February, 2024

IA101

Algorithmic Information & Artificial Intelligence

Micro-study

teaching.simplicitytheory.science

Names : Duhazé Titouan, Rouillé Nathan

Implementation of compression distance to predict the price of NFTs

GitHub

Link: https://github.com/titiuo/Normalized-Compression-Distances-for-NFT/tree/main

Abstract

Given a collection of NFTs listed on a marketplace, the goal is to develop an algorithm capable to predict the price of any NFT of this collection based on the price of the other and their compression distance to our NFT.

Problem

How to predict the price of a NFT of a given collection with minimal relative error?

Method

To compute our compression distance, we first used NCD to get the visual similarities between 2 NFTs with their data compression.

We had a 2nd approach by using the NID with a link between Kolmogorov complexity and frequency of NFTs i.e. the attributes apparition probability of our NFT.

After getting the distances between NFTs, we find different method to predict the price of a NFT:

- Search the closer one in the collection and attribute its price.

- Search the x closers one and attributes the minimum price.
- Compute an average price by using the distances for the coefficient.
- Compute an average price by using f(distances) for the coefficient with f a function well chosen to increase the weight of NFTs closer and decrease the weight of NFTs farthest.
- Compute a neural network which takes in entry the prices and the distances of all the NFT of the collection between the one we want to predict and predict the prices.

Results

The first approach gives us good results but in a very long time (1 second to compute the NCD between 2 pictures, so approximately 10min to compare all the NFTs of a thousand NFT's collection). An other problem is the fact that $Z(x,x) \neq Z(x)$ (with the library lzma) what should not be the cased.

The second approach was much better in terms of rapidity (1min maximum to compare all the NFTs of a collection) but also for the result of the distances.

The results of the price predicted for each method with and without a price filter to use only relevant NFTs, here we arbitrary chose under 11 Sol which represent 95.2% of the total listed supply:

- The closer one approach gives us a **Δ_{mean} = 3.2 Sol** meaning a relative error of 61.2% because we took the offers prices which can be extremely high and far from the reality. By taking just the NFTs with a price < 11 Sol we obtained a **Δ_{mean} = 2.1 Sol** meaning a relative error of 50.5%

- The x closers one approach gives us a **Δ_{mean} = 2.7 Sol** meaning a relative error of 51.8% with x=3 which is better than the precedent but still high because we underestimate some prices by taking the minimum price between the x closers. By taking just the NFTs with a price < 11 Sol we obtained a **Δ_{mean} = 1.26 Sol** meaning a relative error of 30.3%

- The average price approach gives us a **Δ_{mean} = 2.66 Sol** meaning a relative error of 50.9% which is close from the precedent. By taking just the NFTs with a price < 11 Sol we obtained a **Δ_{mean} = 1.71 Sol** meaning a relative error of 41.2% which is this time, very less good than the precedent.

- The average price with a function approach gives us a **Δ_{mean} = 2.64 Sol** meaning a relative error of 50.4% which is similar than the average price approach. By taking just the NFTs with a price < 11 Sol we obtained a **Δ_{mean} = 1.7 Sol** meaning a relative error of 40.9%.

- The neural network approach gives us a **Δ_{mean} = 2.2 - 2.6 Sol** meaning a relative error of 57.8% which is quite good but to improve it we need more data and especially more coherent. By taking just the NFTs with a price < 11 Sol we obtained a **Δ_{mean} = 1.39 Sol** meaning a relative error of 33.7%.

Discussion

We had those results with one collection which had just 280 NFTs. The fact that we obtained those results with a lack of data could be a problem. However, now that we have the code which can be used on all the collection, we can remake our analysis of result on several collections.

Bibliography

- GitHub for NCD: https://github.com/alephmelo/pyncd
- Explanation of NCD for pictures: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7512663/
- API for magic Eden: https://docs.magiceden.io/reference/solana-overview
- Collection of NFTs named Sandbar: https://magiceden.io/marketplace/sandbar