Enhancing Dynamic Portfolio Allocation in Finance: Leveraging Deep Reinforcement Learning with Risk-Aware Optimization

Sabuni Emmanuel
Sitali Gabriel Sitali

Author Affiliation(s)
Sabuni0101@hotmail.com
sitaligabrielsitali@gmail.com

Abstract

This study explores the creation of a customized deep reinforcement learning (DRL) model for the finance sector's dynamic portfolio allocation. Conventional approaches frequently find it difficult to adequately manage risk and adjust to changes in the market. The goal of this study is to incorporate risk awareness into the optimization process in order to overcome these issues. The main goals are to create a new reward system, investigate sophisticated DRL approaches, apply risk control strategies, carry out a lot of backtesting, analyze sensitivity, and produce insights that investors and portfolio managers can use. This study aims to expand the use of DRL in risk-aware portfolio allocation by investigating cutting-edge strategies and unique method combinations.

Keywords: *deep reinforcement learning (DRL), backtesting, dynamic portfolio allocation, HMM, risk mitigation*

Introduction:

The finance sector has been at the forefront of implementing cutting-edge technologies to enhance decision-making processes. Deep reinforcement learning (DRL) approaches have gained significant attention due to their ability to adapt to complex, non-linear environments, particularly in dynamic portfolio allocation. This literature review examines the research landscape on integrating DRL techniques in finance, focusing on how risk awareness is incorporated into the optimization process.

Problem statement

This project aims to develop a specialized deep reinforcement learning (DRL) model for dynamic portfolio allocation within the finance sector. Traditional methods often struggle to adapt to changing market conditions and effectively manage risk. By incorporating risk awareness into the optimization process, this project seeks to overcome these limitations. Historical data will be utilized for model training and backtesting, with monthly portfolio rebalancing. Key objectives include creating a unique reward function, exploring advanced DRL methodologies, integrating risk management techniques, conducting thorough backtesting, performing sensitivity analysis, and deriving actionable insights for investors and portfolio managers. The project aims to make

ISSN: XXXX-XXXX IJXX

distinctive contributions by exploring innovative approaches and novel combinations of methodologies, ultimately advancing the application of DRL in risk-aware portfolio allocation.

Goals and Objectives:

1. Develop a deep reinforcement learning (DRL) system tailored for dynamic portfolio allocation, allowing for adaptive asset allocations over time.
2. Design and implement a novel reward function integrating risk-adjusted performance metric like Sharpe ratio, aiming to optimize the balance between risk management and maximizing returns.
3. Integrate risk management techniques directly into the DRL framework, including constraints on maximum drawdown, portfolio volatility, and other relevant risk measurements, to ensure robust and resilient portfolio management strategies.

Background

Optimizing one's portfolio is a crucial part of trading systems. The goal of optimization is to choose the optimal allocation of assets within a portfolio to maximize returns at a specific risk level. This theory, also referred to as modern portfolio theory (MPT), was developed by Markowitz in his early work. Building a portfolio of this type has several advantages and among them being the encouragement of diversification which evens out the equity curve and yields a higher return per risk than trading individual assets.

Theoretical Framework

Literature Review:

Deep Reinforcement Learning in Finance:

Deep reinforcement learning (DRL) combines reinforcement learning with deep learning, demonstrating remarkable performance in resolving intricate sequential decision-making scenarios. In finance, DRL has been utilized for asset pricing, portfolio management, and algorithmic trading.

Dynamic Portfolio Allocation:

Dynamic portfolio allocation aims to optimize returns and minimize risk by adjusting investment portfolios gradually. Conventional approaches often rely on heuristics or static models, which may not be flexible enough to respond to changing market conditions. DRL presents a viable method for dynamic portfolio allocation by leveraging historical data and current market information to identify optimal trading strategies.

Incorporating Risk Awareness:

Risk management is essential for investors to achieve their financial objectives while mitigating negative risk impacts. Integrating risk awareness into the optimization process ensures portfolio

ISSN: XXXX-XXXX IJXX

resilience against unfavorable market conditions. DRL approaches can be enhanced to incorporate risk measurements such as volatility into the reward function.

Recent research has explored various DRL strategies for dynamic portfolio allocation, with a focus on risk awareness. [7] Proposed a DRL framework for dynamic portfolio optimization that directly optimizes the Sharpe ratio, effectively balancing risk and return objectives. [11] Introduced a risk-sensitive reinforcement learning approach using deep deterministic policy gradient (DDPG) for dynamic portfolio allocation. Additionally, hybrid approaches combining DRL with other optimization techniques have been investigated, such as mean-variance optimization combined with DRL by [5].

New directions in risk-aware portfolio allocation have emerged with advancements in DRL algorithms, including distributional reinforcement learning (DRL). [21] Presented a distributional reinforcement learning method for dynamic portfolio allocation, directly optimizing risk-adjusted returns distributions for more robust risk awareness.

Competitor Analysis:

Competitors in the deployment of DRL techniques for dynamic portfolio allocation have made significant strides, but there are still opportunities for innovation. Traditional portfolio optimization techniques and heuristic-based approaches have limitations in adapting to changing market conditions and incorporating risk management. Existing DRL-based models for portfolio allocation lack explicit risk management integration, leaving room for further development.

Business Case:

The finance industry's increasing digitalization necessitates advanced tools for adaptive portfolio management and risk mitigation. Deep reinforcement learning offers a promising solution by leveraging machine learning technologies to optimize portfolio allocation policies based on data. By incorporating risk awareness into portfolio allocation, the proposed DRL-based model aims to address current market challenges and provide a competitive value proposition.

SWOT Analysis of Competitors:

Competitors' strengths lie in their optimized methodologies supported by strong theoretical foundations. However, weaknesses include a lack of adaptability to market dynamics and absence of risk management, presenting opportunities for innovation in DRL-based portfolio allocation. Threats include competition from traditional financial organizations and potential legal difficulties, but advancements in DRL-based systems mitigate these risks.

Research design & methodology

In this section, we will introduce our framework and discuss how Sharpe ratio can be optimized through gradient ascent.

