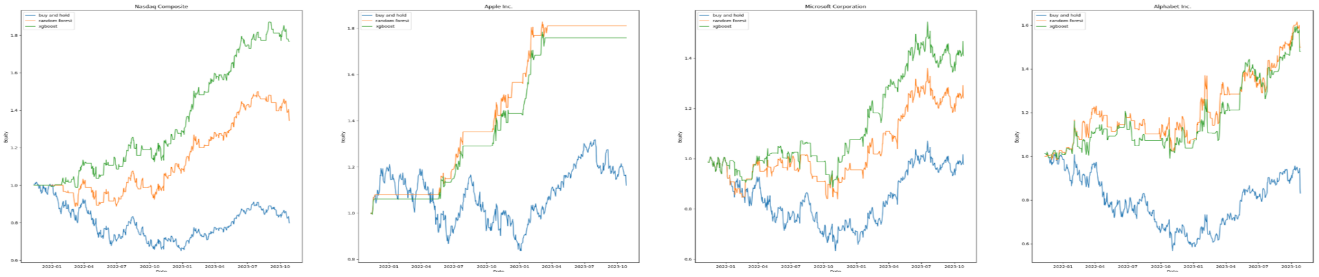

Figure 7: The equity curve of the first 4 Information Technology stocks with LSTM

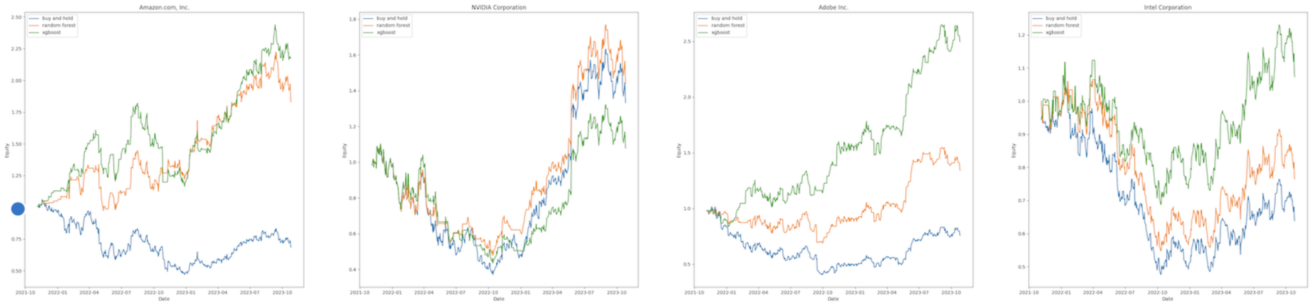

Figure 8: The equity curve of the last 4 Information Technology stocks with close price

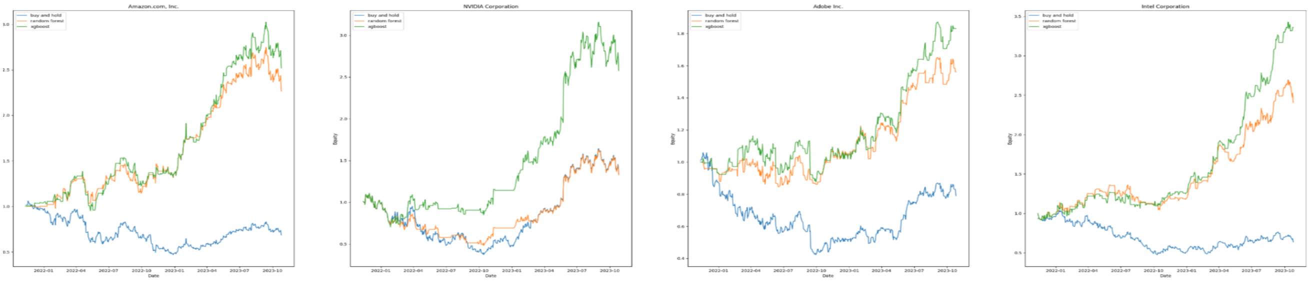

Figure 9: The equity curve of the last 4 Information Technology stocks with LSTM

	B&H	RF1	XGB1	RF2	XGB2
Average Sharpe Ratio	-0.02	0.66	1.21	1.31	1.69
Average Maximum Drawdown	48.14%	31.04%	21.68%	21.82%	19.92%
CAGR	-8%	13%	31%	27%	47%

Figure 10: The statistics for the IT sector

5.2 Health Care Sector

A simple observation reveals that the health sector tends to exhibit more abnormal patterns compared to the other seven sectors. Upon researching historical stock prices, it was discovered that Merck & Co and Eli Lilly And Co reached historic highs during the testing period, while Amgen experienced significantly larger fluctuations. This can be attributed to the high demand for developing new treatments during the COVID period, causing companies in the industry to behave differently from the training period.

However, similar to the Information Technology sector, the LSTM-XGB model outperforms the other three models. Additionally, when controlling for variables, both LSTM and XGB demonstrate superiority over their counterparts.

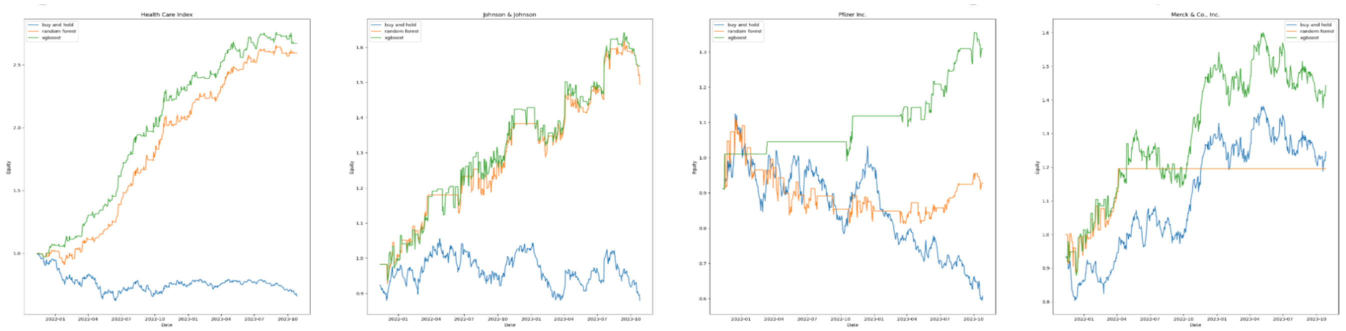

Figure 11: The equity curve of the first 4 Information Technology stocks with close price

