Exploring Non-Fungible Tokens via Twitter

Juliana McCausland

The goal of this project was to explore twitter data relating to non-fungible tokens (NFTs) and present it in a way that could be interesting for an NFT newcomer or enthusiast. I was hoping to find actionable insights for potential investors, but the NLP analysis was lackluster. I ended up making an autoencoder that attempts to generate NFTs based on an input image, and I also made an NFT recommendation engine that recommends NFT artwork based on an input image using euclidean distances.

Design

Recently, and especially over the last year, NFTs, have exploded in popularity, particularly in the art and finance communities. Artists have been taking advantage of the hype, and selling basic digital art for hundreds of thousands, and sometimes even millions of dollars. Even notable auction houses like Christie's are beginning to sell digital artworks. It goes without saying that this is a lucrative market, and a lot of people are interested in figuring out how to profit in some way. This project was intended to explore NFT-related content on twitter to make an interesting web application that utilizes the information presented on twitter in ways that may be interesting or entertaining to users

Data

I scraped my data from Twitter using Twint. I scraped over 500,000 tweets using various keywords, primarily related to different NFT collections (e.g. CryptoPunks, FameLady Squad, SupDucks, etc). The data was split up based on keywords, and later various subsets of the data were combined. I ended up focusing primarily on image data by the end of the project to create the autoencoder and the NFT recommendation engine.

Algorithms

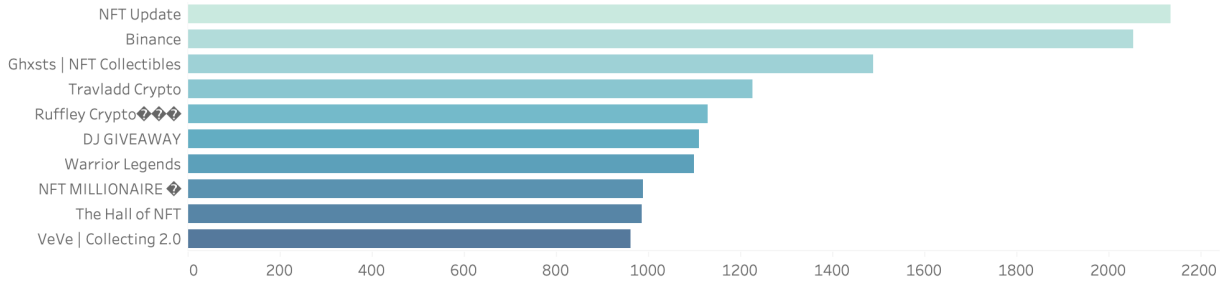
I explored the data in both pandas and mongodb. Mongodb was especially useful for downloading images from collections within my database. Mongodb was much easier than pandas considering the scale of the data. I performed machine learning when creating my autoencoder, tweaking my hyperparameters and observing loss values. I also used euclidean distances to find similarities between images.

I did some NLP on the data. I mostly performed NMF topic modeling and created word clouds.

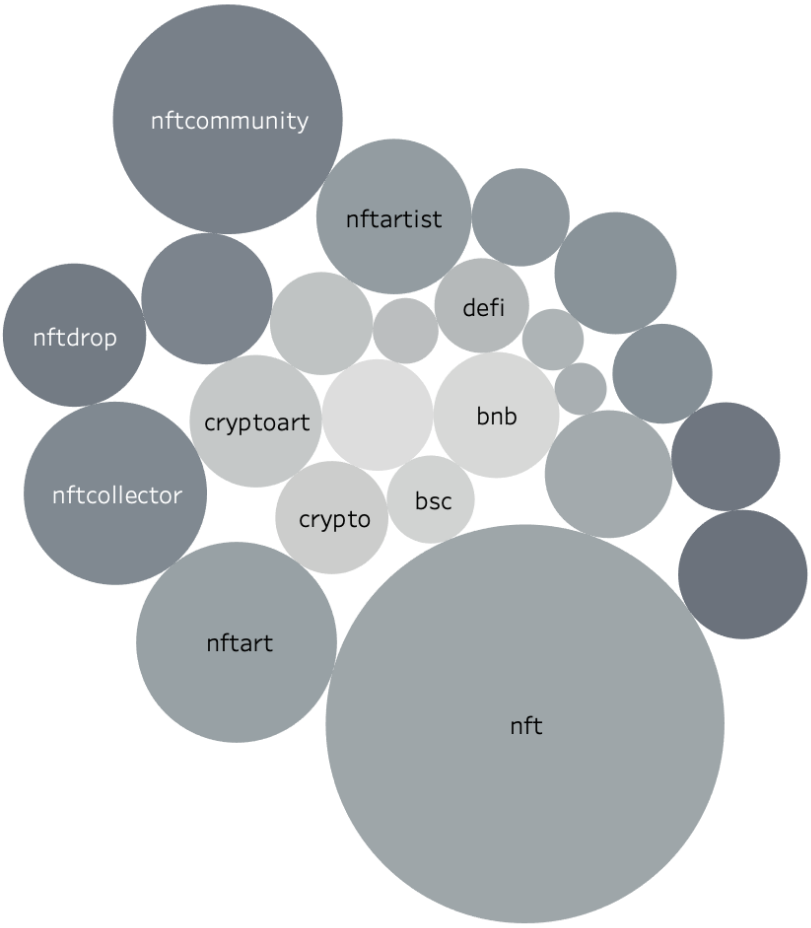
Tools

Mongodb, Spyder (for python code), Streamlit, Tableau, pandas, keras, jupyter notebook, Twint. I attempted to use Google Cloud, but after a couple of days of debugging because of importing issues, I gave up.

Communication



Hashtags that get the most retweets



Hashtags

- binancesmartchain
- bnb
- bsc
- crypto
- cryptoart
- cryptodogs
- cryptonews
- defi
- doge
- dogearmy
- giveaway
- nft
- nftart
- nftartist
- nftartists
- nftcollectibles
- nftcollection
- nftcollector
- nftcollectors
- nftcommunity
- nftdrop
- nftgaming
- nftgiveaway