Bitcoin Price Prediction using LSTM and Transformer

Overview

This project implements a Bitcoin price prediction model using a **Hybrid LSTM-Transformer** approach. Initially, the model used a **90-day time window** for training and prediction. Later, improvements were made by integrating **Bitcoin's 4-year halving cycle** and **expanding the time regime to 180 days**, leading to better forecasting accuracy.

Features

- Uses **LSTM + Transformer** to capture both sequential and global dependencies.
- Integrates **Bitcoin's 4-year cycle** as an additional feature.
- Adjusts the **time window from 90 to 180 days**, improving prediction performance.
- Evaluates performance using **MAPE and RMSE**.
- Provides visualizations for **training loss, batch size effects, and actual vs predicted prices**.

Setup Instructions

1☐ Install Dependencies

Ensure you have the required libraries installed:

```
pip install yfinance numpy pandas torch matplotlib scikit-learn
```

2☐ Run the Jupyter Notebook

Launch Jupyter Notebook and open:

```
jupyter notebook Bitcoin_price_predict_using_LSTM_and_Transformer.ipynb
```

3☐ Train and Evaluate

- Execute the notebook cells to **train the model** and **visualize results**.
- Evaluate prediction accuracy using **MAPE and RMSE**.

Results

Baseline Model (90-day window, no cycle feature)

- Predictions had **higher error** due to missing cycle information.
- **MAPE and RMSE were suboptimal**.

Improved Model (180-day window + 4-year cycle)

- The model better captured **BTC price trends over longer periods**.

Contribution & Future Work

- Explore **other deep learning architectures** (e.g., Transformer-only models).
- Experiment with **additional macroeconomic indicators**.
- Optimize hyperparameters for **better generalization**.

Author

Yang An