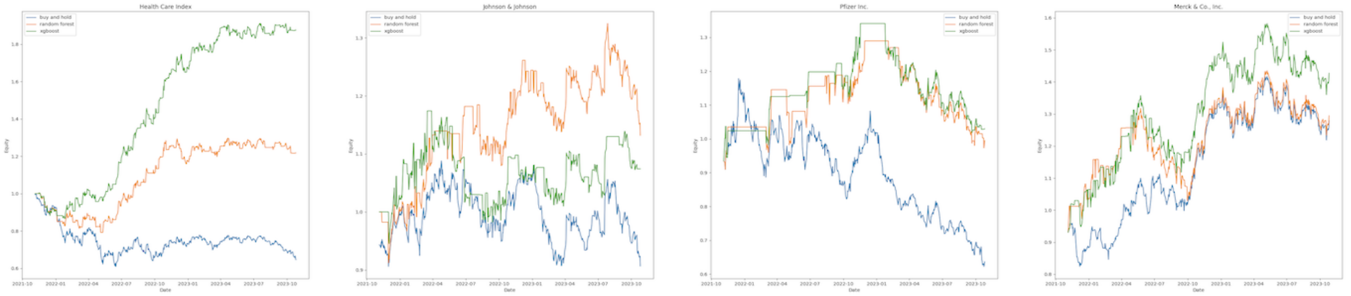

Figure 12: The equity curve of the first 4 Health Care stocks with LSTM

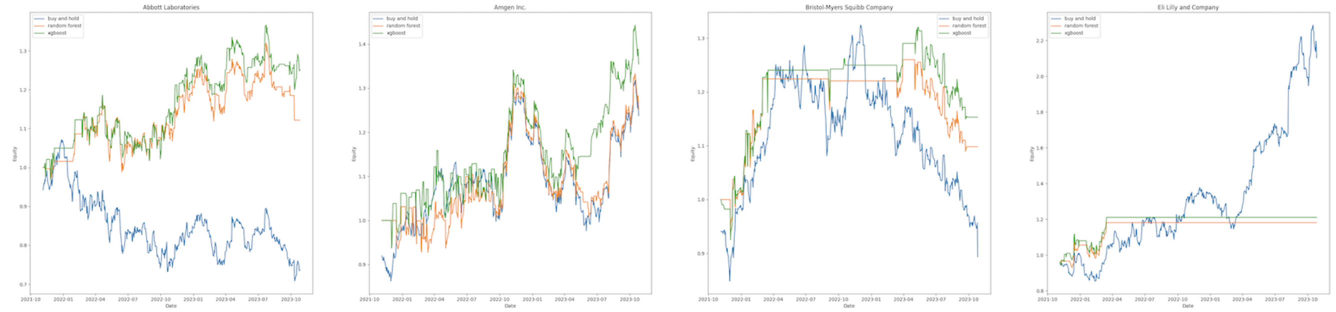

Figure 13: The equity curve of the last 4 Health Care stocks with close price

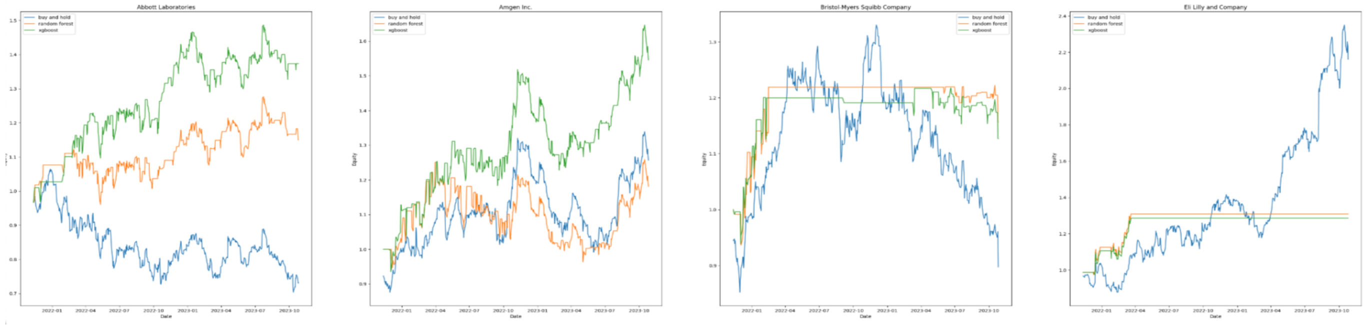

Figure 14: The equity curve of the last 4 Health Care stocks with LSTM

	B&H	RF1	XGB1	RF2	XGB2
Average Sharpe Ratio	0.20	0.66	0.96	1.13	1.51
Average Maximum Drawdown	28.12 %	17.66 %	14.74%	13.39%	9.51%
CAGR	-6%	21%	38%	16%	43%

Figure 15: The statistics for the health care sector

5.3 Consumer Discretionary Sector

When compared to the healthcare sector, which COVID-19 significantly impacted, both strategies demonstrate greater success in the consumer discretionary sector. Additionally, the close price-based model is not as effective as the LSTM-based models in extracting potential features from stock prices. Upon careful examination of the price patterns for each stock, it was discovered that the two underperforming stocks exhibited completely different patterns in the training and testing sets. Tesla displayed larger fluctuations with a higher standard deviation in the testing set, while Home Depot reached its historical high point at the beginning of 2022 before experiencing a decline.

Once again, LSTM showcases its capability to extract meaningful features from sequential data. The LSTM-XGB model is still the best model among the four.

