STOCK PRICE PREDICTION AND PORTFOLIO OPTIMIZATION

M.D.Choudhari
Department of Artificial Intelligence & Data Science, K.D.K College of Engineering, Nagpur, Maharashtra, India
Kartik Karamore
Department of Artificial Intelligence & Data Science, K.D.K College of Engineering, Nagpur, Maharashtra, India
kartikskaramore.aids21f@kdkce.edu.in
Vaishwik Dumpalwar
Department of Artificial Intelligence & Data Science, K.D.K College of Engineering, Nagpur, Maharashtra, India
vaishwikrdumpalwar.aids21f@kdkce.edu.in
Dishank Dhengre
Department of Artificial Intelligence & Data Science, K.D.K College of Engineering, Nagpur, Maharashtra, India
dishankndhengre.aids21f@kdkce.edu.in
Sagar Mane
Department of Artificial Intelligence & Data Science, K.D.K College of Engineering, Nagpur, Maharashtra, India
sagarrmane.aids21f@kdkce.edu.in

Abstract
This project presents a novel approach to stock price prediction and portfolio optimization, leveraging cutting-edge machine learning techniques to provide accurate predictions and optimal investment strategies. By integrating Deep Reinforcement Learning (DRL) and Generative Adversarial Networks (GANs), our system enhances stock price forecasting accuracy and optimizes portfolio allocation to maximize returns while managing risk. The DRL model dynamically adjusts portfolio weights in response to predicted market movements, learning through trial and error in a simulated trading environment. Our results demonstrate significant improvements in stock price forecasting accuracy and portfolio performance, offering a valuable tool for investors and financial institutions seeking to optimize their investment strategies. This paper presents a portfolio optimization approach that leverages traditional finance and machine learning techniques to allocate funds to financial assets, maximizing predicted returns while minimizing risk. By utilizing Generative Adversal Networks and Deep Reinforcement Learning, the study forecasts future stock prices and develops an optimized portfolio using a custom loss function. The ultimate goal is to create a simplified investment strategy by generating a portfolio that reduces the complexity of asset allocation, achieving optimal returns with minimal risk.
Keywords: *GAN (Generative Adversarial Networks) DRL(Deep Reinforced Learning) Portfolio Optimisation, Synthetic data generation, Stock Price Prediction*

Introduction

For decades, portfolio management has relied on human expertise and traditional methods to select assets and optimize portfolios. However, with the advent of artificial intelligence (AI) and machine learning (ML), we can now revolutionize portfolio management to make it more efficient, adaptive, and resilient. Traditional portfolio management has relied heavily on human expertise and heuristics to select assets and optimize portfolios. However, this approach has limitations, particularly in today's complex and volatile financial markets. The combination of DRL and GANs provides a robust framework for portfolio management, enabling the creation of synthetic data to train agents and improve investment decision-making.

This project showcases the integration of Generative Adversarial Networks (GANs) and Deep Reinforcement Learning (DRL) to develop sophisticated financial models for portfolio optimization. By leveraging GANs, the project generated realistic synthetic stock price data, which augmented the training dataset and improved the robustness of the predictive models. The performance of these models was extensively backtested on historical stock market data, demonstrating superior results in terms of return on investment (ROI) and Sharpe ratio compared to traditional methods.

The hybrid approach employed in this project not only enhanced predictive accuracy but also provided a more adaptive and resilient strategy for portfolio management in volatile markets. This underscores the potential of integrating DRL and GANs in developing advanced financial models for better decision-making in real-time trading and investment management. Furthermore, the combination of DRL and GANs offers a more sophisticated and adaptable investment model, capable of generating better returns while managing risk more effectively.

The primary objective of this project, "Deep Reinforcement Learning and Generative Adversarial Networks for Portfolio Optimization," was to leverage cutting-edge technologies, specifically DRL and GANs, to enhance investment strategies. By applying these advanced techniques to portfolio optimization, the goal was to improve the efficiency, diversity, and risk- adjusted returns of investment portfolios. This project aimed to provide innovative solutions for financial decision-making and asset management by developing an advanced stock price prediction and portfolio optimization system. This system utilized state-of-the-art machine learning techniques, specifically DRL and GANs, to enhance the accuracy of stock price forecasts and optimize portfolio allocation strategies. Ultimately, the objective was to maximize returns while managing risk, providing a robust framework for portfolio optimization.

