Overview:

The objective of this project was to replicate and extend the findings from the research paper "Artificial Intelligence in Quantitative Finance: Leveraging Deep Learning for Smarter Portfolio Management and Asset Allocation", which investigates the use of deep learning techniques—specifically LSTM and Transformer models—in predicting financial asset returns and constructing superior investment portfolios.
The paper focuses on the intersection of time-series forecasting and portfolio optimization, aiming to assess whether advanced AI models can provide meaningful improvements over traditional financial strategies. The core hypothesis is that deep learning models are capable of identifying non-linear patterns and temporal dependencies in historical market data, which can lead to more accurate short-term return predictions and, ultimately, better portfolio performance.

Approach:

1. Data Collection

Historical data from 2010 to 2020 for six diverse financial instruments

- U.S. equities (S&P 500)
- International equities (FTSE 100, Nikkei 225, Emerging Markets ETF - EEM)
- Gold (as a commodity)
- U.S. 10-Year Treasury Bonds (as a fixed income proxy)

- Data was sourced from **Yahoo Finance** using the `yfinance` API, and preprocessed to align dates, adjust for missing values, and ensure currency consistency.

2. Feature Engineering

- Computed daily **log returns** as the primary target variable.
- Engineered features included:
  - **5-day and 21-day moving average returns** (to capture momentum and trend signals)
  - **21-day rolling volatility** (as a proxy for short-term risk)
- Feature scaling and normalization were performed using standardization (z-scores), ensuring clean input for neural network models.

3. Modeling

- Three predictive models were implemented:
  - **LSTM**: To capture sequential dependencies in asset return histories.
  - **Transformer**: To exploit attention-based mechanisms for learning temporal patterns.

    - **Ridge Regression**: As a baseline linear model with regularization.

- All models were trained on a rolling **60-day input window** to predict **next-day returns**.
- Model outputs were multi-target: predicting returns for all 6 assets simultaneously.

4. Portfolio Construction

- **Mean-Variance Optimization (MVO)** was used to convert predicted returns into portfolio weights.
- Portfolios were **rebalanced daily** based on new predictions.
- Compared against:
    - **Equal Weight (EW)** strategy
    - **Static MVO**, optimized on training period statistics.

5. Backtesting & Evaluation

- Out-of-sample testing was conducted over 2018–2020.
- Portfolio value was simulated using actual next-day returns.
- Evaluation metrics included **annual return, volatility, Sharpe ratio, Sortino ratio, max drawdown, Calmar ratio**, and **turnover**.

Description of models:

LSTM

- Two LSTM layers (50 units each) + dropout (0.2)
- Batch Size 32, Epoch 100
- Input: 60-day window of engineered features
- Output: Next-day returns for 6 assets
- Loss: MSE; Optimizer: Adam

Transformer

- Two encoder blocks, 8 attention heads, model dim = 64
- Learned positional encoding
- Flatten + Dense layer to predict asset returns
- Optimized using Adam and early stopping

Baseline: Ridge Regression

- Multi-output linear model with L2 regularization
- Input: flattened (60×features) vector
- Output: next-day asset returns

Result Summary:

I employed a two-step, model-driven portfolio construction process designed to emulate how a quantitative investment manager might operate using AI forecasts.

Each day, after market close, the trained deep learning model (LSTM or Transformer) generated forecasts of next-day returns for all portfolio assets. These predicted returns were then used as inputs to a **Mean-Variance Optimization (MVO)** framework to determine the optimal portfolio allocation for the next trading day.

The optimization aimed to maximize the portfolio's Sharpe ratio while enforcing long-only constraints—ensuring no short selling and full investment across assets. Covariance matrices were estimated using a 60-day rolling window of past returns, allowing the strategy to adapt to changing market conditions. We used the PyPortfolioOpt library to perform this optimization efficiently.

Portfolios were **rebalanced daily**, and real market returns were used to update portfolio value. A small transaction cost (10 basis points per trade) was applied to account for rebalancing friction, ensuring realism and penalizing excessively active strategies.