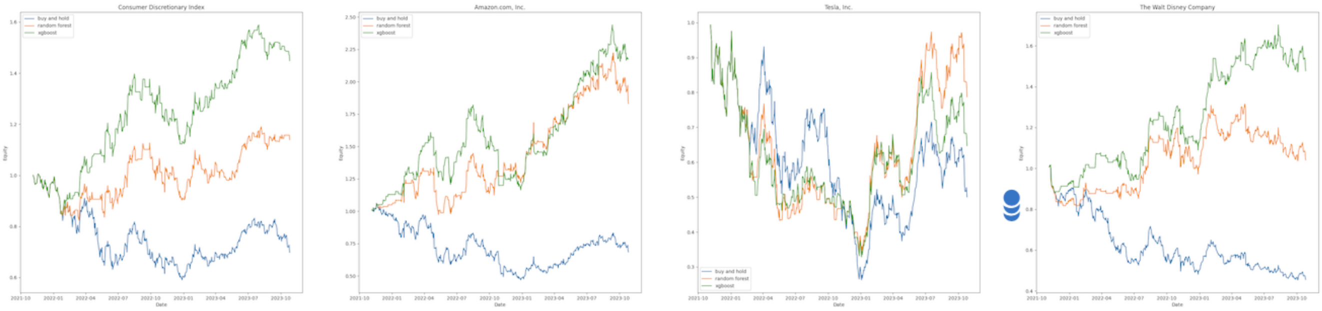

Figure 16: The equity curve of the first 4 Consumer Discretionary stocks with close price

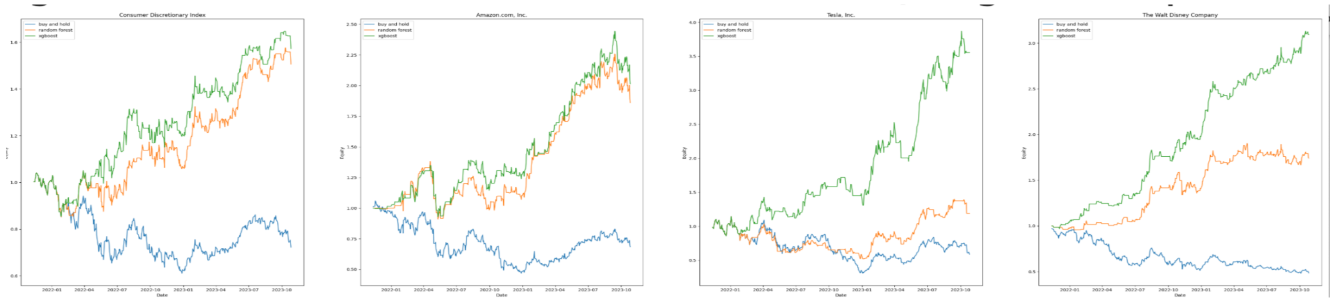

Figure 17: The equity curve of the first 4 Consumer Discretionary stocks with LSTM

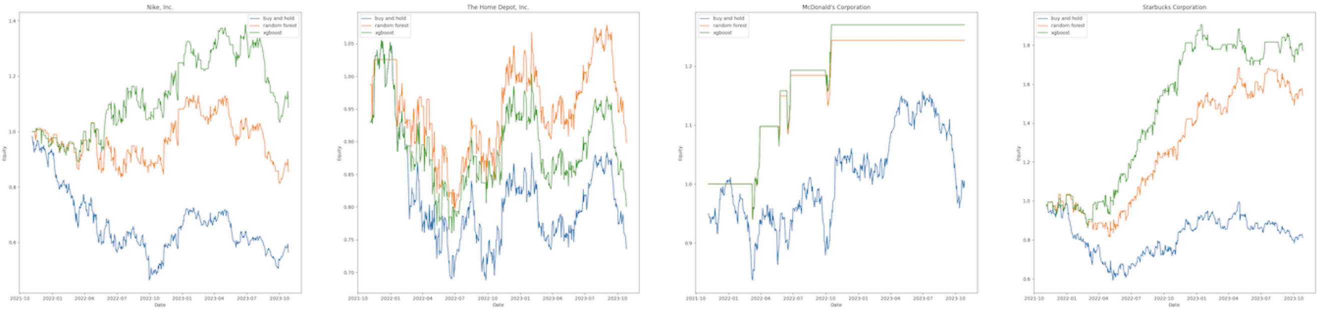

Figure 18: The equity curve of the last 4 Consumer Discretionary stocks with close price

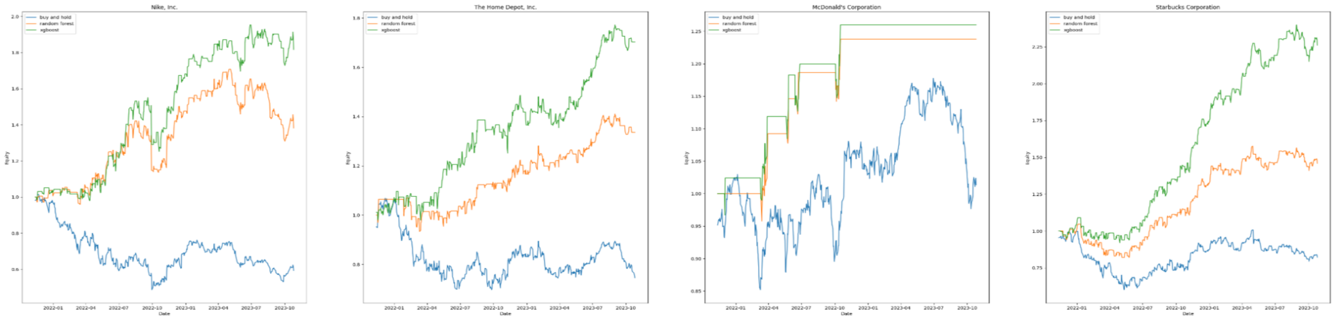

Figure 19: The equity curve of the last 4 Consumer Discretionary stocks with LSTM

	B&H	RF1	XGB1	RF2	XGB2
Average Sharpe Ratio	-0.43	0.53	0.86	1.09	1.80
Average Maximum Drawdown	46.32%	26.56%	26.37%	23.07%	16.33%
CAGR	14%	7%	19%	36%	129%

Figure 20: The statistics for the Consumer Discretionary sector

5.4 Financials Sector

During the COVID-19 pandemic, the financial sectors exhibited greater stability, which led both strategies to partially avoid significant drawdowns in stocks. However, the LSTM-based strategies achieved more consistent profits and were able to mitigate losses more effectively. The statistics presented in the chart strongly support the superiority of the LSTM-XGB-based model.