Objective Function

The Sharpe ratio, which is defined as expected return over volatility (risk-free rate excluded for simplicity), was used to calculate the return per risk of a portfolio.

$$L = \frac{E(R_p)}{Std(R_p)} \tag{1}$$

Where E(Rp) and Std(Rp) are the estimates of the mean and standard deviation of portfolio returns. Specifically, for a trading period of t = {1, · · · , T}, we could maximize the following objective function:

$$L_t = \frac{E(R_{p,t})}{\sqrt{E\left(R_{p,t}^2\right) - (E(R_{p,t}))^2}} \tag{2}$$

$$E\left(R_{p,t}\right) = \frac{1}{T}\sum_{t=1}^{T} R_{p,t} \tag{3}$$

where Rp,t is realized portfolio return over n assets at time t denoted as:

$$R_{p,t} = \sum_{i=1}^{n} w_{i,t-1} * r_{i,t} \tag{4}$$

where $r_{i,t}$ is the return of asset i with $r_{i,t} = (p_{i,t}/p_{i,t}-1 - 1)$. We represent the allocation ratio (position) of asset i as $w_{i,t} \in [0, 1]$ and $\sum_i^n w_{i,t} = 1$. In our approach, a neural network f with parameters θ is adopted to model $w_{i,t}$ for a long-only portfolio:

$$w_{i,t} = f(\theta | x_t) \tag{5}$$

Where xt represents the current market information and we bypass the classical forecasting step by linking the inputs with positions to maximize the Sharpe over trading period T, namely LT. However, a long-only portfolio imposes constraints that require weights to be positive and summed to one, we use softmax outputs to fulfill these requirements:

$$w_{i,t} = \frac{\exp(\overline{w}_{i,t})}{\sum_j^n \exp(\overline{w}_{j,t})} \tag{6}$$

Where $\overline{w}_{i,t}$ are the raw weights

Unconstrained optimization techniques can be used to optimize such a framework. In particular, to maximize the Sharpe ratio, use gradient ascent. An excellent derivation of the gradient of LT with regard to parameters θ can be found in [4, 9]. After obtaining $\partial L_T / \partial \theta$, we can use gradient ascent to update the parameters and repeatedly compute this value using training data:

ISSN: XXXX-XXXX IJXX

$$\theta_{\text{new}} = \theta_{\text{old}} + \alpha \frac{\partial LT}{\partial \theta} \tag{7}$$

Where α is the learning rate and the process can be repeated for many epochs until the convergence of Sharpe ratio or the optimization of validation performance is achieved.

Efficient Frontier

An efficient frontier is a graph where the y-axis represents the expected return and the x-axis represents the portfolio's risk, which is determined by the standard deviation of the portfolio. It aids in the visualization of efficient, global minimal variance, and inefficient portfolios as well as the risk-return trade-off of a portfolio.

This is the tool we used to make more informed investing decisions because of its ability to create portfolios with higher expected returns and lower risks attached to them.

The efficient frontier is calculated as below:

$$E(Rp) = w_1E(R_1) + w_2E(R_2) \tag{8}$$

Where w_1 and w_2 are the weights of each asset.

Graphically it can be represented as follows:

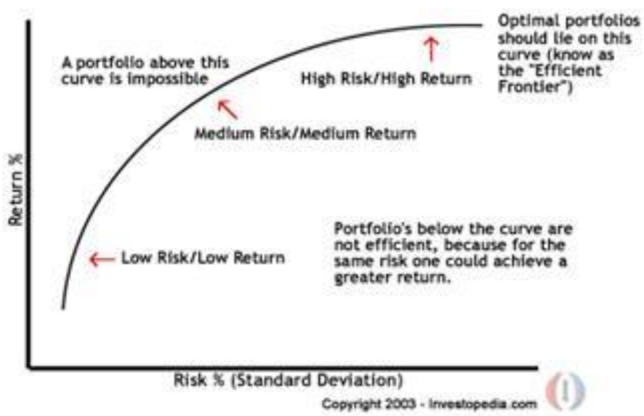

Fig.1 efficient frontier graph

Hidden Markov Models

The Hidden Markov Model (HMM) approach will be applied to the stock prediction problem and the reason for using this approach is fairly intuitive. HMM's have been successful in analyzing and predicting time depending phenomena, or time series. Because of this inherent relationship between time and the stock market prediction problem we feel it can be also applicable to the stock market.

Deep Reinforcement Learning Models (DRL)

ISSN: XXXX-XXXX IJXX

Our DRL model performed asset allocation based on PPO RL algorithms.

Deep Neural Networks Framework

A Deep Neural Network is proposed as the agent which finds an optimal policy that acts on the state of the environment to produce actions that maximize the reward function. The DRL process is modeled as follows

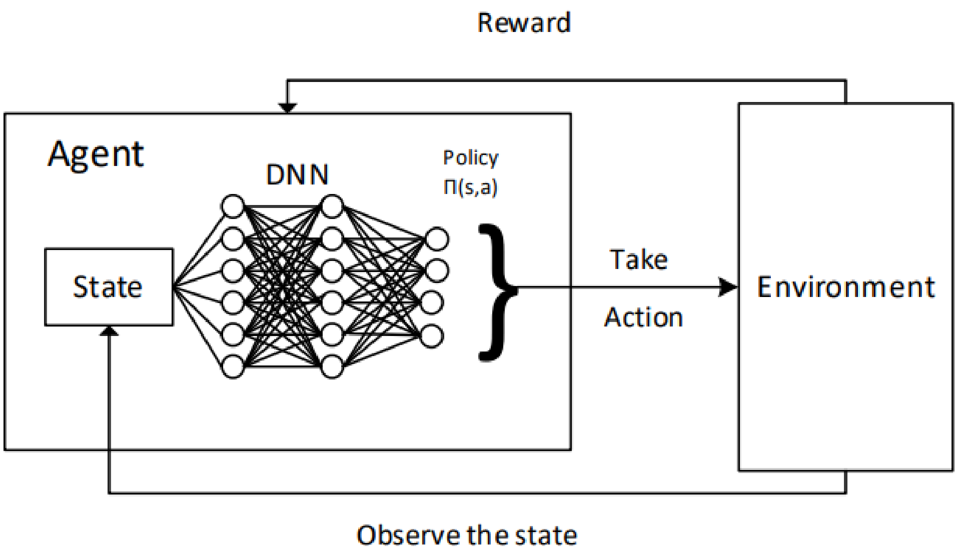

Figure 1: Basic Representation of a DRL Framework

Data Preparation:

Historical financial data was gathered including stock prices, economic indicators, and any other relevant factors, the data was preprocessed by handling missing values, normalizing or standardizing features, and splitting it into training, validation, and test sets.

Model Architecture:

We defined the Reinforcement Learning Configuration of our model as follows:

1. The state (S) includes the High, Low, Close, the covariance matrix of close prices and the identified market financial indicators that are used as inputs;

2. The action (A) is the desired weights allocation. The action at time t will be represented by the weights vectors at time $t-1$: $A_{t-1} = W_{t-1}$. As a result of the action at A_{t-1} and the state inputs, we will get an action or weights allocation at time t which is $A_t = W_t$. Taking into consideration the

ISSN: XXXX-XXXX IJXX

transaction costs, our model should make a decision on how the weights are rebalanced from W_{t-1} weights W_t. This is done to maximize the accumulative portfolio value.

