NFT Price Prediction based on various Neural Network

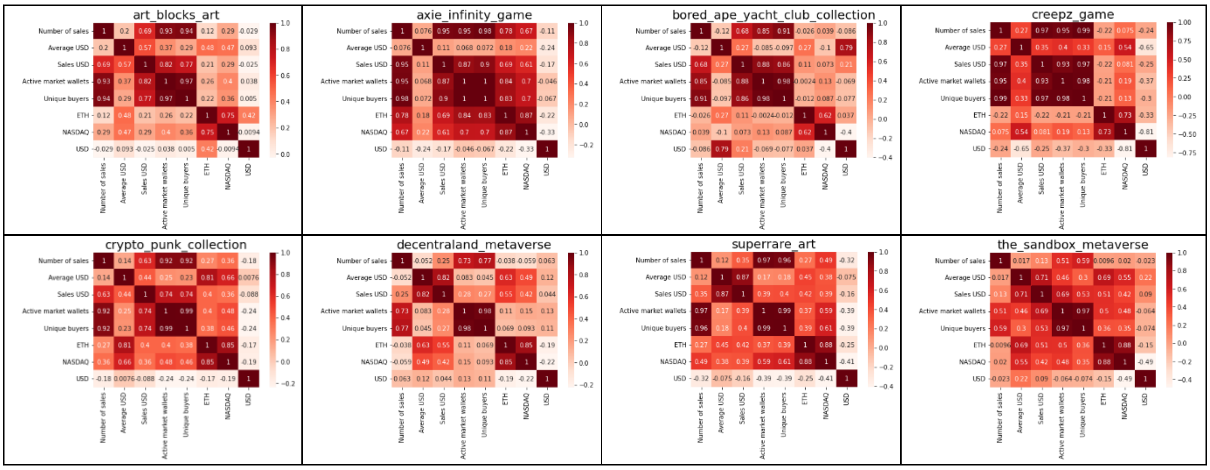
20170435 윤지언 20170653 차정엽 20170783 이동규 20190584 정서경

1. Motivation

Various resources related to blockchain are emerging nowadays, and among them our team chose to focus on the “NFT”. The difference between NFTs and other cryptocurrencies is that subjectivity influences NFT. Since NFTs have unique differences from other digital tokens, our team aims to find the best algorithm for predicting NFT values, and eventually come out with an application result of applying the representative algorithms for stock and bitcoin price prediction on NFTs.

2. Data

We picked 2 projects each from art, game, collection, metaverse. We considered the market cap and popularity. For each project, we used number of sales, sales amount in USD, average USD of NFT, number of active market wallets, and number of unique buyers on daily and hourly data. In the macroeconomic perspective, we collected Ethereum price, OHLCV of NASDAQ index, and dollar index data on daily and hourly data. Below are the correlation heatmap visualization between the features.



<Figure 1: Correlation heatmap between features>

3. Method

3.1 Linear Regression

Linear regression is the most simple and traditional method to make prediction in time series data. We made pure linear regression, with external features, and with lasso regularization model for the baseline.

3.2 Stacked LSTM

Since traditional RNN models have difficulties in learning long term dependencies, we used stacked LSTM to learn important part of the sequences. In general, a single LSTM model consists of three gates: forget, input and output gates. In the stacked LSTM model, a single LSTM cell in t-1 time step gives the hidden unit h_t-1 not only to next cell, but also to the next LSTM layer’s t-1 LSTM cell. By giving information to the next layer directly, entire model could have much accurate.

3.3 Reinforced Learning

3.3.1 Agent & Observation space

The agent keeps track of the most recently observed target value to update its current location, and learns to take an action by going up or down in some value, making a prediction of the price in the next timestep. We used the basic NFT data features as the observations.

3.3.2 Policy network & Algorithms

We used an MLP Policy network that has 64 inputs to a feature extractor, to a 64 input Fully Connected layer that outputs a final action. We used actor critic RL algorithms that could support continuous action spaces, A2C and PPO.