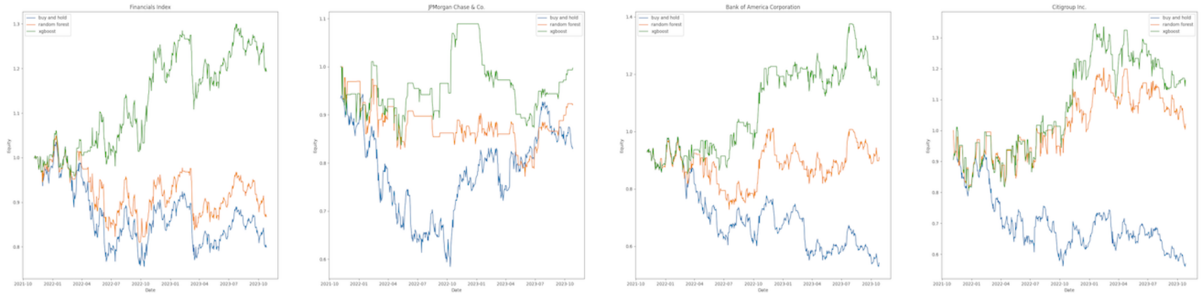

Figure 21: The equity curve of the first 4 Financials stocks with close price

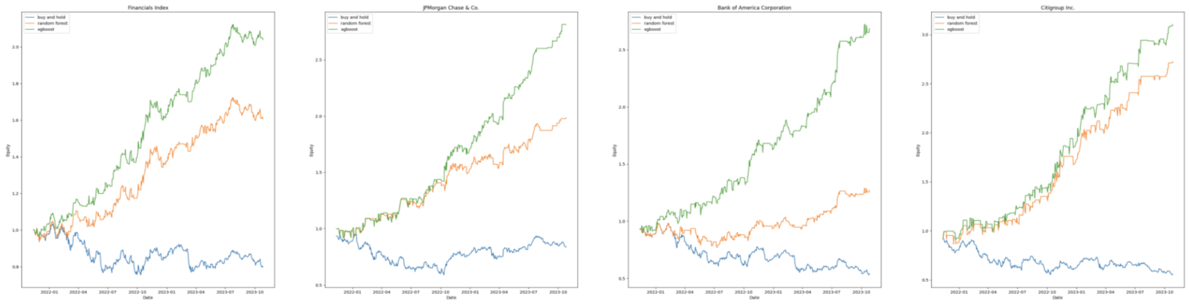

Figure 22: The equity curve of the first 4 Financials stocks with LSTM

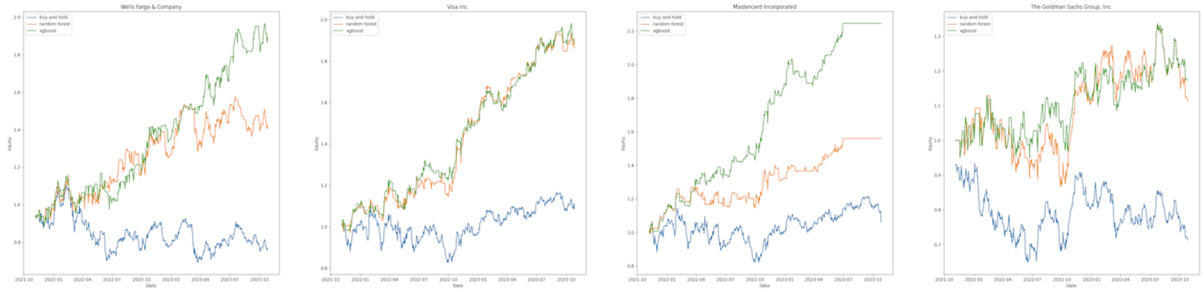

Figure 23: The equity curve of the last 4 Financials stocks with close price

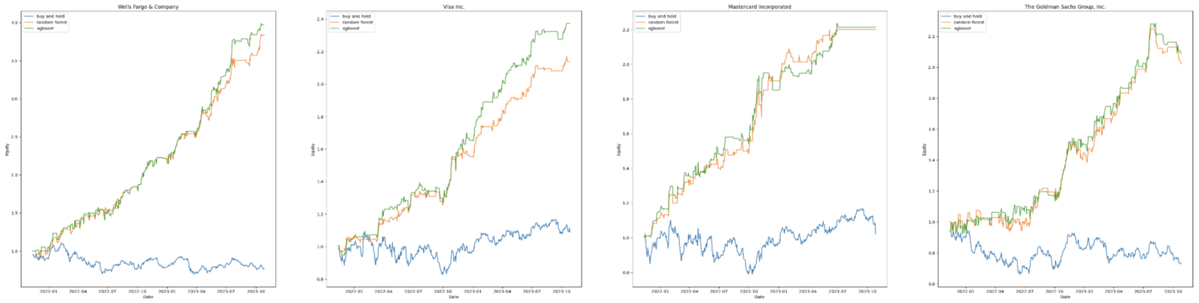

Figure 24: The equity curve of the last 4 Financials stocks with LSTM

	B&H	RF1	XGB1	RF2	XGB2
Average Sharpe Ratio	-0.29	0.61	1.14	2.19	2.64
Average Maximum Drawdown	33.91%	18.11%	15.25%	11.07%	9.45%
CAGR	-16%	7%	18%	23%	58%

Figure 25: The statistics for the Financials sector

5.5 Materials Sector

The materials sector exhibits a similar pattern to the financial industry, and the LSTM and XGB-based models outperform their counterparts in this sector as well.