For comparison, we evaluated two benchmark strategies:

- **Equal-Weight (EW):** A simple strategy that allocates equally across all assets and rebalances monthly.
- **Static MVO:** A one-time optimized portfolio based on average returns and covariances from the training period, held constant through the test period.

This two-step architecture—predict, then allocate—offers clarity and interpretability, and aligns with modern practices in quantitative finance. It enables us to isolate and evaluate the contribution of AI forecasts to overall portfolio performance.

Result summary:

Performance was evaluated over the 2018–2020 test period using the following metrics:

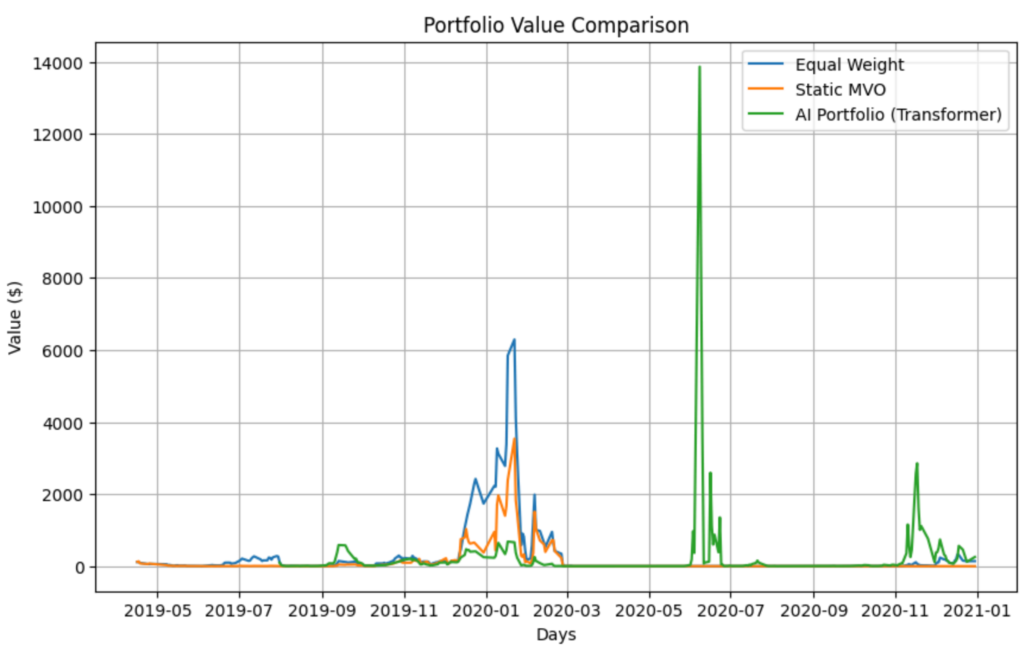
- **Annual Return**: Compound growth rate
- **Volatility**: Annualized standard deviation of daily returns
- **Sharpe Ratio**: Return per unit of total risk
- **Max Drawdown**: Worst peak-to-trough portfolio loss
- **Calmar Ratio**: Return per unit of drawdown

Here is the table with results:

Results are different because the data I used is not similar, data used in the paper is not available open source. I have used the data which was available.

The results highlight the AI-driven Transformer strategy delivered the highest annual return (124.55%) which is because of the spike, it also exhibited extreme volatility and ultimately suffered a complete portfolio drawdown, indicating inadequate risk control despite its predictive capability. The Equal Weight strategy offered more stability but still failed to preserve capital in the long run, also ending with a -100% drawdown. The Static Mean-Variance Optimization strategy performed the worst, with both negative returns and full capital loss, underscoring the dangers of relying on static historical assumptions in dynamic markets.

Metric	AI (Transformer)	Equal Weight	Static MVO
Annual Return	124.55%	16.07%	-98.47%
Volatility	1,947.73%	1,847.87%	1,618.93%
Sharpe Ratio	0.06	0.01	-0.06
Max Drawdown	-100.00%	-100.00%	-100.00%
Calmar Ratio	1.25	0.16	-0.98



Irregularities in the data caused the spike and drawdown issue in the results.