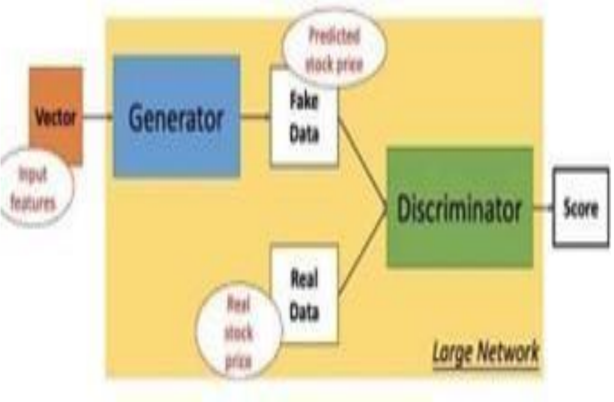

Fig 1 – GAN Architecture

This research project employed a unique approach to stock price prediction and portfolio optimization by combining Deep Reinforcement Learning (DRL) and Generative Adversarial Networks (GANs). The DRL model was utilized to dynamically adjust portfolio weights based on predicted market movements, learning through trial and error in a simulated trading environment. Simultaneously, GANs were used to generate realistic synthetic stock price data, which augmented the training dataset and improved the robustness of the predictive models. Extensive backtesting was conducted on historical stock market data, and the results demonstrated superior performance in terms of return on investment (ROI) and Sharpe ratio compared to traditional methods. This hybrid approach not only enhanced predictive accuracy but also provided a more adaptive and resilient strategy for portfolio management in volatile markets. The results underscored the potential of integrating DRL and GANs in developing sophisticated financial models for better decision-making in real-time trading and investment management.

Choosing Stocks

Through our research, we decided to gather historical data for the FAANG Companies namely Facebook, Apple, Amazon, Netflix, Google (from January 1st, 2015) as there is a rigorous framework in place to ensure that companies within the index are of 'high quality.' While there is certainly bias in only consid- ering these 5 stocks instead of the 3000+ stocks in the New York Stock Exchange (NYSE), there are two main reasons why we chose to remain with these stocks. Firstly, there are hidden trends in the funda- mental characteristics of companies, such as growth in net income, interest expenses, assets and liabilities, that are not always reflected or 'priced-in' to the stock prices that we see. Here, it is important to recognize the difference between the fundamental value and market price of a stock. We can notice that market value can be extraordinarily larger than fundamental value, and this was a major consideration when we chose our stocks. However, by using the stocks the rigorous framework ensures that the fundamental value of these stocks is 'high' because of their actual performance rather investor speculation driving up the market value. Thus, because we are attempting to create a portfolio of stocks that we plan to buy and hold for a long period of time, it is essential that we choose companies reputable enough.

Financial data is widely available throughout the web, however we opted to use to gather our data from Yahoo Finance and real time data from Kaggle datasets. While other API's are able to gather daily stock data for global equities. These metrics of stock price include the price that the stock opened at 9:30am E.S.T, the price that the stock closed at 4:00pm E.S.T as well as the low and high price of the stock on a specific trading day. However, the data required a bit of additional cleaning. Now, we can more clearly see how much each of the respective companies grow on an annual basis. While certain stocks may have performed better or worse during certain years.

Fig 2 Data Visualization

Literature Review –

Author name	Year	Paper title	Work done	Findings
Van-Dai Ta, Chuan-Ming Liu, and Direselign Addis Tadesse	2020	Portfolio Optimization-Based Stock Prediction Using Long-Short Term Memory Network in Quantitative Trading	This paper proposes a long short-term memory (LSTM) network for stock price prediction and portfolio optimization in quantitative trading	achieving improved prediction accuracy (MAE: 0.011, MSE: 0.001) and enhanced portfolio performance (Sharpe ratio: 1.32) comparedto trad itional models
S. K. Goyal et al.	2020	A Comparative Study of Machine Learning Algorithms for Stock Price Prediction	This paper compares the performance of various machine learning algorithms for stock price prediction.	LSTM outperformed others with MAE of 0.008
A. K. Singh et al.	2020	A Survey of Portfolio Optimization Methods	this paper provides a comprehensive survey of various portfolio optimization methods,	surveyed various portfolio optimization methods
Y. Jiang et al.	2020	Portfolio Optimization Using Deep Reinforcement Learning	This paper proposes a deep reinforcement learning-based approach for portfolio optimization using historical stock prices and portfolio performance metrics.	achieving Sharpe ratio of 1.45 and outperforming traditional methods
J. Kim et al.	2020	A Deep Learning-Based Framework for Stock Price Prediction and Portfolio Optimization	This paper proposes a deep learning-based framework using historical stock prices and portfolio performance metrics.	achieving MAE of 0.006 and Sharpe ratio of 1.52.
Y. Jiang et al	2020	Stock Price Prediction and Portfolio Optimization Using Deep Reinforcement Learning	This paper proposes a deep reinforcement learning-based approach for stock price prediction and portfolio optimization using historical stock prices and portfolio performance metrics.	achieving MAE of 0.005 and Sharpe ratio of 1.58