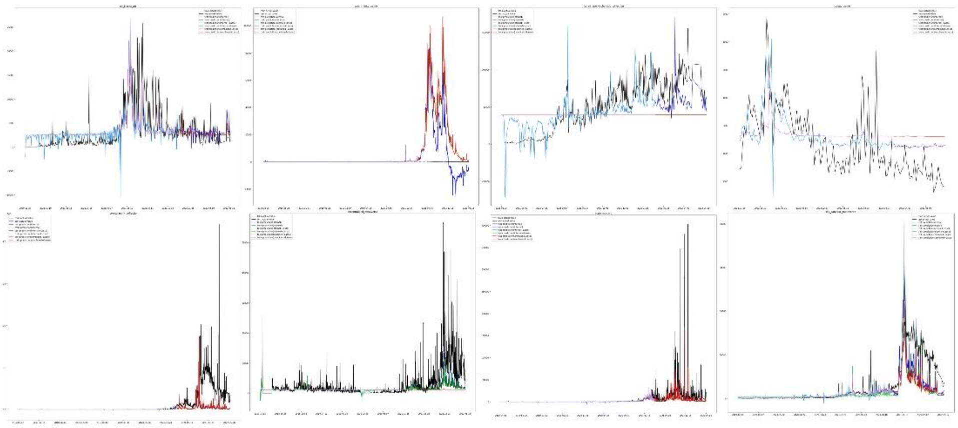
4. Experiments

4.1 Linear regression

4.1.1 Prediction method

We first used only internal data (Number of sales, Sales amount of USD, Average USD of collection) for the linear regression. After that, we added lasso regularization term. Finally, we combined the external data (Ethereum price, NASDAQ index, Dollar index) with lasso regularization. Below is the result of each experiment.

4.1.2 Result and Analysis



<Figure 1: Prediction results of linear regression model>

	Internal		Internal + lasso		Combined + lasso		Lowest RMSE
	RMSE	R^2	RMSE	R^2	RMSE	R^2	
Art blocks	2920.7	-1.4549	1728.8	0.1398	1729.2	0.1394	1728.8
Axie infinity	866012.9	-76423280.1822	36690.4	-137176.4424	38420.1	-150780.8623	36690.4
Bored ape yacht club	87804.9	-18.2843	50756.2	-5.4438	50756.2	-5.4438	50756.2
Creepz	1251.5	-9.7342	1345.3	-11.4036	1345.3	-5.443885	1251.5
Crypto punk	296089.3	-1.3429	269438.9	-0.9401	267900.5	-0.9173	296089.3
Decentraland	10890.4	-1.0595	8502.9	-0.2555	10113.5	-0.7743	8502.9
Superrare	44565.4	0.2672	31868.4	0.3519	32090.0	0.3429	32090.0
The sandbox	10230	-4.3069	6734.4	-1.2997	6776.4	-1.3285	6734.4

<Figure 2: Results table of linear regression model>

Although we tried to improve linear regression model with various methods, results are not meaningful. Except Art blocks and Superrare, coefficient of determination of each collection model show negative value with big RMSE. It means model is worse than model that predicts all target variable as mean. This is because extreme high correlation between features. Figure1 Heatmap graph of correlation between features shows many high correlations with dark red color. Furthermore, nonlinearity between target variable and features can be another reason because automated feature selection with Lasso regression cannot give better result.