3. The reward function is given based on the total portfolio value adjusted for the transaction costs. The reward function adjusted for transaction costs is given by:

$$(S_{t-1}, A_{t-1}) = \ln(A_{t-1} \cdot Y_{t-1} - \mu \textstyle\sum_{i=1}^{n} \left| A_{i,t-1} - W_{i,t-1} \right| \tag{9}$$

The reward function is fed back to the agent to reinforce the policy of making actions so that future actions are optimized to reach the objective function.

4. Policy, $(\boldsymbol{s}, \boldsymbol{a})$. The policy determines the action to take that maximizes the reward at each state. In our case the policy is made by a Deep Neural network using a Reinforcement Learning algorithm.

The Deep Neural Network was trained using the training data and defined a loss function that captured the performance of our portfolio allocation strategy, such as PPO RL algorithms and mean squared error (MSE) or a custom loss function that considers risk-adjusted returns.

Mean Squared Error (MSE):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y_i})^2 \tag{10}$$

Optimization of the neural network parameters (weights and biases) using gradient-based optimization techniques like stochastic gradient descent (SGD), Adam, or RMSprop be used

Validation and Hyperparameter Tuning:

We validated the model performance using the validation dataset including monitoring metrics like Sharpe ratio, cumulative returns, or portfolio volatility.

We performed hyperparameter tuning to optimize the neural network architecture, learning rate, batch size, number of epochs, etc. and also use techniques like grid search or random search to explore the hyperparameter space.

Evaluation:

We evaluated the final model on the test dataset to assess its performance on unseen data and compare the performance of the neural network-based portfolio allocation strategy with traditional approaches such as mean-variance optimization or equal-weighted portfolios.

Deployment:

Once satisfied with the performance, we deployed the trained neural network model to make real-time portfolio allocation decisions and monitor the model's performance over time and retrain as necessary to adapt to changing market conditions.

ISSN: XXXX-XXXX IJXX

Overfitting and non-stationarity

The fact that we will be using historical data like time series to be specific, the challenges of overfitting and non-stationarity cannot be escape and as such we intend to deal with it as follows:

1. Overfitting:
    - Cross-validation: the dataset will be split into training and validation sets and techniques like k-fold cross-validation will be sued to evaluate the performance of the model on unseen data.
    - Regularization: we intend to apply techniques like L1 or L2 regularization to penalize large coefficients thereby prevent overfitting.
2. Non-stationarity:
    - Seasonal Adjustment: techniques like seasonal decomposition will be use to try and remove seasonal patterns.
    - Rolling Statistics: we will use rolling averages or rolling standard deviations to smooth out fluctuations which may be caused by non-stationarity.

Backtesting

We tested the performance of the DRL model against the mean variance and the equal weights portfolios when the DRL model was based on the PPO algorithm

Empirical Results

Dataset

For this experiment, we selected 15 assets from yahoo finance following the most performing assets in the last 14 years from January 1, 2010 to January 1, 2024 intuitively and captured the adjusted daily close and converted them into expected returns. The table below shows the head of the downloaded data frame of the 15 assets

Ticker	AAPL	AMZN	GOOG	HD	INTC	JNJ \
Date						
2015-01-02	24.402172	15.4260	26.168653	83.401016	28.073416	80.554398
2015-01-05	23.714722	15.1095	25.623152	81.651253	27.756857	79.991829
2015-01-06	23.716955	14.7645	25.029282	81.401268	27.239561	79.598732
2015-01-07	24.049524	14.9210	24.986401	84.191246	27.810911	81.355965
2015-01-08	24.973553	15.0230	25.065184	86.053917	28.328218	81.995636

ISSN: XXXX-XXXX IJXX

Ticker	JPM	MA	META	NFLX	NVDA	PG \
Date						
2015-01-02	48.268330	80.792534	78.366852	49.848572	4.832578	69.090240
2015-01-05	46.769829	78.519997	77.108192	47.311428	4.750956	68.761765
2015-01-06	45.557133	78.350266	76.069290	46.501431	4.606914	68.448524
2015-01-07	45.626663	79.569046	76.069290	46.742859	4.594910	68.807579
2015-01-08	46.646252	80.806671	78.097137	47.779999	4.767760	69.594437

Ticker	TSLA	UNH	V
Date			
2015-01-02	14.620667	87.571030	62.018215
2015-01-05	14.006000	86.128586	60.649254
2015-01-06	14.085333	85.954796	60.258442
2015-01-07	14.063333	86.832436	61.065796
2015-01-08	14.041333	90.977211	61.884830

Table 1: Stocks Data Frame of daily adjusted close

List of 15 tickers

‘AAPL’, ‘AMZN’, ‘GOOG’, ‘HD’, ‘INTC’, ‘JN’, ‘JPM’, ‘MA’, ‘META’, ‘NFLX’, ‘NVDA’, ‘PG’, ‘TSLA’, ‘UNH’, ‘V’

We calculated the covariance to establish the linear relationship amongst the assets and below is the covariance matrix for the 15 assets

Covariance Matrix for All Assets:

Ticker	AAPL	AMZN	GOOG	HD	INTC	JNJ	JPM \
Ticker							
AAPL	0.084324	0.054972	0.051678	0.037654	0.051861	0.019914	0.035967
AMZN	0.054972	0.110670	0.062523	0.035056	0.047055	0.015116	0.027556

ISSN: XXXX-XXXX IJXX

GOOG	0.051678	0.062523	0.081754	0.034988	0.048019	0.018200	0.034120
HD	0.037654	0.035056	0.034988	0.061295	0.039991	0.018774	0.036153
INTC	0.051861	0.047055	0.048019	0.039991	0.111970	0.020810	0.041940
JNJ	0.019914	0.015116	0.018200	0.018774	0.020810	0.033677	0.021164
JPM	0.035967	0.027556	0.034120	0.036153	0.041940	0.021164	0.077114
MA	0.048541	0.045135	0.046950	0.037899	0.046560	0.021713	0.045590
META	0.060289	0.074129	0.070122	0.040449	0.055715	0.017090	0.034884
NFLX	0.055630	0.077962	0.059988	0.036586	0.054340	0.014437	0.029767
NVDA	0.079386	0.082731	0.075666	0.054154	0.082922	0.019798	0.047775
PG	0.021634	0.015604	0.018567	0.020983	0.022478	0.019151	0.019257
TSLA	0.070397	0.074432	0.060760	0.044634	0.061899	0.013739	0.042006
UNH	0.032933	0.025565	0.029775	0.030644	0.030882	0.023218	0.033941
V	0.043037	0.039662	0.042013	0.035197	0.041095	0.020406	0.041597

