Capstone Project Proposal

Rajgowthaman Rajendran

Project Title: Predicting Bitcoin Price Movements and Volatility Using Transformer-Based Deep Learning Models.

Project Description:

Bitcoin's price is highly volatile, making accurate short-term price and volatility predictions crucial for traders and investors. This project aims to develop a Transformer-based deep learning model to forecast Bitcoin's short-term price movements and volatility based on historical price data. Unlike traditional methods, Transformers efficiently capture long-range dependencies in time-series data, making them well-suited for this task.

The model will leverage self-attention mechanisms to identify complex temporal patterns in Bitcoin price fluctuations. It will predict short-term price trends (e.g., next-minute, next-hour, next-day) and volatility levels to help traders make informed decisions. Key features include opening, closing, highest, and lowest prices, volume data, and volatility indicators to enhance forecasting accuracy.

Why is this project good?

- **Real-world application:** Bitcoin price prediction is a real-world problem with high relevance for financial markets and trading algorithms.
- **Impactful:** Accurate price forecasting can assist investors and traders in making informed decisions.
- **What makes Transformers stand out** is their self-attention mechanism, which allows them to efficiently process long-range dependencies in data**.** Transformers of deep learning provide a powerful alternative to study time-series data, and this project will provide practical experience with them.
- **Scalability:** The model can be expanded to predict prices and volatility for other cryptocurrencies or even other financial instruments.

How will I do it?

Data Preprocessing: Clean and preprocess the dataset, handle missing values, and scale the features (e.g., normalization of price data).

Feature Engineering: Create new features, such as moving averages, relative strength index (RSI), and other technical indicators, to improve predictions.

Model Development: Use Transformer models (e.g., Time Series Transformer or Informer) to capture long-range dependencies.

Training & Optimization: Train the model in PyTorch using mean squared error (MSE) loss and optimize using Adam optimizer.

Evaluation & Fine-Tuning: Evaluate the model using metrics like RMSE (Root Mean Squared Error), accuracy, and mean absolute error (MAE), and directional accuracy; fine-tune hyperparameters.

Visualization & Deployment: Build a simple dashboard to visualize predicted vs. actual Bitcoin prices over time and evaluate the model's performance.

Dataset:

Bitcoin Historical Data - https://www.kaggle.com/datasets/mczielinski/bitcoin-historical-data

This dataset contains historical data for Bitcoin, including the opening, closing, high, low prices, and trading volume for each day from 2012 to the present. It provides ample data for training time-series models and making future price predictions.

Rows: 6944800

Column: 7

Dataset Structure:

The dataset consists of the following columns:

1. **Timestamp (Unix Epoch)**
    - This column represents the **Unix timestamp** in seconds (since January 1, 1970).
    - It can be converted into a human-readable **date and time** format.
2. **Open**
    - The opening price of **Bitcoin (BTC)** in USD for the given minute.
3. **High**
    - The highest price of **Bitcoin (BTC)** in USD during that minute.
4. **Low**
    - The lowest price of **Bitcoin (BTC)** in USD during that minute.
5. **Close**
    - The closing price of **Bitcoin (BTC)** in USD for that minute.
6. **Volume**
    - The volume of **Bitcoin** traded in that minute, measured in **BTC**.
7. **Datetime**
    - A human-readable version of the **Timestamp** in **UTC** time zone, showing the date and time in **YYYY-MM-DD HH:MM:SS+00:00** format.

Evaluation of system performance:

- Root Mean Squared Error (RMSE): To assess prediction accuracy for continuous price predictions.
- Mean Absolute Error (MAE): To evaluate the model’s prediction quality and match with actual values.
- Directional Accuracy: For classification tasks predicting whether the price will rise or fall.
- Visualization: Plot actual vs. predicted prices and trends over time for a clear comparison of model performance.