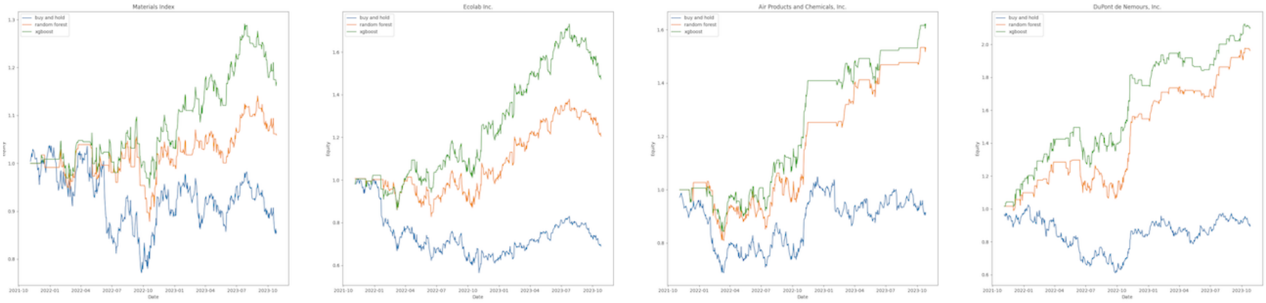

Figure 26: The equity curve of the first 4 Materials stocks with close price

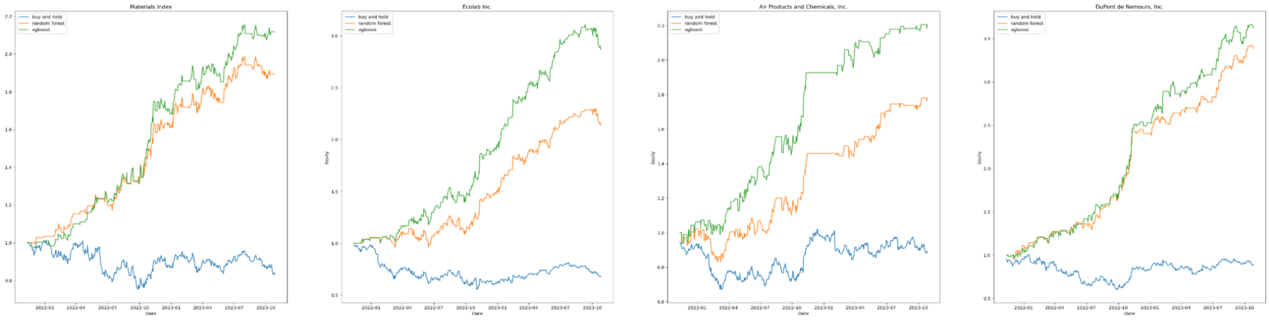

Figure 27: The equity curve of the first 4 Materials stocks with LSTM

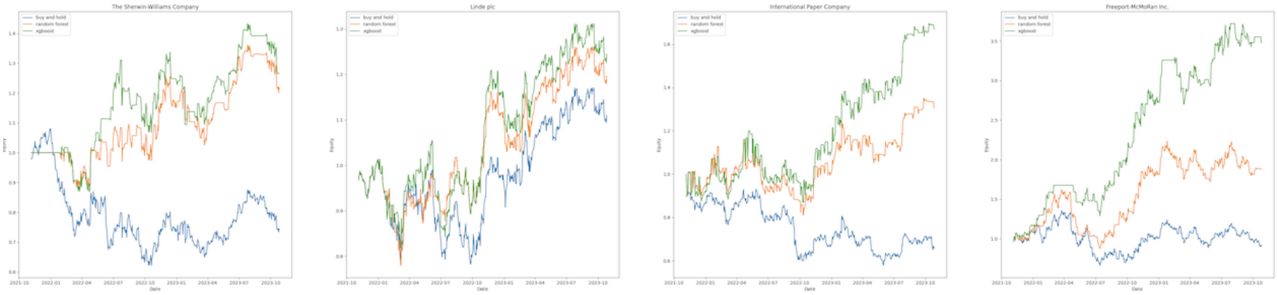

Figure 28: The equity curve of the last 4 Materials stocks with close price

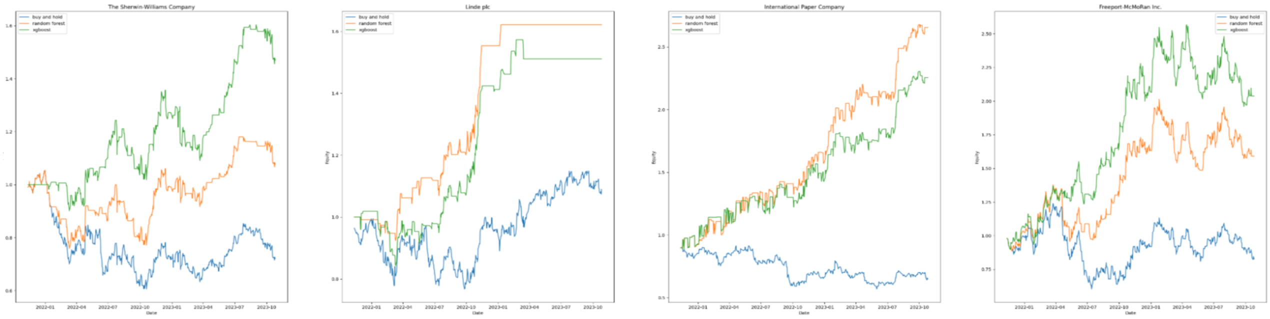

Figure 29: The equity curve of the last 4 Materials stocks with LSTM

	B&H	RF1	XGB1	RF2	XGB2
Average Sharpe Ratio	-0.13	0.89	1.21	1.86	2.11
Average Maximum Drawdown	36.81%	23.93%	18.37%	15.65%	15.21%
CAGR	-9%	21%	36%	41%	65%

Figure 30: The statistics for the materials sector

5.6 Industrials Sector

The potential of LSTM-based models becomes evident when applied to trading stocks from companies like Honeywell International Inc. and 3M Company. While the Close price-based models struggle to generate substantial profits in these cases, the LSTM-based models successfully avoid significant losses by intelligently responding to plummeting stock prices. Consequently, the LSTM-XGB strategy remains the most effective approach.

