

Project: AI & Cryptocurrency Analytics

Rania HENTATI KAFFEL ,
Sarra Ben YAHIA,
José Ángel GARCÍA SÁNCHEZ

November 2024

Objective

The goal of this project is to leverage AI techniques to analyze cryptocurrency market trends, assess risks, and predict price movements. This involves the use of blockchain data, social media sentiment, and historical price trends.

Methodology

1. Data Collection

Several methods can be used to gather data, with web scraping being the minimum requirement. Additional approaches, such as API calls or downloading data from the internet, can also be incorporated to complement the process.

- **Blockchain Data:** Gather transaction volumes, wallet activity, and on-chain metrics (e.g., miner rewards, gas fees).
- **Market Data:** Collect historical price data, trading volumes, and volatility metrics from APIs like Binance, CoinGecko, or Alpha Vantage.
- **Social Media Sentiment:** Scrape data from platforms like Twitter, Reddit, and crypto-specific forums to gauge sentiment.
- **Macro Events:** Include news headlines and regulatory updates impacting cryptocurrency markets.

Data Sources

- **Blockchain APIs:** Etherscan, Glassnode, Chainalysis.
- **Market Data:** Binance, CoinMarketCap, CryptoCompare.
- **Social Media:** Twitter, Reddit, web scraping tools for forums.
- **News and Events:** Google News, CryptoPanic.

2. Feature Engineering

- Create on-chain metrics such as active wallets, token circulation, and transaction velocity.
- Compute technical indicators like moving averages, RSI (Relative Strength Index), and Bollinger Bands.
- Use NLP techniques to extract sentiment scores from social media and news data.

3. Machine Learning Model Development

- Train machine learning models to predict price movements or volatility. This could include:
 - **LSTM (Long Short-Term Memory)** networks for time series forecasting.
 - Experiment with ensemble methods like **Random Forests** or **XGBoost**.
 - Fine-tune **Transformer models** (e.g., BERT) for sentiment analysis.
- Evaluate model performance using relevant metrics.

4. Portfolio Optimization

- Incorporate predicted price trends and volatility to optimize cryptocurrency portfolios.
- Compare various allocation strategies:
 - Equal weighting.
 - Maximum Sharpe ratio allocation.
 - Risk-parity allocation.

FastAPI API (Backend)

Your project should include a FastAPI-based API that allows users to interact with various components of your work. This API will follow the standard API conventions facilitating usage and deployment for performing tasks like launching web scraping, making predictions with machine learning models, or processing data. Below are the essential endpoints that must be implemented, along with guidelines for their functionality.

Required Endpoints

1. Index Endpoint

- Purpose: Brief explanation of the application's purpose and capabilities. Provide API documentation link and available endpoints

2. Health Check Endpoint

- Purpose: Monitor API availability and system status

- Expected Response: Service health metrics and status

3. Web Scraping Endpoint

- Purpose: Execute web scraping tasks based on provided parameters
- Webscraped Data

4. Model Inference Endpoint

- Purpose: Execute machine learning and LLM model predictions
- Some possible features:
 - Implement pipelines
 - Support for different model types
 - Input validation and preprocessing
 - Batch prediction capabilities

Note: *Additional endpoints should be implemented as needed for specific workflows. For example, when using multiple models in sequence (such as an LLM generating features for an ML classifier), separate endpoints for each processing step will improve modularity and testing capabilities.*

Livrables

- **Notebook**

- Deliver a well-documented and structured notebook.
- Ensure maximum visual representation (graphs, charts, visualizations) with all results clearly explained and commented.

- **Research Paper (Overleaf):**

- Share a paper written on Overleaf, granting us *Reader Access*.
- Ensure it is well-structured with the following sections:
 1. **Methodology:** Explain the steps and processes used.
 2. **Data Collection:** Detail the sources and preparation of the data.
 3. **Results and Interpretation:** Present results with commentary and provide thorough interpretations.

- **GitHub Repository:**

- Provide a GitHub repository containing:
 - * The full codebase with proper documentation and organization.
 - * Example datasets or a clear reference to the data used.
 - * Instructions for running the code, including dependencies and setup steps.
 - * Respect of the good coding practices. (PEP8)
 - * Any additional resources, such as scripts for visualizations or report generation.