Chris Darley

12/11/2024

Capstone Project Summary

My capstone project is based around developing a trading strategy for the Solana cryptocurrency market. My goal is to be able to monitor the market in near real time (10 minute intervals) and try to identify trends that would be informative for a trading strategy. My project focuses on ‘shitcoins’ on the Solana blockchain which start out on the pump.fun market <https://pump.fun/board>. Around 1% of these will graduate to the Raydium Dex (decentralized exchange) after reaching around $100,000 in trade volume. These coins are basically memes, they regularly reach their all-time high within a week or shorter, virtually none of them have any actual utility, yet the market sees a substantial amount of trading volume (~4 billion dollars in the last 24 hours). <https://dexscreener.com/solana> is a dashboard of the currently popular coins, many of which have increased in price 100x or 1000x over the course of a day. To narrow the focus, my project starts tracking coins after they graduate from pump.fun to Raydium.

The idea behind my capstone project is that if there is any way to find a somewhat reliable trading signal in this market data, there would be a large potential for profitable trading. The goal of my project is not to make a trading strategy from the outset, but more to observe the market by capturing data, making some analysis, and then evaluating if anything I learned could be used in a profitable trading strategy. My project currently tracks the price, liquidity, and market cap of all coins that have entered the Raydium market within the past 7 days. After collecting data for a couple weeks, I will be able to see the distribution of how long coins usually take to reach their all-time high, and if this timeframe is different for the more successful coins. This price data will also enable me to do lagged price change percentages for the last 10 minutes, 30 minutes, 1 hour, 6 hours, and 24 hours, and I can sort these to find the coins which are currently gaining or losing the most over these time frames in near real time.

The second stage of my project which I have not yet implemented is to monitor the activity of other traders buying and selling these tokens. There are many online dashboards which people use to track the price action of tokens, but public tools to track the activity of different wallets is a bit less common. One goal would be to see if certain wallets consistently perform well and may be worth copying. I would also be evaluating statistics like number of wallets holding each token, change in number of holders over time, and distribution of holders (Is the token held by many wallets with smaller positions, or do fewer wallets hold a larger portion of the supply?)

The main technical difficulties I faced during this project were surrounding access to data and scale of data, especially in regard to monitoring wallet activity. The way Solana data works is that there are different degrees of data you can get access to. For my project, the ‘primary source’ would be subscribing to the WebSocket logs of the Raydium market transactions directly from the publicly available Solana node API. I looked at this, and there was no easy way I found to easily translate the logs into transactions, because my knowledge of blockchain events is way too low. The second degree down from that would be to subscribe to an API provider that parses the Solana blockchain logs in real time and sends data about transactions through a WebSocket API. This was out of my range because many of these start around $200 per month, ranging into the thousands, I do not have experience handling streaming WebSocket data, and the scale of the data would likely be far too large for me to handle. The third option which I intend to implement in some way later on is to get data about ownership through an http API. This would enable me to track the ownership of different tokens at ten minute intervals, and I can see which wallets are buying and selling each token by capturing change in ownership over time.

My project uses 2 different endpoints from the solanatracker.io API to track tokens graduating from pump.fun to Raydium, and to track the price of these tokens for 7 days after graduation. Another service called quicknode lets people perform functions on a remote Solana node. Using this service, I have tested out code using quicknode which gets the amount of token owned and the wallet address for every owner of a given token. This code can be executed for all tokens, so I would essentially be able to capture a complete picture of ownership change for all of the tokens. However, this volume of data is way too large, and the ownership table alone would zoom past 10 million rows if I was just to track this data for coins that are less than 24 hours old. The solanatracker.io API has various other endpoints which I plan to test out, and hopefully there is one that I will be able to incorporate into my project. The **GET /tokens/{tokenAddress}/holders** endpoint returns the top 100 holders of a token, the amount owned, and what percentage of total ownership each holder makes up. Queried over time, this endpoint would allow me to track wallets that are top holders across multiple tokens, and it would allow me to measure how ‘top-heavy’ the ownership of a given token is. Another idea is to use the **GET /top-traders/{token}** endpoint after 7 days of trading for each token. This would enable me to build a ranking of wallets that have the highest profit/loss ratio for each token. Using this historical ranking, I could use the **GET /wallet/{owner}/trades** endpoint to track and possibly copy trades being executed by these wallets. I could also add a metric to the real time tracking which would be proportion of top historical wallets which are currently holding a given token. One of the main reasons that I am tracking ‘time\_to\_alltime\_high’ is so that I can get a sense of how long I should monitor the ownership of coins for after they graduate to Raydium, because doing it for a full week would likely be too expensive.

Overall, I am very happy about this project, and I intend to continue working on it. In addition to incorporating ownership data, I think there are many areas where I could work to improve what I have built. I actually have not traded in this market myself, and I don’t have that much experience with crypto other than buying a few NFTs a couple years ago. I got most of the background information about these markets from my younger brother who has traded some of these coins in the past. He told me many things to watch out for, such as ‘syndicates’ of people (or few people with many wallets) who will buy up a token and then sell it simultaneously to lock in their profits, so I could track when this happens and then create a blacklist of offending wallet addresses. There are also certain coins which skyrocket in price because they allow people to buy but don’t allow anyone to sell, so I could create some code which excludes these tokens from my models. I could also extend the project to monitor coins in pump.fun, and I could try to monitor messages in certain telegram channels where most of these coins originally gain traction. Finally, there are some aspects of the project that I didn’t have time to get to, including adding some sort of front end and potentially converting my standalone python functions used to load data to source tables into DBT files that would be better documented, possibly using UDFs. I didn’t get to producing a visualization, but 2 sql files are included in the capstone\_submission\_files folder which would serve as the basis for different visualizations. Also, I didn’t get around to changing or deleting many of the jaffle\_shop models and configurations from the template for the project because I was not sure about git implications. I got confirmation from Mitali that I can delete and change these files, so my project still has some unnecessary files and configuration names that need to be changed which I will not have time to do safely before submitting.