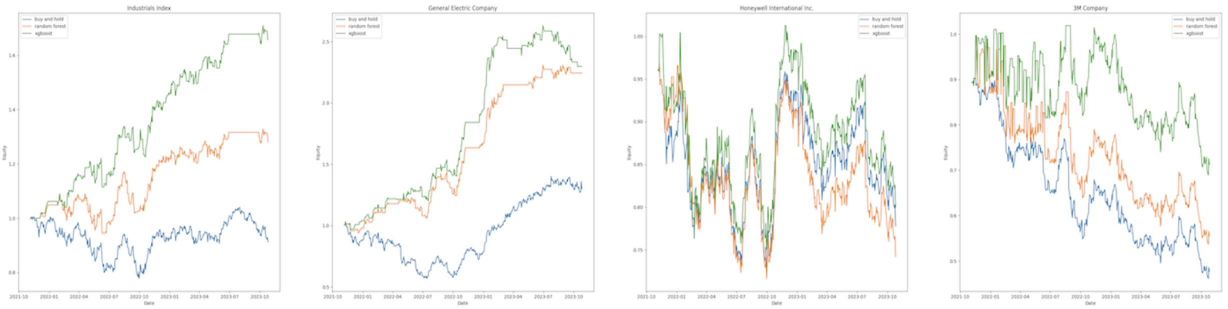

Figure 31: The equity curve of the first 4 Industrials stocks with close price

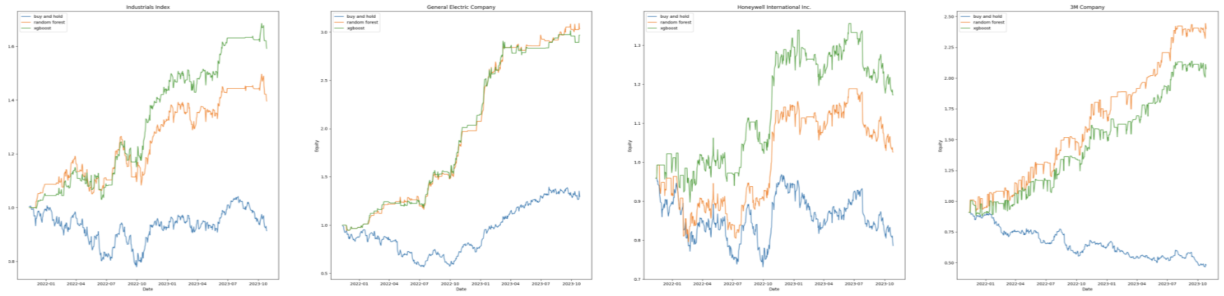

Figure 32: The equity curve of the first 4 Industrials stocks with LSTM

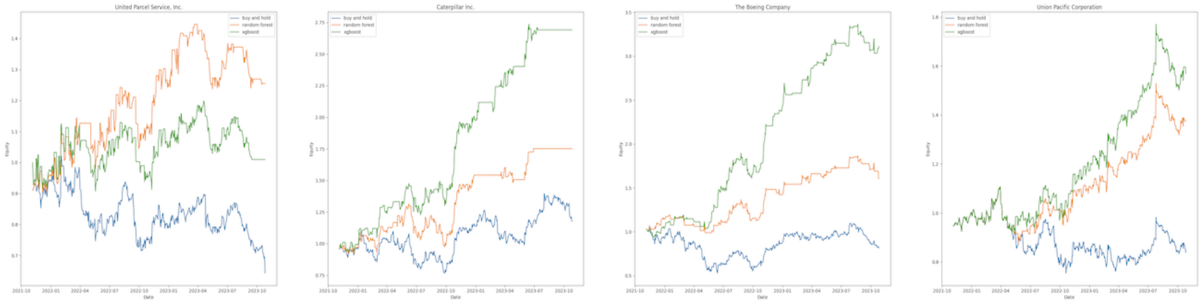

Figure 33: The equity curve of the last 4 Industrials stocks with close price

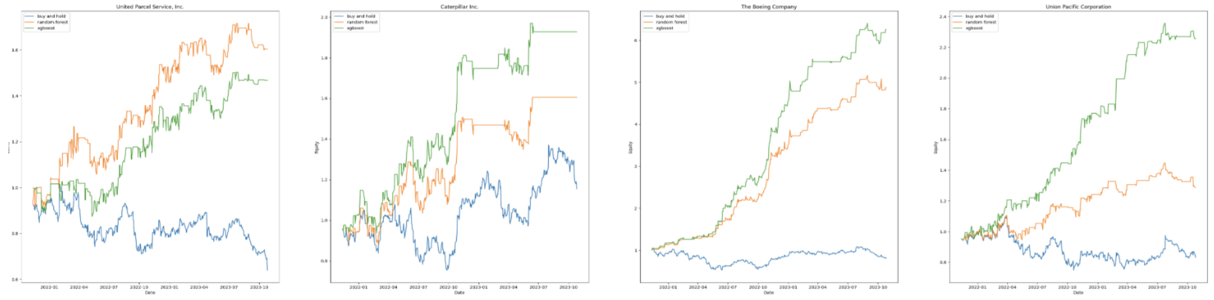

Figure 34: The equity curve of the last 4 Industrials stocks with LSTM

	B&H	RF1	XGB1	RF2	XGB2
Average Sharpe Ratio	-0.13	0.89	1.25	1.87	2.11
Average Maximum Drawdown	36.69%	23.93%	18.37%	15.65%	13.21%
CAGR	-9%	15%	40%	54%	77%

Figure 35: The statistics for the industrials sector

5.7 Utilities Sector

Similar to the industrials sector, the superiority of LSTM-based models over close price-only models is once again evident in the cases of Duke Energy Corporation and Sempra Energy. However, it is worth noting that all four models demonstrate limited effectiveness when applied to the Exelon Corporation, which underscores the existing limitations and highlights the potential for future improvements in these models.

The LSTM-XGB model is regarded as the best-performing model in this sector.

