Parallel Financial Prediction and Strategies

CS205 Final Project

Junkai Ong, Sivananda Rajananda, Kevin Hare

Problem Statement

- Financial time series data is generated at very high frequencies
  - E.g. 5.3 billion shares of the S&P 500 are traded per day

- Many financial trading strategies rely on short-term prediction and decision-making
  - E.g. Momentum investing, a short-term strategy where a stock is purchased at any price and sold once a price drop is anticipated
  - Requires low latency and high compute

Parallel Computation and Processing

- Goal 1: Train/Deploy Expressive, Computationally Expensive Models
  - Long Short Term Memory (LSTM) Models
    - State-of-the-art deep learning sequential model suitable for time series like stock prices
    - Computationally expensive to train and to predict
  - *Reach Goal*: Implement transformers or transformer encoder with downstream regression

- Goal 2: Achieve Near Real-Time Stock Prediction
  - Intervals for prediction data can be discretized at a fairly high resolution
  - Once trained, models must be able to fetch data and make prediction in near real time (5 minutes ahead of predicted event)
  - When deployed, it could integrate with automated trading platform

Model & Data

- LSTM model used for stock prediction
  - LSTM is a type of Recurrent Neural Network (RNN) that is well suited for sequence prediction problems
- Data will be requested through an API and streamed to the models
  - Using yfinance API for the large historical data - 1-minute resolution
  - Using IEX Cloud’s API for real-time stock information
    - For prediction and real-time updates to the model
- The data to train, develop and test our models comprises of two aspects
  - Prices and volumes of a particular stock in the S&P 500
    - Key features: ‘Adjustment close price’, ‘trading volume’
  - Accounting and Corporate Statistics of stock in the S&P 500
    - Key features: ‘Debt-to-Equity Ratio’ , ‘Return on Equity’, ‘Price-to-Book Ratio’, ‘Profit Margin’, ‘Diluted Earnings Per Share’, ‘Company Beta’

Programming Models, Tools and Infrastructures

- Big Data Processing:
    - Spark - DataFrame: Process historical data, approx. 20GB
    - Spark Streaming: Processing data in real time for allocation to nodes and updates of multiple models
- Big Compute:
    - Horovod: Distributed deep learning platform created by Uber; facilitates training across many GPU cores on single node and MPI scaling
- Infrastructure:
    - AWS
        - S3 for data storage
        - EC2 (GPU instance) for model training, prediction