Ticker	MA	META	NFLX	NVDA	PG	TSLA	UNH \
Ticker							
AAPL	0.048541	0.060289	0.055630	0.079386	0.021634	0.070397	0.032933
AMZN	0.045135	0.074129	0.077962	0.082731	0.015604	0.074432	0.025565
GOOG	0.046950	0.070122	0.059988	0.075666	0.018567	0.060760	0.029775
HD	0.037899	0.040449	0.036586	0.054154	0.020983	0.044634	0.030644
INTC	0.046560	0.055715	0.054340	0.082922	0.022478	0.061899	0.030882
JNJ	0.021713	0.017090	0.014437	0.019798	0.019151	0.013739	0.023218
JPM	0.045590	0.034884	0.029767	0.047775	0.019257	0.042006	0.033941
MA	0.076418	0.052117	0.045131	0.068395	0.021789	0.055346	0.034337
META	0.052117	0.141398	0.078434	0.088671	0.017890	0.072368	0.027187
NFLX	0.045131	0.078434	0.201734	0.090949	0.015502	0.087610	0.030770
NVDA	0.068395	0.088671	0.090949	0.231897	0.022565	0.115494	0.039747

ISSN: XXXX-XXXX IJXX

PG	0.021789	0.017890	0.015502	0.022565	0.035424	0.014415	0.020543
TSLA	0.055346	0.072368	0.087610	0.115494	0.014415	0.318396	0.031284
UNH	0.034337	0.027187	0.030770	0.039747	0.020543	0.031284	0.067703
V	0.062505	0.045246	0.040811	0.060201	0.020512	0.050173	0.032197

Ticker	V
Ticker	
AAPL	0.043037
AMZN	0.039662
GOOG	0.042013
HD	0.035197
INTC	0.041095
JNJ	0.020406
JPM	0.041597
MA	0.062505
META	0.045246
NFLX	0.040811
NVDA	0.060201
PG	0.020512
TSLA	0.050173
UNH	0.032197
V	0.063157

Table 2: covariance matrix for all assets

ISSN: XXXX-XXXX IJXX

International Journal of xxxxxx
Vol. x, No. x, (20xx), pp. xx-xx

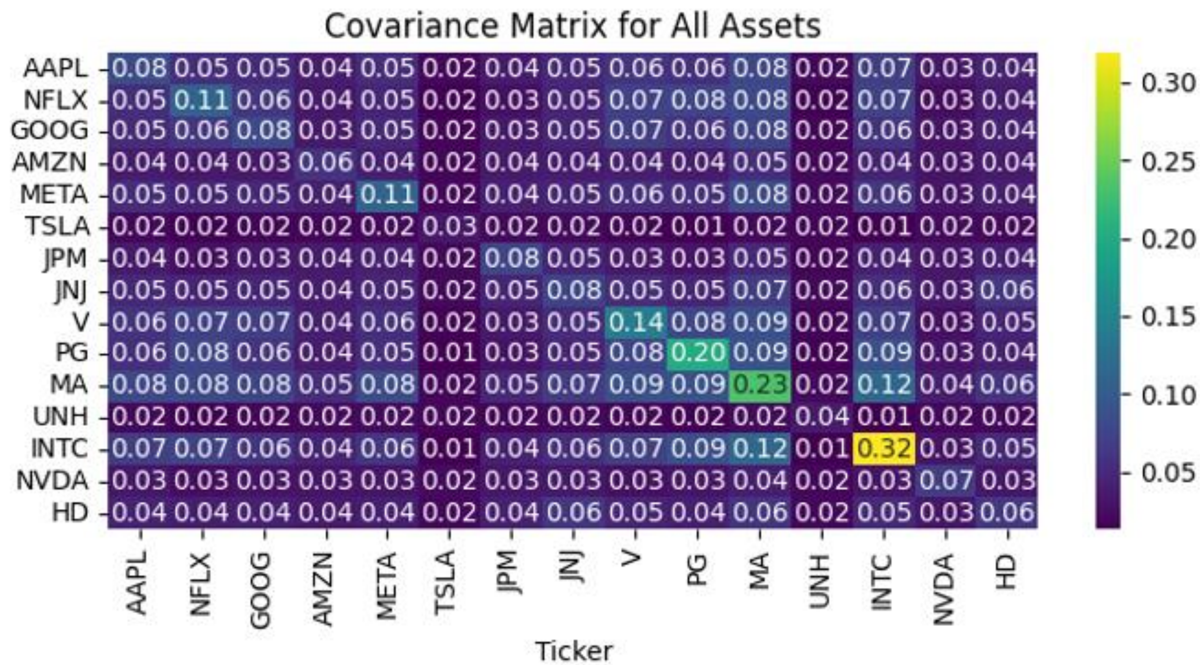

Table 3: covariance matrix for all assets

After using the efficiency frontier, the following assets were found to be the best suit

'MA', 'UNH', 'INTC', 'NVDA', 'HD'

The table below shows the head of the downloaded data frame:

	HD	INTC	MA	NVDA	UNH
Date					
2010-01-04	20.428894	13.605182	23.820492	4.240229	25.497299
2010-01-05	20.578539	13.598674	23.750000	4.302146	25.456848
2010-01-06	20.507273	13.553061	23.715666	4.329666	25.707539
2010-01-07	20.749550	13.422742	23.560692	4.244816	26.694113
2010-01-08	20.649788	13.572608	23.569046	4.253988	26.443430

Table 3: daily adjusted close price

ISSN: XXXX-XXXX IJXX

The model iterated through all the combination of 5 assets in order to come up with the optimal asset combination by comparing the expected returns, risk and sharpe ratios. Below is a matrix of the 5 optimal assets from which a portfolio of optimal asset will be allocated

Covariance Matrix for Optimal Assets:

Ticker	MA	UNH	INTC	NVDA	HD
Ticker					
MA	0.076418	0.034337	0.046560	0.068395	0.037899
UNH	0.034337	0.067703	0.030882	0.039747	0.030644
INTC	0.046560	0.030882	0.111970	0.082922	0.039991
NVDA	0.068395	0.039747	0.082922	0.231897	0.054154
HD	0.037899	0.030644	0.039991	0.054154	0.061295

