AI-Portfolio-Manager

Explanation Report

1. Approach and Methodology

I worked step by step to build and test an AI-driven portfolio optimization system.

Step 1: Collecting Data
I collected historical financial data (daily price data of selected stocks and indices).
The raw data files are stored in the data/raw folder. After cleaning, I stored them in the data/processed folder.

Step 2: Preprocessing the Data
I removed missing values, aligned all time series to the same dates, and normalized the input data so that my models could learn patterns more easily.

Step 3: Feature Preparation
I used a sliding window (lookback) to create input sequences.
For each day, I used the previous N days of returns as input to predict the next day's returns.

Step 4: Building Models
I used two deep learning models:

- LSTM (Long Short-Term Memory
- Transformer
  Both models were trained on the same dataset for a fair comparison.

Step 5: Predictions to Portfolio Weights
The models predict expected returns.
I converted these predictions into valid portfolio weights by:

- Removing negative values.
- Normalizing them so that all weights add up to 1.

Step 6: Calculating Portfolio Performance
Using the predicted weights, I simulated portfolio returns.
I compared these AI-based strategies against a simple "hold" strategy.

2. Description of Models Used

1. **LSTM (Long Short-Term Memory) Model**

The LSTM model is designed to learn from sequences of past returns and predict future returns. It processes data step by step, remembering important patterns while ignoring noise.

Architecture Details:

Layer (Type)	Output Shape	Parameters
LSTM	(None, 60, 50)	16,200
Dropout	(None, 60, 50)	0
LSTM	(None, 50)	20,200
Dropout	(None, 50)	0
Dense	(None, 6)	306

Total Parameters: 36,706
Trainable Parameters: 36,706
Non-trainable Parameters: 0

How it works in my project:

- The first LSTM layer reads sequences and outputs hidden features for each time step.
- A Dropout layer reduces overfitting by randomly turning off some connections.
- The second LSTM layer compresses information into a smaller vector.
- Another Dropout layer adds regularization.
- Finally, a Dense layer predicts returns for all 6 assets at once.

2. **Transformer Model**

The Transformer model uses self-attention to focus on the most important parts of the input sequence. Instead of processing data step by step like an LSTM, it looks at the entire sequence at once and learns which time steps are most relevant.

Architecture Details:

- Input: A sequence of length 30 with 30 features.
- Multi-Head Attention Layer: Captures relationships between different time steps.
- Layer Normalization: Stabilizes and speeds up training.
- Feed-Forward Layers: Dense layers with ReLU activation to extract deeper patterns.
- Global Average Pooling: Reduces the sequence into a single vector.
- Output Layer: A Dense layer that outputs predictions for 6 assets.

Key Points:

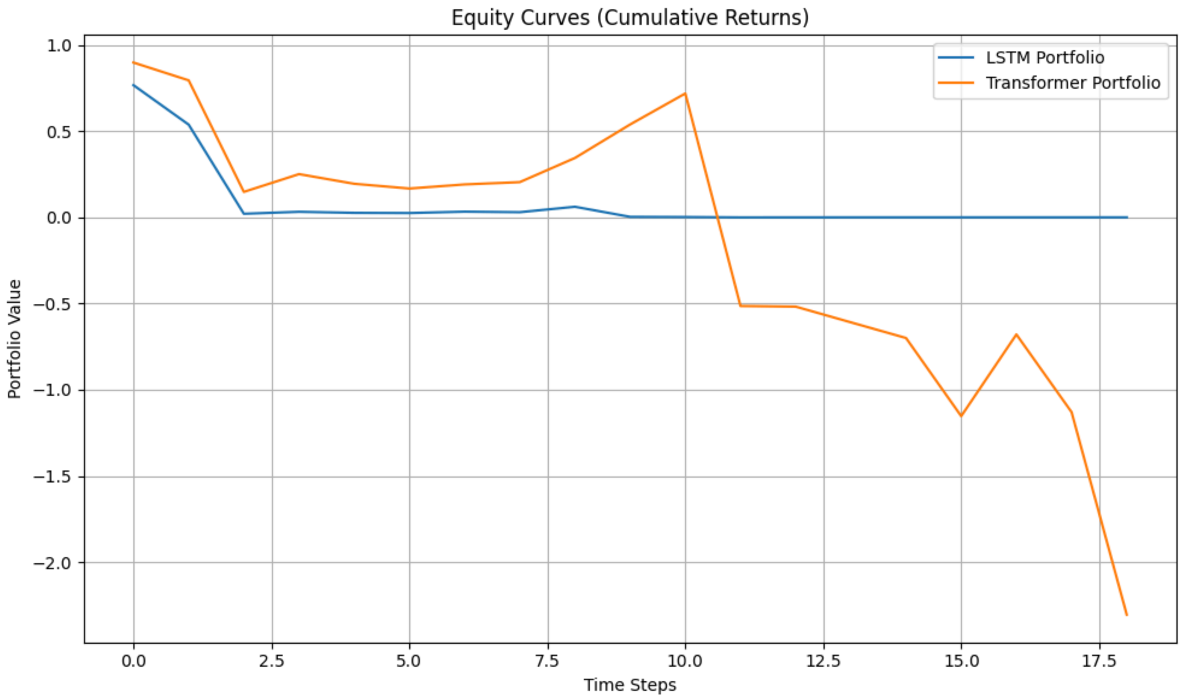
- Multi-head attention helps the model learn long-range dependencies.
- Dense layers transform features into meaningful representations.
- Global pooling and the final dense output generate portfolio predictions.