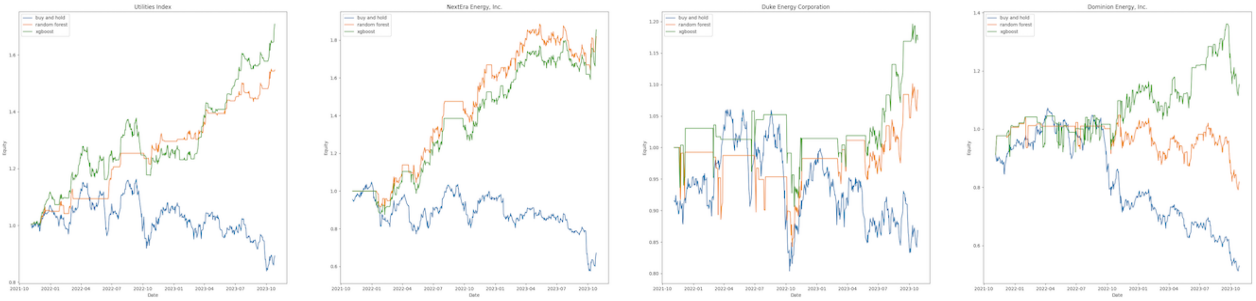

Figure 36: The equity curve of the first 4 Utilities stocks with close price

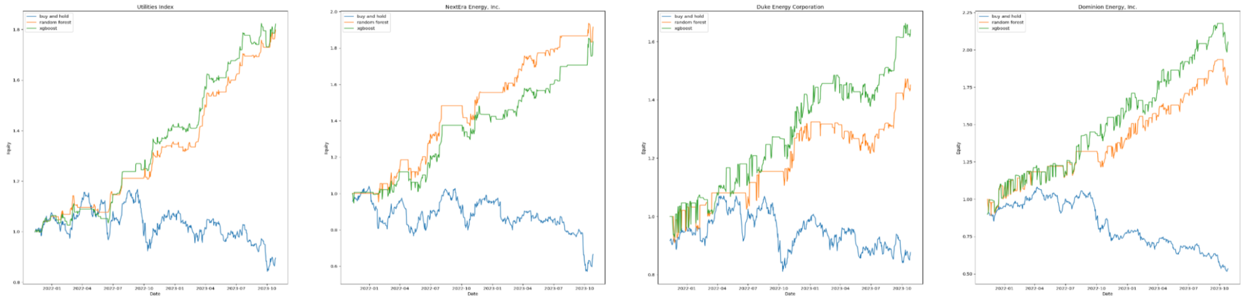

Figure 37: The equity curve of the first 4 Utilities stocks with LSTM

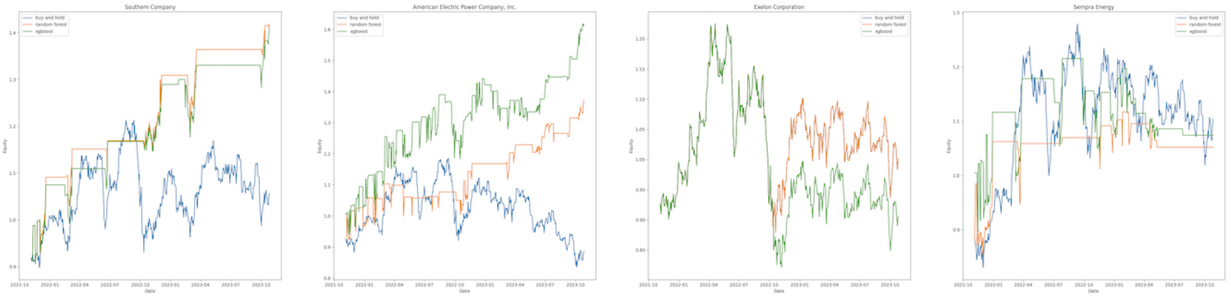

Figure 38: The equity curve of the last 4 Utilities stocks with close price

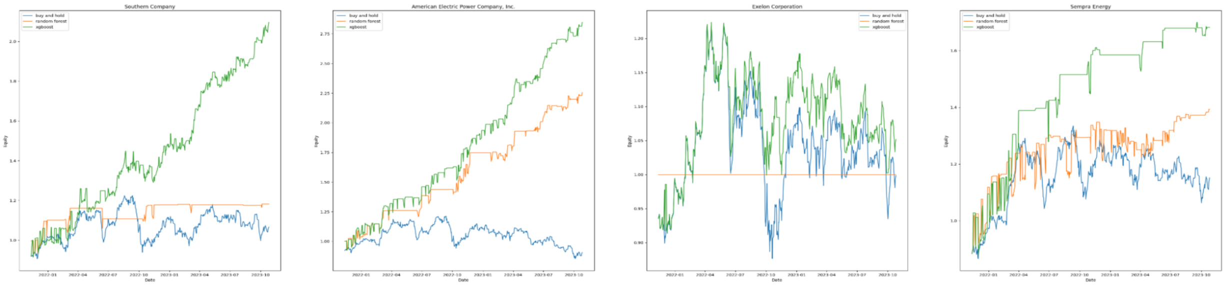

Figure 39: The equity curve of the last 4 Utilities stocks with LSTM

	B&H	RF1	XGB1	RF2	XGB2
Average Sharpe Ratio	-0.06	0.99	1.01	1.62	1.64
Average Maximum Drawdown	36.40%	22.63%	19.05%	14.09%	11.68%
CAGR	-8%	13%	24%	29%	36%

Figure 40: The statistics for the Utilities sector

5.8 Real Estate Sector

The LSTM-XGB model outperforms the other 3 in this sector.