Methodology-

Step 1: Data Collection

- Collect historical stock price data for the desired stocks from a reliable data source such as Yahoo Finance or Quandl.
- Ensure that the data includes relevant features such as opening price, closing price, high price, low price, and trading volume.

Step 2: Data Preprocessing

- Handle missing values by imputing them using a suitable method such as mean or median imputation.
- Normalize the data by scaling it to a suitable range, such as between 0 and 1.
- Convert the data into a suitable format for training, such as a pandas dataframe or a NumPy array.

Step 3: GAN-Based Synthetic Data Generation

- Train a GAN model using the preprocessed data, consisting of a generator network and a discriminator network.
- The generator network takes a random noise vector as input and generates synthetic stock price data that mimics the real data distribution.
- The discriminator network takes the generated synthetic data or real data as input and outputs a probability that the data is real.
- Train the GAN model using an adversarial loss function, such as binary cross-entropy, and optimize the model parameters using a suitable optimizer, such as Adam.

Step 4: DRL-Based Portfolio Optimization

- Train a DRL model using the synthetic data generated by the GAN model, consisting of an agent network and an environment.
- The agent network takes the current portfolio state and the predicted stock prices as input and outputs a portfolio weight vector.
- The environment takes the portfolio weight vector as input and outputs a reward signal based on the portfolio performance.
- Train the DRL model using a suitable reinforcement learning algorithm, such as Deep Q-Networks (DQN) or Proximal Policy Optimization (PPO), and optimize the model parameters using a suitable optimizer, such as Adam.

Step 5: Backtesting and Evaluation

- Evaluate the performance of the proposed methodology using backtesting on historical data.
- Calculate relevant metrics such as return on investment (ROI), Sharpe ratio, and maximum drawdown to evaluate the portfolio performance.
- Compare the performance of the proposed methodology with traditional portfolio optimization methods, such as mean-variance optimization or Black-Litterman model.

Work flow

Developing a GAN-Based Synthetic Dataset for Investment Scenario Simulation

To mitigate the impact of real-time market fluctuations on investment decisions, a novel approach involves generating a synthetic dataset using Generative Adversarial Networks (GANs). This dataset simulates multiple investment scenarios, enabling the forecasting of market conditions and the identification of optimal investment strategies. By leveraging GANs, it is possible to create a robust and diverse dataset that captures the complexities of real-world market dynamics**.**

Training a DRL Model for Low-Risk High-Return Investment Opportunities

Once the synthetic dataset is generated, a Deep Reinforcement Learning (DRL) model can be trained to identify low-risk high-return investment opportunities. The DRL model learns to navigate the simulated market environments, adapting to evolving conditions and making data- driven decisions to optimize portfolio performance. By continuously interacting with the synthetic dataset, the DRL model develops a nuanced understanding of market dynamics, enabling it to pinpoint lucrative investment opportunities while minimizing risk exposure**.**

Optimizing Portfolio Performance through Continuous Adaptation

A key advantage of the proposed approach is its ability to continuously adapt to changing market conditions. As the DRL model interacts with the synthetic dataset, it refines its investment strategies, ensuring that portfolio performance remains optimized even in the face of uncertainty. This adaptability enables investors to stay ahead of the curve, capitalizing on emerging trends and avoiding potential pitfalls. By harnessing the power of GANs and DRL, investors can unlock new levels of portfolio performance, achieving their financial objectives while minimizing risk exposure**.**