3. Challenges Faced and Solutions

Challenge	How I solved it
Missing values in data	I filled or removed missing data and aligned dates.
Models predicting negative values	I clipped negative predictions to zero and re-normalized weights.
Overfitting during training	I used a validation split and early stopping to avoid overfitting.
Managing folder structure in Git	I cleaned up the repository, moved all scripts into scripts/, and updated the .gitignore.

4. Summary of Results and Metrics

1. **Equity Curves (Cumulative Returns)**



- The **blue line** represents the LSTM Portfolio over time.
- The **orange line** represents the Transformer Portfolio.
- This graph shows how the portfolio values change step by step.

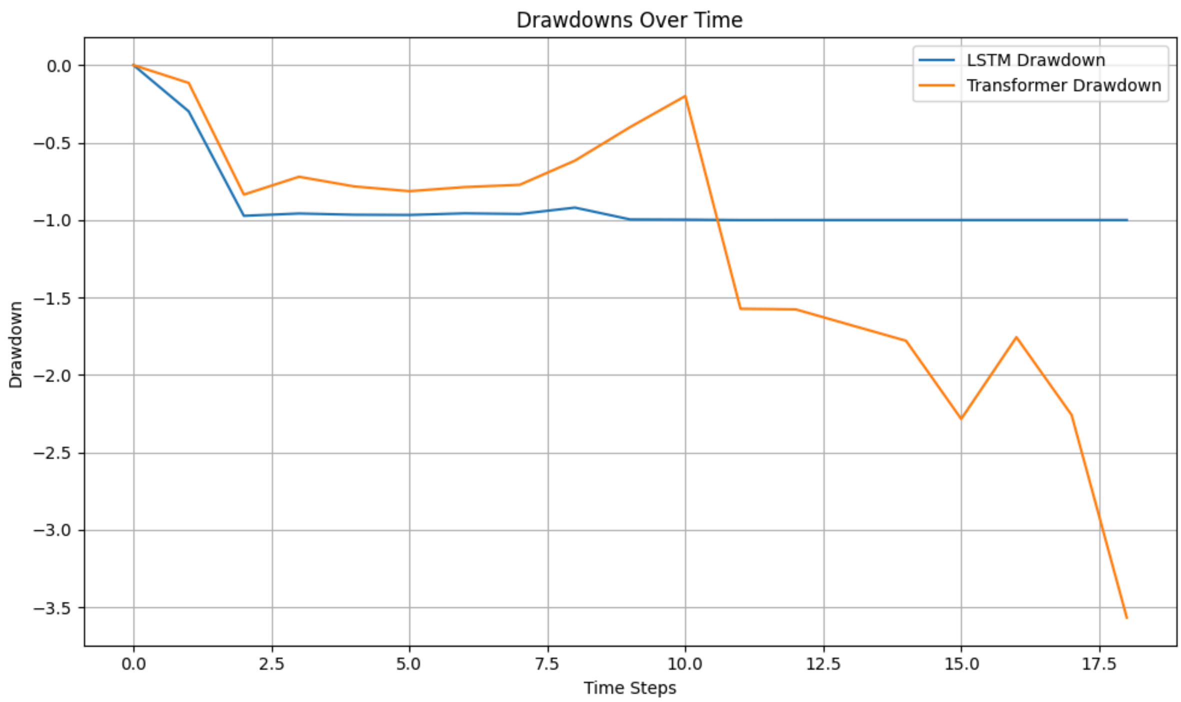
Observation:

- The LSTM portfolio stays relatively stable after initial fluctuations.
- The Transformer portfolio initially performs better but then declines significantly after step 10.

Performance Metrics:

1. **LSTM Portfolio**
   - Annualized Return: **-36.9151**
   - Annualized Volatility: **9.3867**
   - Sharpe Ratio: **-3.9327**
   - Max Drawdown: **-1.0004**
2. **Transformer Portfolio**
   - Annualized Return: **21.9511**
   - Annualized Volatility: **9.7281**
   - Sharpe Ratio: **2.2565**
   - Max Drawdown: **-3.5676**

2. Drawdowns Over Time



- The **blue line** shows the drawdowns for the LSTM portfolio.
- The **orange line** shows the drawdowns for the Transformer portfolio.
- Drawdown indicates how much the portfolio fell from its peak at each time step.

Observation:

- LSTM drawdowns remain controlled (around -1.0).
- Transformer drawdowns increase heavily after step 10, indicating higher risk despite higher returns.