Table 4: Covariance Matrix for Optimal Assets

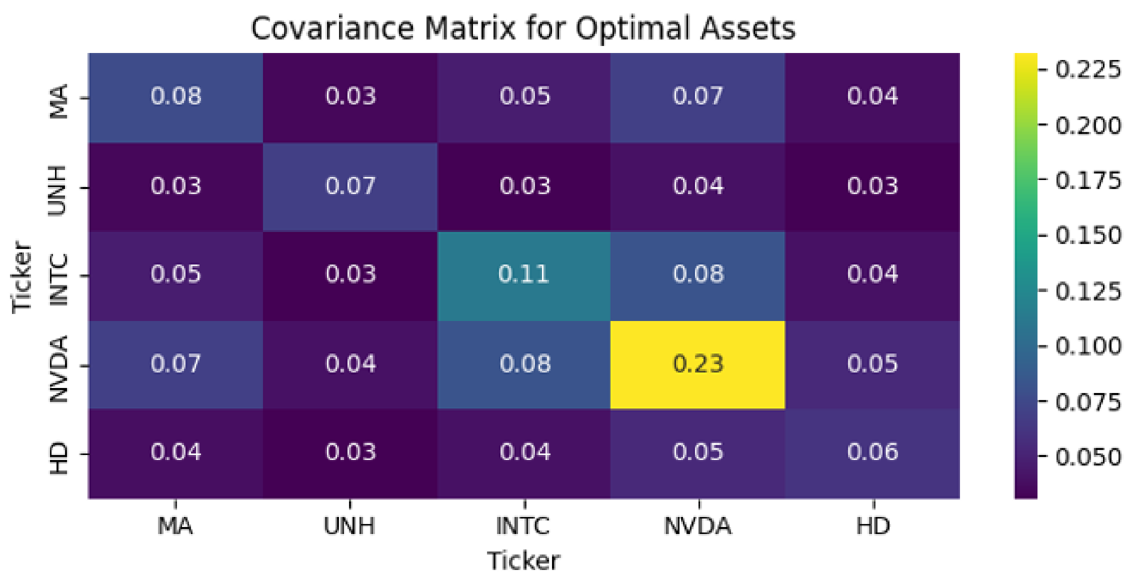

Table 5: Covariance Matrix for Optimal Assets

But then before we trained the model, the discrete allocation was found to be as below:

OrderedDict([('INTC', 12), ('MA', 7), ('UNH', 4)])

Where ‘INTC’ had the largest number of share of about 12 followed by ‘MA’ with 7 and ‘UNH’ had only 4 as is depicted by the histogram below

ISSN: XXXX-XXXX IJXX

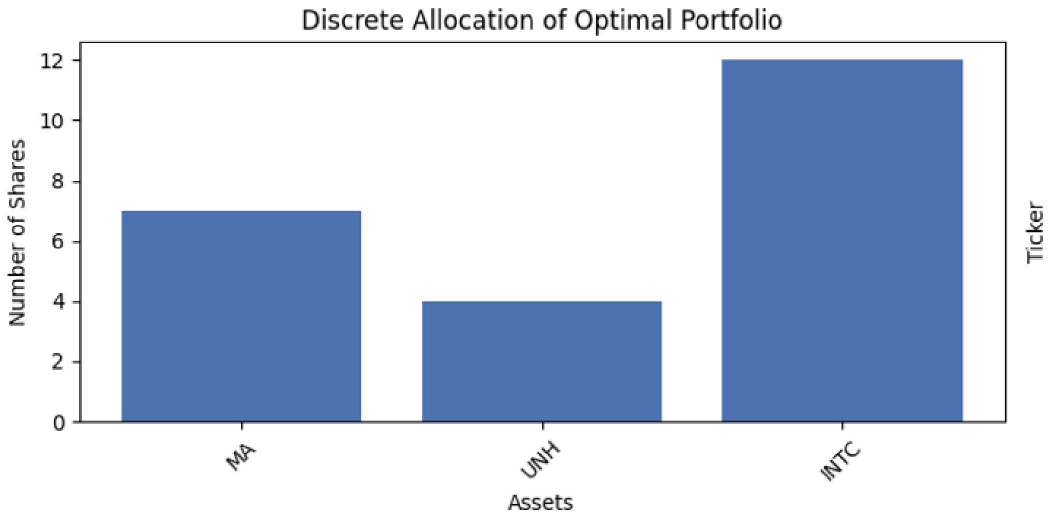

Figure 2: discrete allocation of optimal portfolio

Expected annual return: 51.1%

Annual volatility: 34.9%

Sharpe Ratio: 1.41

But then after training the model, the following parameters were found that

Annualized Return: 2.10%

Annualized Volatility: 16.32%

Sharpe Ratio: 0.13

Winning Days Ratio: 49.71%

Information Ratio: 3.96

ISSN: XXXX-XXXX IJXX

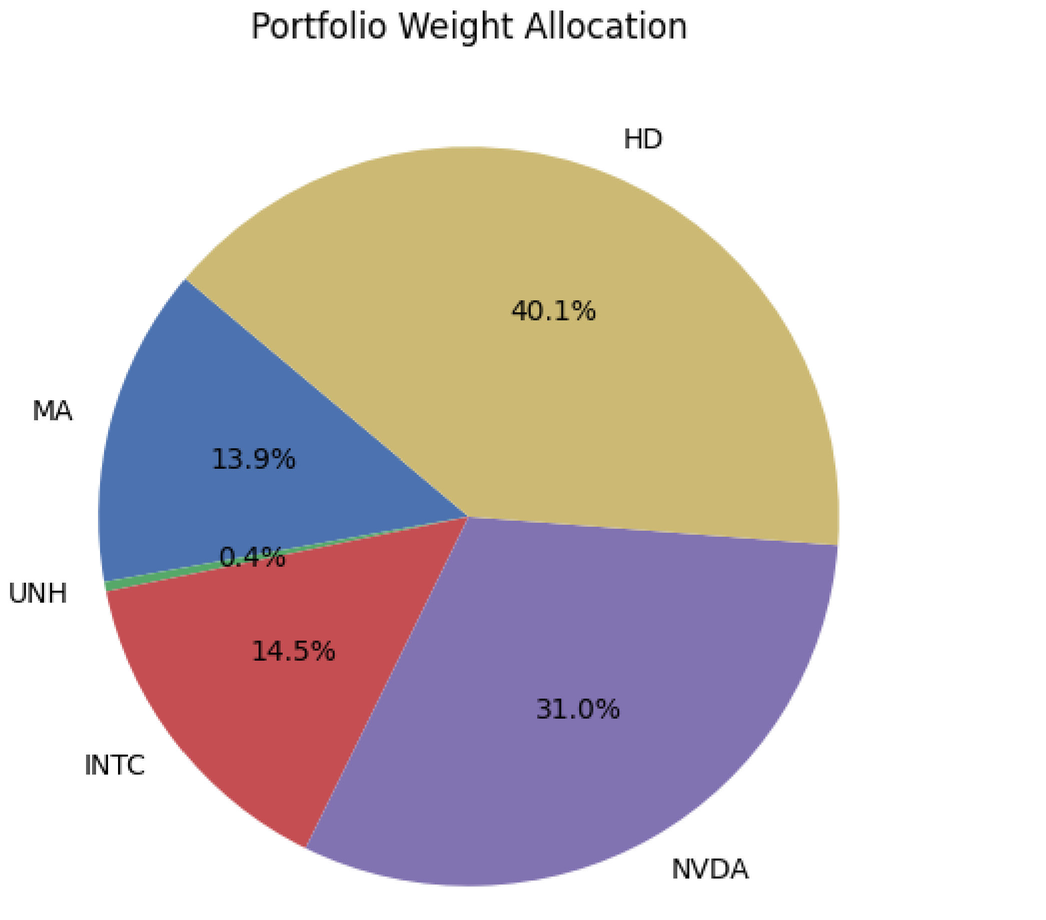

Figure 3: portfolio weight allocation

The figure above shows the weight allocation for the equal weights portfolio and the DRL portfolio rebalance the weights at each every time step or when it iterates using a police that maximizes the portfolio's cumulative value.

