GDA Hackathon Data Science and Machine Learning(21 Dec - 24 Dec)
Team Alpha Go

Table of Contents

Executive Summary
The Data Science Hackathon is aimed to provide participants with exposure to the open source L3 Atom platform. The six questions provided a wide scope of the implementation of data science principles in the crypto trading space, while simultaneously pulling back the curtain on the poor architecture and questionable practices of current major exchanges such as ByBit. The lack of valuable data and possible manipulation in an unregulated industry is a serious issue. The first three tasks explore what GDA engineers are doing to solve such a pivotal problem in the emerging industry and what we can do about it.

Problem Statement
Crypto exchanges provide low quality data with not enough detail/granularity due to the lack of regulations currently placed on the crypto market. This allows the CEX to manipulate data impacting crypto traders who rely on the data provided.

Defining the tasks

Constraints

Task 1: Creating a stable connection to ByBit Public Limit Order and Public Trade Messages WebSocket endpoint and normalizing the data in memory

Challenges
- Price Jumps
- Pybit Package -Websocket not always giving snapshot on startup
- Matching the trades while the limit order data is streaming in parallel

Methodology

Solution
- Multi-thread

Task 2: Using the normalized data, build a limit order book in memory. Measure the average time to update the limit order book over 5 seconds.

Challenges

- Order ID matching the price
- Determining optimal datastructure for storing the order book
  Having a well defined interface to integrate different components/modules easier

  Methodology
  Tried 2 different data structure. Dictionary(Hash table) first, numpy array implementation second. No significant differents
  Solution
  Attempted granularity

Task 3: Calculate as many metrics from our list & OFI & TFI for every 100ms pulse, use them to predict market snooping, measure the average time taken to calculate.

  Challenges

- Matching the trading data
- Finding optimal way to calculate metrics from the limit order book and the trade data data structures

  Methodology
- Spoofing and Metric calculations

  Solution

Task 4: Build a data pipeline using AirFlow to Store and retrieve historical ByBit data on timescale DB. Also, do a comparative case study on Kafka Vs Pulsar.

Task 5: Setup Kafka Cluster and Pulsar Cluster and publish the ByBit data on topics as asset pairs.

https://bybit-exchange.github.io/docs/inverse/?python--pybit#t-setrisklimit