4.2. Stacked LSTM

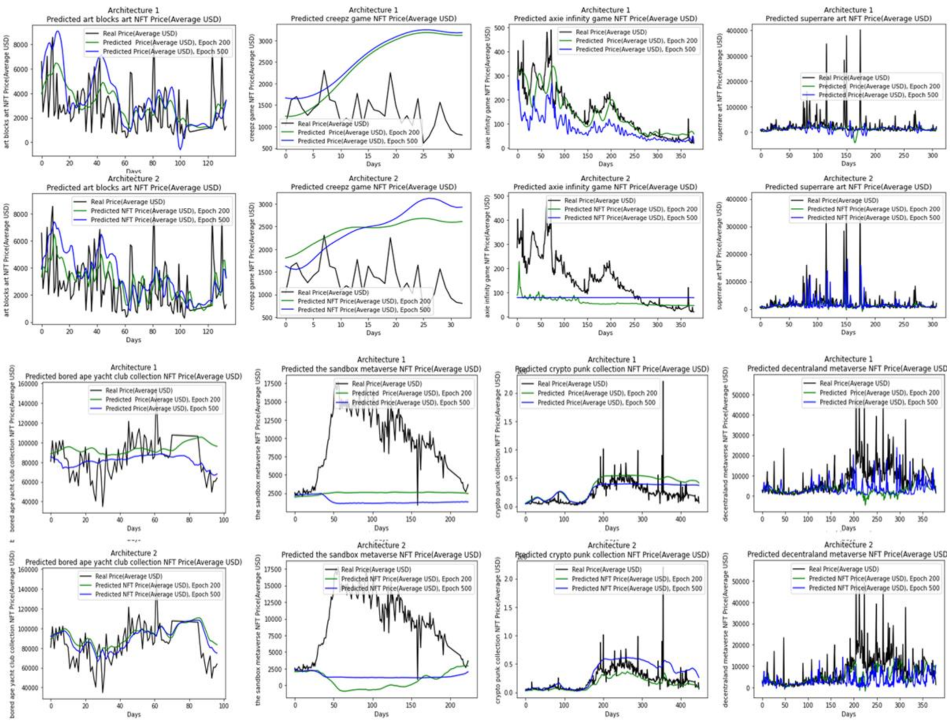
4.2.1 Stacked LSTM architecture

We’ve implemented two keras stacked LSTM. Architecture 1 consists of 4 hidden LSTM layers, most of its output layer size equal to 50. Architecture 2 consists of 3 hidden LSTM layers, most of its output layer size decreases by power of 2, starting from 128.

4.2.2 Training method

Similarly as before, our team used 8 categories on prediction experiment. We used 70% of valid data as training set and 30% of valid data as a test set. Valid data is a result of a data excluding the first successive data which has its element 0. Input was made by appending consecutive sequences, where each sequence contains 60 consecutive daily average USD price. Each architecture was trained twice, which different epochs (200, 500).

4.2.3 Result and Analysis



<Figure 4: Prediction results of stacked LSTM model>

Data Category	RMSE				Lowest RMSE
	Architecture 1		Architecture 2		
	Epoch 200	Epoch 500	Epoch 200	Epoch 500	
art blocks art	2035	2589	1879	2249	1879
Creepz game	1464	1521	1206	1355	1206
The sandbox metaverse	8112	9279	10262	9284	8112
Superrare art	43569	45175	44324	48966	43569
Bored ape yacht club collection	21582	18245	18262	16691	16691
Axie infinity game	39	42	40	120	39
Crypto punk collection	204144	163752	151422	191073	151422
Decentraland metaverse	9702	9448	7703	9238	7703

Data Category	RMSE		Lowest RMSE
	Architecture 2		
	Epoch 200	Epoch 500	
art blocks art	1807↓	1839↓	1807
Creepz game	1430↑	873↓	873
The sandbox metaverse	8781↓	8121↓	8121
Superrare art	43515↓	44205↓	43515
Bored ape yacht club collection	17288↓	16615↓	16615
Axie infinity game	99↑	62↓	62
Crypto punk collection	135683↓	151824↓	135683
Decentraland metaverse	8787↑	10377↑	8787

<Figure 5: Results table of stacked LSTM model >

The stacked LSTM prediction graph of NFT's price is on figure 4. The RMSE value between the stacked LSTM prediction and the real value is in figure 5, which left side is original model and right side is after adding regularization to the architecture 2. Architecture 2's RMSE had a greater number of the lowest RMSE than architecture 1, and in architecture 2, result of epoch 200 had a greater number of best RMSE than result of epoch 500. Since this may indicate overfitting, we have done extra experiment using architecture 2 by adding L1 and L2 regularization. After the regularization, it seemed overfitting to be solved. However, our model shows low prediction on outlier value, and this architecture does not consider external data such as ETH price and NASDAQ, further research can be done to use external data to predicted outlier value well.