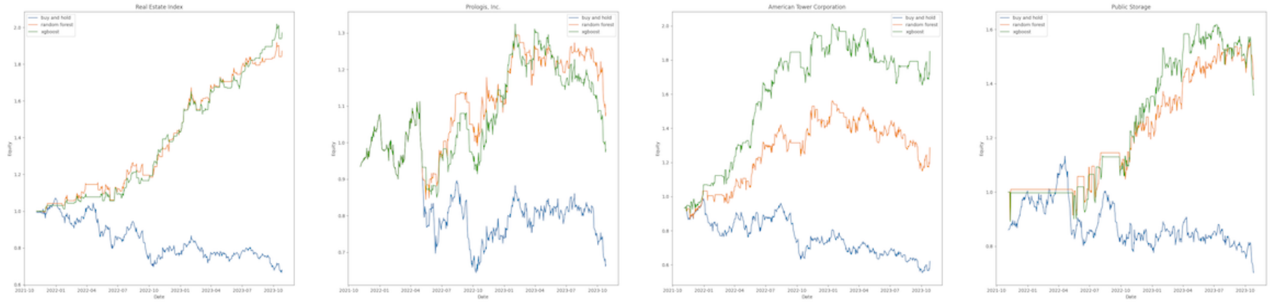

Figure 41: The equity curve of the first 4 Real Estate stocks with close price

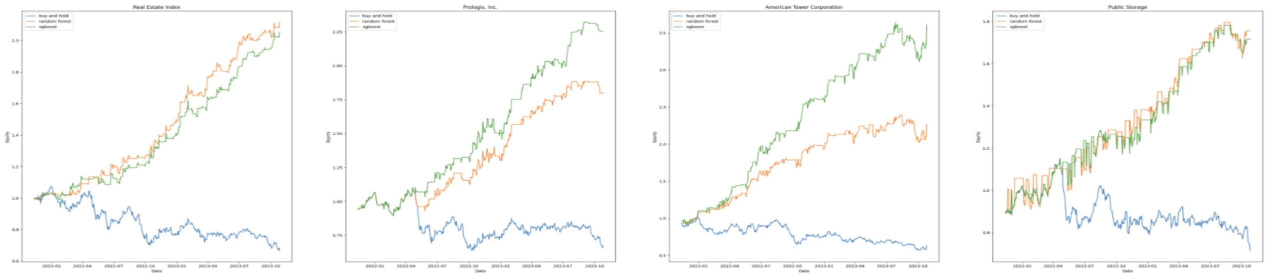

Figure 42: The equity curve of the first 4 Real Estate stocks with LSTM

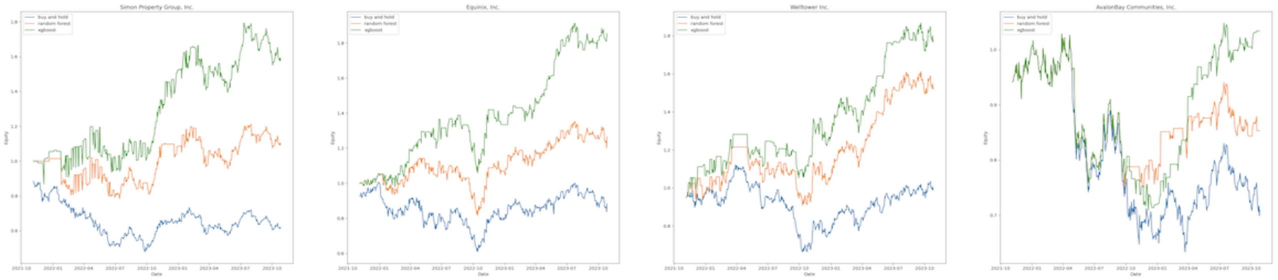

Figure 43: The equity curve of the last 4 Real Estate stocks with close price

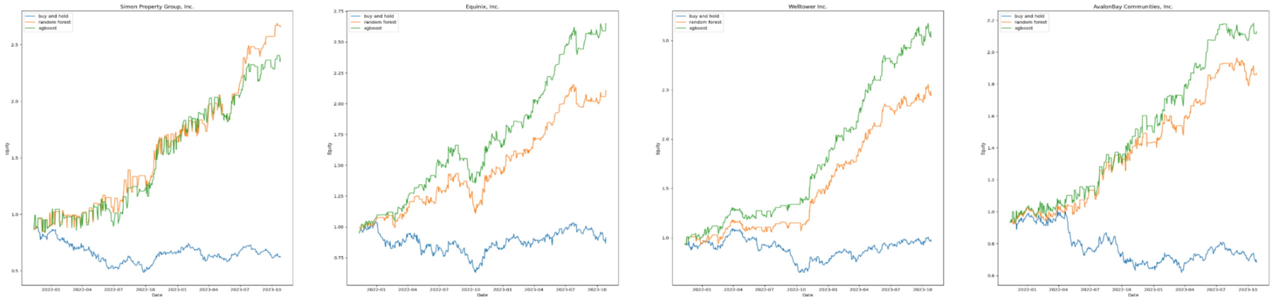

Figure 44: The equity curve of the last 4 Real Estate stocks with LSTM

	B&H	RF1	XGB1	RF2	XGB2
Average Sharpe Ratio	-0.38	0.84	1.22	2.03	2.39
Average Maximum Drawdown	40.81%	21.98%	20.41%	13.55%	13.27%
CAGR	-18%	16%	29%	52%	69%

Figure 45: The statistics for the Real Estate sector

6 Conclusion

1. Across all eight sectors, the LSTM-XGB-based model consistently outperforms other models based on metrics such as average maximum drawdown, Sharpe ratio, and equity curve analysis. The success of this model can be attributed to the LSTM's ability to extract relevant features from stock data, which contributes to its superior performance.
2. However, when using a close price-based model, it is observed that the model begins to malfunction when the training data undergoes significant changes. This suggests that the close price-based model has limitations in certain circumstances.
3. In the case of the healthcare sector, even the best-performing model fails to generate substantial profits in nearly half of the stocks. This can be attributed to the strong correlation between the healthcare sector and COVID-19. It is crucial to address this issue and explore ways to upgrade the models so that they can perform effectively in more adverse environments.

7 Future Aspects to Explore

1. Previous research papers have consistently shown that transformer models exhibit higher efficiency and effectiveness in training and predicting long-term trends compared to LSTM models. These findings strongly support the superiority of transformer models in extracting long-term patterns. Furthermore, subsequent studies have provided additional evidence that transformer models can independently extract patterns from data without requiring supplementary techniques or models.

2. The paper establishes a clear differentiation between the training and testing periods. However, a limitation of this approach is the occasional suboptimal performance of the model due to discrepancies between the training and testing sets. To overcome this limitation, future research could explore the continuous integration of testing data into the training data. This approach would enable the model to adapt and learn from the evolving characteristics of the testing data, potentially enhancing its overall performance and robustness.

3. The underperformance of models in the Health Care sector emphasizes the importance of incorporating macroeconomic factors into trading strategies to mitigate potential losses during market fluctuations. To tackle this challenge, one approach is to integrate fundamental analysis as a supplementary method to the existing framework. By considering fundamental factors such as industry trends, financial statements, and market conditions, we can enhance our understanding of the Health Care sector and make more informed trading decisions. This holistic approach, which combines the original method with fundamental analysis, has the potential to improve performance and better navigate the dynamic nature of the market.

8 References

1. *Sang C, Di Pierro M, Improving Trading Technical Analysis with TensorFlow Long Short-Term Memory (LSTM) Neural Network, The Journal of Finance and Data Science, https:// doi.org/10.1016/j.jfds.2018.10.003.*
2. *Ding, Q., Wu, S., Sun, H., Guo, J., & Guo, J. (2020, July). Hierarchical Multi-Scale Gaussian Transformer for Stock Movement Prediction. In IJCAI*
3. *Sang, Chenjie & Di Pierro, Massimo. (2018). Improving Trading Technical Analysis with TensorFlow Long Short-Term Memory (LSTM) Neural Network. The Journal of Finance and Data Science. 5. 10.1016/j.jfds.2018.10.003.*
4. *arXiv:2305.14368, Support for Stock Trend Prediction Using Transformers and Sentiment Analysis*
5. *Kim, R., So, C.H., Jeong, M., Lee, S., Kim, J., & Kang, J. (2019). HATS: A Hierarchical Graph Attention Network for Stock Movement Prediction. ArXiv, abs/1908.07999.*
6. *Fong, Simon & Tai, Jackie & Si, Yain Whar. (2011). Trend Following Algorithms for Technical Trading in Stock Market. Journal of Emerging Technologies in Web Intelligence (JETWI). 3. 136-145. 10.4304/jetwi.3.2.136-145.*