Rolling Volatility (6 months) and Rolling Sharpe Ratio

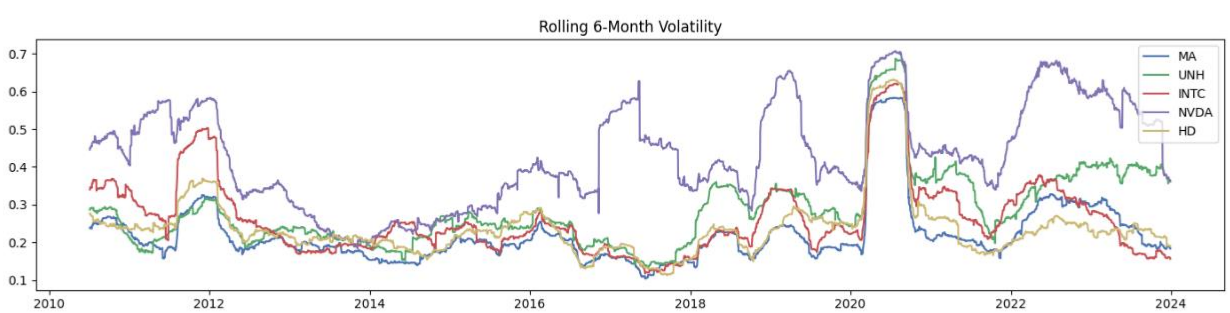

Figure 4: 6 month rolling volatility

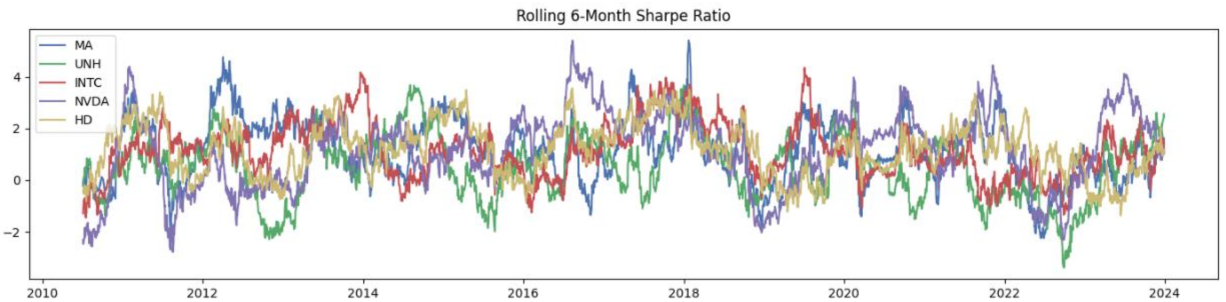

Figure 5: 6 months rolling sharpe ratio

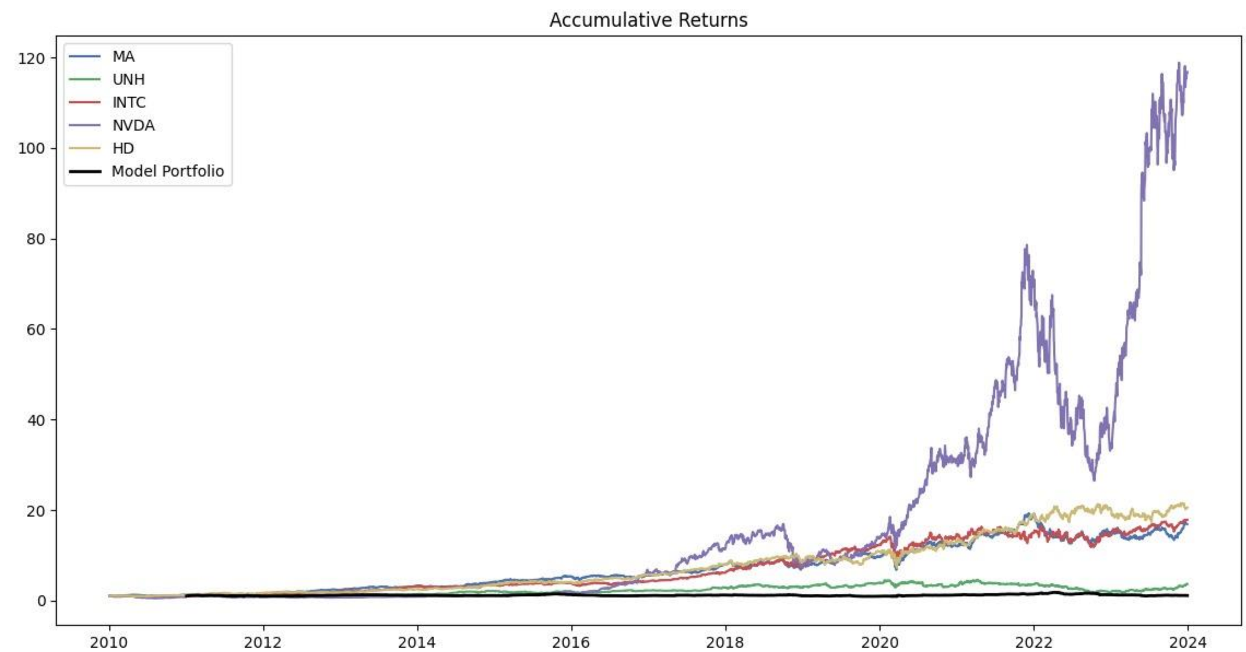

Figure 6: accumulative returns

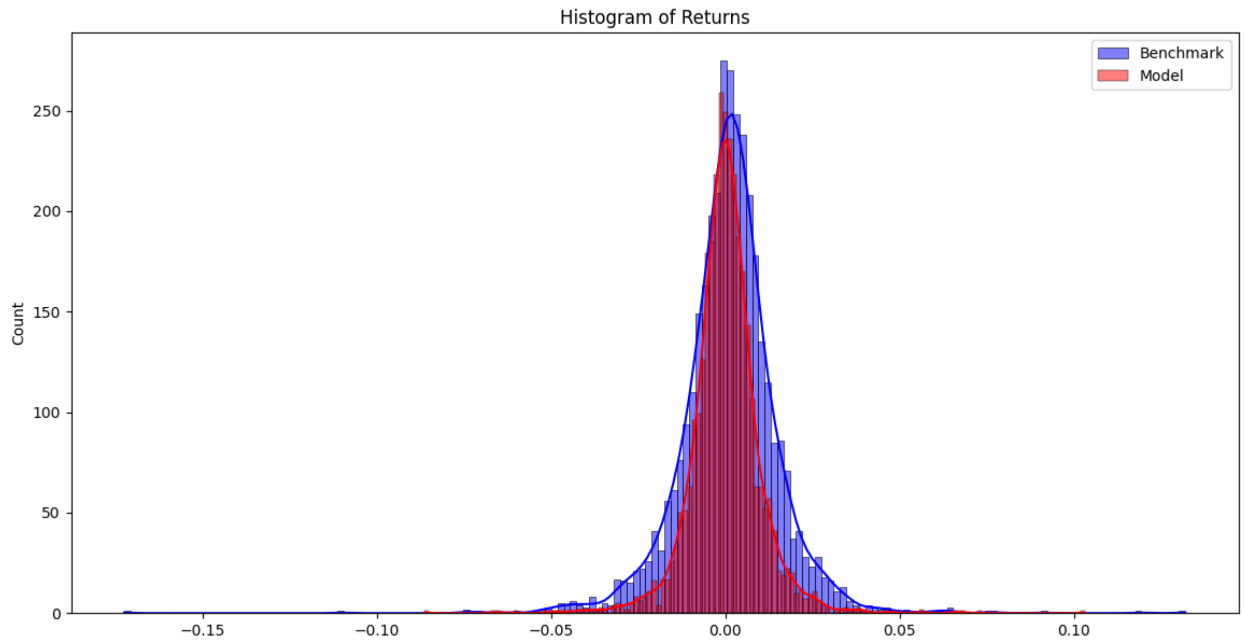

Figure 7: histogram of returns

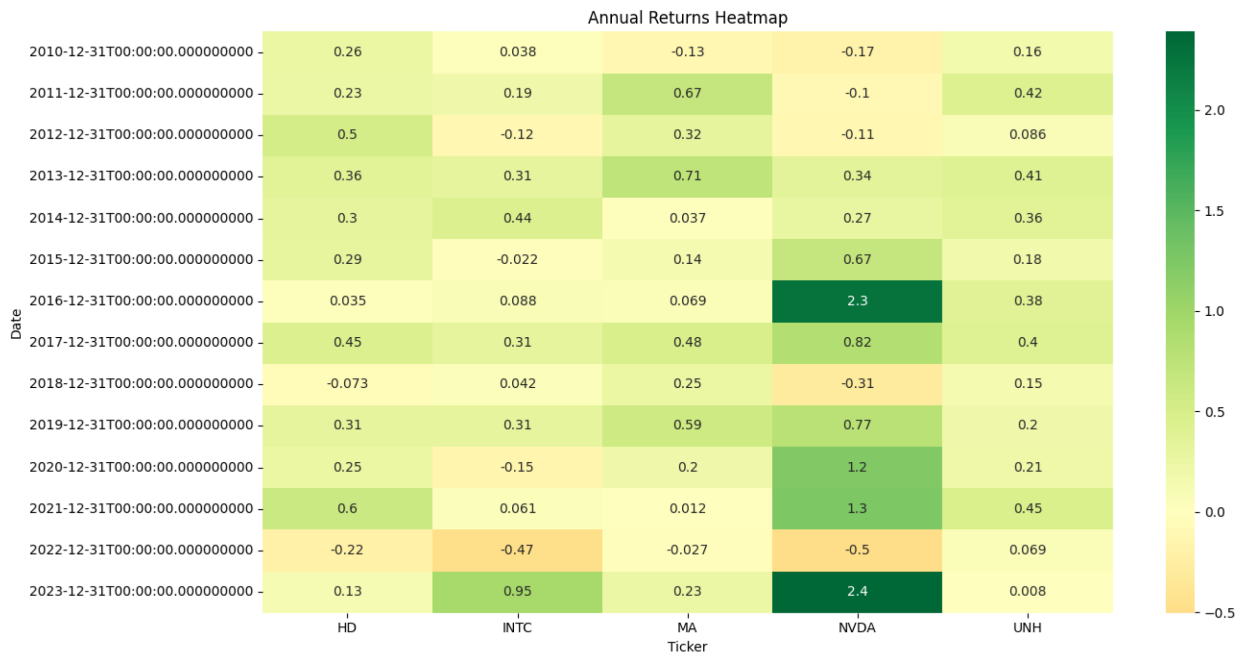

Table 6: heatmap of Annual Returns

Discussion of Results

15 asset where downloaded from yahoo finance and the daily adjusted closed where capture as indicated in table 1 above. These assets were chosen intuitively after observing their performances thereby not ruling completely the input of an asset or portfolio manager who perhaps uses his

ISSN: XXXX-XXXX IJXX

experience and expert knowledge. Covariance matrix was plotted as indicated in tables 2 and 3 which shows the linear relationship between assets in terms of risk and returns – INTC as 0.32 if you follow the longest diagonal in the matrix.

Then the DRL model iterated in combination of 5 in order to come up with the best 5 which suited the combination and table 4 and 5 shows these as optimal. Figure 2 shows also the discrete allocation of optimal portfolio and found that Expected annual return was 51.1%, Annual volatility: 34.9%, Sharpe Ratio: 1.41before the model was trained.

The pie chart in figure 3 shows the weight allocation in order to optimize the portfolio by rebalancing the weights and as such 40.1% was allocated to HD as the highest and only 0.4% to UNH as we started training the model.

Using a 6 months rolling volatility we could see that from about 2010 all the asset showed a high volatility which could probably be due to the 2008 financial crisis with NVDA and INTC showing the riskiest assets, the same trend is seen again around 2018 and peaked in the first half of 2020 where all the asset's volatility was very high implying very high risk which was due to COVID 19, however we a downward trend from 2023 according to figure 4.

Looking at figure 5, we see that around 2010 the sharpe ratio of all the assets was zero or negative meaning the returns in the investment in the said asset was very low when compared to their risk, but we see that in 2016 NVDA recorded the highest sharpe ratio of over 4 and in 2020 it still showed the upward positive trend despite COVID because of the rush to find the COVID cure and so scientists need fast processors to process the data and NVDA happens to manufacture fast processors and graphic cards.

Following the accumulated returns as depicted in figure 6, we see the upward trend from 2015 between 17 and 110 while the other 4 asset remain almost constant a small rise of up to 17. But the rise in the accumulated Return of NVDA could be attributed to the high demand of fast processing processor and gaming graphics. From around 2016 we saw the rise the gaming industry with high demand for fast processors and graphics then during the COVID 19 period there was need for the fast processors and graphics due to the urgent need to find the cure.