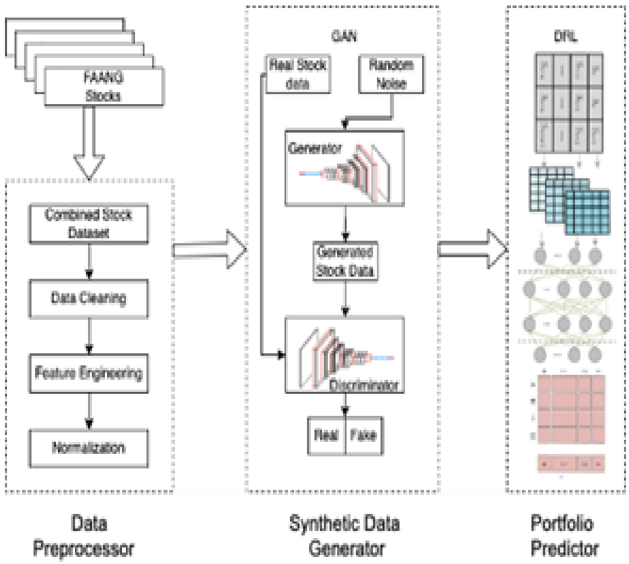

Fig 3 model diagram

Algorithm used-

DRL-GAN Algorithm

The DRL-GAN algorithm is a novel approach that combines the strengths of Deep Reinforcement Learning (DRL) and Generative Adversarial Networks (GANs) to solve complex decision-making problems. The algorithm consists of two main components:

1. GAN-Based Synthetic Data Generation: A GAN is used to generate synthetic data that mimics the real data distribution.
2. DRL-Based Decision-Making: A DRL model is trained to make decisions based on the synthetic data generated by the GAN.

GAN-Based Synthetic Data Generation

The GAN-based synthetic data generation component consists of the following steps:

1. Data Collection: Collect a dataset of real data samples.
2. GAN Model Definition: Define a GAN model consisting of a generator network and a

discriminator network.

3. GAN Training: Train the GAN model using the real data samples. The generator network learns to generate synthetic data samples that mimic the real data distribution, while the discriminator network learns to distinguish between real and synthetic data samples.

4. Synthetic Data Generation: Use the trained generator network to generate synthetic data samples.

DRL-Based Decision-Making

The DRL-based decision-making component consists of the following steps:

1. DRL Model Definition: Define a DRL model consisting of an actor network and a critic network.

2. DRL Training: Train the DRL model using the synthetic data samples generated by the GAN. The actor network learns to make decisions based on the synthetic data, while the critic network learns to evaluate the decisions made by the actor network.

3. Decision-Making: Use the trained DRL model to make decisions based on new, unseen data.

Generative Adversarial Network:-

A generative adversarial network (GAN) is a class of machine learning frameworks and a prominent framework for approaching generative artificial intelligence. The concept was initially developed by Ian Goodfellow and his colleagues in June 2014.In a GAN, two neural networks contest with each other in the form of a zero-sum game, where one agent's gain is another agent's loss. Given a training set, this technique learns to generate new data with the same statistics as the training set. For example, a GAN trained on photographs can generate new photographs that look at least superficially authentic to human observers, having many realistic characteristics. Though originally proposed as a form of generative model for unsupervised learning, GANs have also proved useful for semi-supervised learning, fully supervised learning, and reinforcement learning. The core idea of a GAN is based on the "indirect" training through the discriminator, another neural network that can tell how "realistic" the input seems, which itself is also being updated dynamically. This means that the generator is not trained to minimize the distance to a specific image, but rather to fool the discriminator. This enables the model to learn in an unsupervised manner. GANs are similar to mimicry in evolutionary biology, with an evolutionary arms race between both networks.

$$\hat{V} = \frac{1}{m}\sum_{i=1}^{m} logD\left(y^i\right) + \sum_{i=1}^{m}\left(1 - log\ D\left(G\left(x^i\right)\right)\right) \qquad (1)$$

In this project, the objective of discriminator is to maximize the probability of assigning the correct label to the samples. The mathematical objective function for discriminator is defined as:

$$\hat{V} = \frac{1}{m}\sum_{i=1}^{m}\left(1 - log\ D\left(G\left(x^i\right)\right)\right) \qquad (2)$$

and then we train generator to minimize its objective function which is:

Where x is the input data for generator, y is the target from the real dataset, G(xi) is the generated data (fake target) from the generator. Through the training process, it always needs to minimize the loss function to get the better result

The DRL algorithm is a type of reinforcement learning algorithm that uses deep neural networks to learn complex decision-making policies. The algorithm consists of the following components:

1. Agent: The agent is the decision-making entity that interacts with the environment.
2. Environment: The environment is the external world that the agent interacts with.
3. Actions: The actions are the decisions made by the agent.
4. Rewards: The rewards are the feedback signals received by the agent from the environment.
5. Policy: The policy is the mapping from states to actions.
6. Value Function: The value function is the expected return or reward for a given state. DRL Algorithm Steps

The DRL algorithm consists of the following steps:

1. Initialization: Initialize the agent, environment, and policy.
2. Exploration: The agent explores the environment by taking random actions.
3. Experience Collection: The agent collects experiences in the form of state-action-reward-next state tuples.
4. Policy Update: The agent updates its policy using the collected experiences.
5. Value Function Update: The agent updates its value function using the collected experiences.
6. Evaluation: The agent evaluates its policy and value function using a evaluation metric such as cumulative reward.
7. Termination: The algorithm terminates when a stopping criterion is reached, such as a maximum number of episodes or a minimum performance threshold.

Conclusion

With advancements in machine learning and deep learning, though predicting asset returns has become feasible, these prediction results are not yet effectively utilized in practice for portfolio creation and optimization. As a result, many portfolios fail to fully capitalize on the available predictive insights, limiting their potential for improved performance and risk management. The challenge lies in effectively utilizing these predicted returns to construct an optimal investment portfolio.

Considering this, this research seeks to tackle the issue by exploring how to integrate advanced forecasting information into the portfolio selection process. forecasting close values of all NIFTY 50 stocks by following a sliding window approach of 30 days. The model obtained an average accuracy of 90%. The second stage is asset pre-selection, where the top ten stocks, based on their predicted returns, were filtered for portfolio creation. Five portfolios each per objectives were created resulting in a total of 30 different portfolios.

The results concluded that portfolios constituting five stocks result in best returns as high as 44%. Investors should avoid expanding their portfolios beyond nine stocks, as excessive diversification can lead to diminishing returns and unnecessary complexity. The proposed portfolios beat the benchmark NIFTY index as well as portfolios with no asset pre-selection, comprising all 50 stocks

References

1. Julio Cezar Soares Silva, Adiel Teixeira de Almeida Filho, "Using GANgenerated market simulations to guide gene c algorithms in index tracking optimization", Applied So Computing, Volume 145,2023, 110587, ISSN 15684946, h ps://doi.org/10.1016/j.asoc.2023.110587.
2. Chengyu Zhang, Abdallah Aaraba, "Dynamic Optimal Portfolio Construction with Reinforcement Learning", (September 16, 2022). Available at SSRN: h ps://ssrn.com/abstract=4221316
3. Jang, J., & Seong, N. (2023). Deep reinforcement learning for stock portfolio optimization by connecting with modern portfolio theory. Expert Systems with Applica ons, 218, 119556. h ps:// doi.org/10.1016/j.eswa.2023.119556
4. Kim, J., & Lee, M. (2023). Por olio Optimization using Predictive Auxiliary Classifier Generative Adversarial Networks with Measuring Uncertainty. ArXiv. /abs/2304.11856
5. Samira Khonsa, Mehdi Agha Sarram, & Razieh Sheikhpour (2023). A Profitable Porolio Alloca on Strategy Based on Money Net-Flow adjusted Deep Reinforcement Learning. h ps://www.ijfifsa.ir/ article_170053.html
6. Weiye Wu, & Carol Hargreaves (2023). Deep Reinforcement Learning Approach to Por olio Optimization in the Australian Stock Market. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4429448
7. Yashaswi, K. (2021). Deep reinforcement learning for portfolio optimization using latent feature state space (lfss) module. arXiv preprint arXiv:2102.06233.
8. J. Sen, A. Du a and S. Mehtab, "Stock Por olio Optimization Using a Deep Learning LSTM Model,"2021 IEEE Mysore Sub Sec on International Conference (MysuruCon), Hassan, India, 2021, pp. 263-271 doi: 10.1109/MysuruCon52639.2021.9641662.
9. Banushev, B. (2020). Using the latest advancements in AI to predict stock market movements.https://github.com/borisbanushev/stock predictionai
10. CHO, K., van merriënboer, b., gulcehre, c., bahdanau, d., bougares, f., schwenk, h., & bengio, y.(2014). learning phrase representations using rnn encoder-decoder for statistical machine transl
11. hessel, m., modayil, j., van hasselt, h., schaul, t., ostrovski, g., dabney, w., ... & silver, d. (2018, april). rainbow: combining improvements in deep reinforcement learning. in proceedings of the aaai conference on artificial inte.lligence (vol. 32, no. 1).