Figure 7 shows the histogram of returns between the bench mark and the DRL model, we could see from the histogram that there was no very much difference between the bench mark and our model, however comparing the parameters of the benchmark and the model, we see the model as working okay. The following were the parameters of the benchmark Expected annual return: 51.1%, Annual volatility: 34.9%, Sharpe Ratio: 1.41 while the model had the following after training, Annualized Return: 2.10%, Annualized Volatility: 16.32%, Sharpe Ratio: 0.13.

Table 6 of the heatmap of Annual Returns shows NVDA with the highest annual returns of 2.3 in 2016 and 2.4 in 2023 December

ISSN: XXXX-XXXX IJXX

Conclusion:

The DRL model did perform relatively better than the traditional portfolio allocation methods as it was able to leverage risk while maximizing returns which proved helpful to the portfolio manager. However, with more time and other resources more can be done to the model to improve it.

Challenges and Future Directions:

Challenges in stock market prediction include volatility and nonlinearity, prompting exploration of sophisticated machine learning models and algorithms. Big data analytics and hybrid approaches offer opportunities for improved forecasting, highlighting the increasing role of machine learning in trading and stock market prediction.

ISSN: XXXX-XXXX IJXX

References

[1] Basak S, Kar S, Saha S, et al. Predicting the direction of stock market prices using tree-based classifiers. The North American Journal of Economics and Finance 2019; 47: 552–567. doi: 10.1016/j.najef.2018.06.013

[2] Berislav Ž, Hrvoje J. Forecasting stock market indices using machine learning algorithms. Interdisciplinary Description of Complex Systems: INDECS 2020; 18(4): 471–489. doi: 10.7906/indecs.18.4.6

[3] Chen W, Zhang H, Mehlawat MK, Jia L. Mean-variance portfolio optimization using machine learning—Based stock price prediction. Applied Soft Computing 2021; 100: 106943. doi: 10.1016/j.asoc.2020.106943

[4] Gabriel Molina. Stock trading with recurrent reinforcement learning (RRL). CS229, nd Web, 15, 2016.

[5] Ghasemzadeha M, Mohammad-Karimi N, Ansari-Samani H. Machine learning algorithms for time series in financial markets. Advances in Mathematical Finance and Applications 2020; 5(4): 479–490. doi: 10.22034/AMFA.2020.674946

[6] Gu, S., Wu, B., Liu, X., & Song, L. (2022). Deep Reinforcement Learning with Mean-Variance Optimization for Dynamic Portfolio Allocation. Expert Systems with Applications, 187, 115061.

[7] Janková Z. A bibliometric analysis of artificial intelligence technique in financial market. Scientific Papers of the University of Pardubice. Series D, Faculty of Economics & Administration 2021; 29(3). doi: 10.46585/sp29031268

[8] Jiang, Z., Xu, L., Lyu, H., & Ma, S. (2020). Deep Reinforcement Learning for Dynamic Portfolio Optimization. IEEE Transactions on Neural Networks and Learning Systems, 31(11), 4795-4806.

[9] John Moody, Lizhong Wu, Yuansong Liao, and Matthew Saffell. Performance functions and reinforcement learning for trading systems and portfolios. Journal of Forecasting, 17(5-6):441– 470, 1998.

[10] Khan W, Ghazanfar MA, Azam MA, et al. Stock market prediction using machine learning classifiers and social media, news. Journal of Ambient Intelligence and Humanized Computing 2022; 13: 3433–3456. doi: 10.1007/s12652-020-01839-w

[11] Kilimci ZH, Duvar R. An efficient word embedding and deep learning based model to forecast the direction of stock exchange market using twitter and financial news sites: A case of Istanbul stock exchange (bist 100). IEEE Access 2020; 8: 188186–188198. doi: 10.1109/ACCESS.2020.3029860

[12] Lakshminarayanan SK, McCrae JP. A comparative study of SVM and LSTM deep learning algorithms for stock market prediction. In: Proceedings of the 2019 International Conference on Artificial Intelligence and Computer Science (AICS 2019); 12–13 July 2019; Wuhan, China. pp. 446–457.

[13] Liang, X., Li, S., Wang, J., & Yang, Q. (2021). Risk-Sensitive Deep Reinforcement Learning for Dynamic Portfolio Allocation. IEEE Transactions on Neural Networks and Learning Systems, 32(6), 2226-2238.

[14] Long W, Lu Z, Cui L. Deep learning-based feature engineering for stock price movement prediction. Knowledge-Based Systems 2019; 164: 163–173. doi: 10.1016/j.knosys.2018.10.034

[15] Martin D, Póczos B, Hollifield B. Machine learning-aided modeling of fixed income instruments. In: Proceedings of the Thirty-second Conference on Neural Information Processing Systems; 2–8 December 2018; Montréal Canada.

[16] Michele A, Colin V, Santika DD. Mobilenet convolutional neural networks and support vector machines for palmprint recognition. Procedia Computer Science 2019; 157: 110–117. doi: 10.1016/j.procs.2019.08.147

[17] Modi P, Shah S, Shah H. Big data analysis in stock market prediction. International Journal of Engineering Research & Technology (IJERT) 2019; 8(10). doi: 10.17577/IJERTV8IS100224

[18] Mohan S, Mullapudi S, Sammeta S, et al. Stock price prediction using news sentiment analysis. In: Proceedings of the 2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (Bigdata Service); 4–9 April 2019; Newark, CA, USA. pp. 205–208.

[19] Obthong M, Tantisantiwong N, Jeamwatthanachai W, Wills G. A survey on machine learning for stock price prediction: Algorithms and techniques. In: Proceedings of 2nd International Conference on Finance, Economics, Management and IT Business; 5–6 May 2020; Online Streaming.

[20] Reddy CV. Predicting the stock market index using stochastic time series ARIMA modelling: The sample of BSE and NSE. Indian Journal of Finance 2019; 13(8): 7–25. doi: 10.17010/ijf/2019/v13i8/146301

[21] Reddy VKS. Stock market prediction using machine learning. International Research Journal of Engineering and Technology (IRJET) 2018; 5(10): 1033–1035.

[22] Ren R, Wu DD, Liu T. Forecasting stock market movement direction using sentiment analysis and support vector machine. IEEE Systems Journal 2019; 13(1): 760–770. doi: 10.1109/JSYST.2018.2794462

[23] Zhang, Y., Liu, H., & Zhang, W. (2023). Distributional Reinforcement Learning for Risk-Aware Portfolio Allocation. Expert Systems with Applications, 207, 115633

[24] Zhong X, Enke D. Predicting the daily return direction of the stock market using hybrid machine learning algorithms. Financial Innovation 2019; 5(1): 1–20. doi: 10.1186/s40854-019-0138-0

ISSN: XXXX-XXXX IJXX

Appendix:

Link to colab

https://colab.research.google.com/drive/1kGnKKpkndibDugQH0I-XInO1JHxZL_4I#scrollTo=da0Oa6F1mNJX

ISSN: XXXX-XXXX IJXX