4.3 Reinforcement Learning

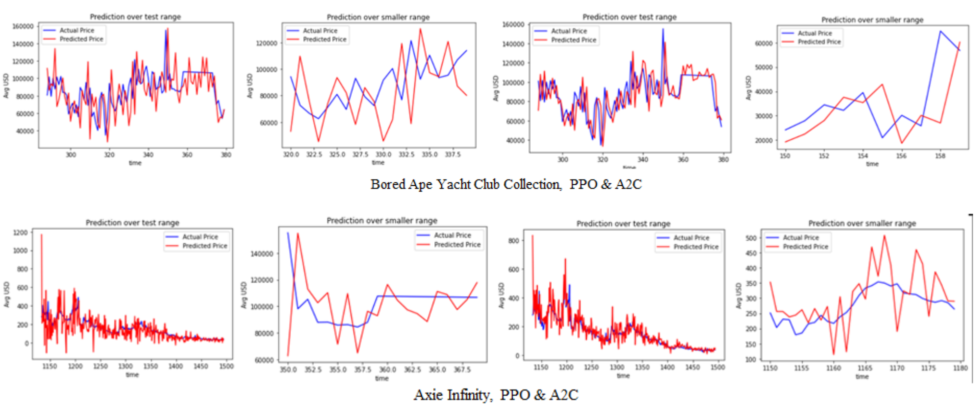
4.3.1 Normalization of action & observation

In order for Actor critic algorithms to learn well, the observation, reward and especially the actions, had to be normalized to reasonable values. We used Tanh as the activation function in the Policy Network to normalize the action space to [-1,1]. Normalizing the observation was not feasible as the agent would be able to predict values only in the normalized environment, thus not able to predict the actual price values. This caused a problem because the action space was limited whereas the observation was not. Thus, we used the action value as a ratio to be multiplied to the agent’s current location to make a prediction. The ratio is defined as such: 1 + (0.5*action), giving the final ratio a range of [0.5, 1.5].

4.3.2 Normalization of the reward

We used the absolute difference as error and its negative value as reward, then normalized it to [-1,0] using moving averages and variances, because we did not want to give positive rewards in order to get the agent to predict as close to the actual value possible.

4.3.3 Result and Analysis



<Figure 6: Prediction results of reinforcement learning model>

	A2C		PPO		Lowest RMSE
	RMSE	R^2	RMSE	R^2	
Bored ape yacht club	19662.73	0.11	24523.273	-0.525	19662.73
Axie infinity	60.950	0.620	100.979	-0.041	60.950

<Figure 7: Results table of linear regression model>

We can see that the A2C performs much better than PPO in both cases. The models at times seem to make bigger changes than necessary, and this can be alleviated by adjusting the factor multiplied to the action value or longer training.

5. Conclusion

	Internal	Internal + lasso	Combined + lasso	LSTM1 Epoch 200	LSTM1 Epoch 500	LSTM2 Epoch 200	LSTM2 Epoch 500	LSTM2 (Regularized) Epoch 200	LSTM2 (Regularized) Epoch 500	A2C	PPO
Art blocks	2920.7	1728.8	1729.2	2035	2589	1879	2249	1807	1839		
Axie infinity	866012.9	36690.4	38420.1	39	42	40	120	99	62	60.950	100.979
Bored ape yacht club	87804.9	50756.2	50756.2	21582	18245	18262	16691	17288	16615	19662.73	24523.273
Creepz	1251.5	1345.3	1345.3	1464	1521	1206	1355	1430	873		
Crypto punk	296089.3	269438.9	267900.5	204144	163752	151422	191073	135683	151824		
Decentraland	10890.4	8502.9	10113.5	9702	9448	7703	9238	8787	10377		
Superrare	44565.4	31868.4	32090.0	43569	45175	44324	48966	43515	44205		
The sandbox	10230	6734.4	6776.4	8112	9279	10262	9284	8781	8121		

<Figure 8: Results table of all models>

We developed 11 forecasting models from simple linear regression to deep neural network and reinforcement learning for predicting future average price of specific NFT collectibles. We validate our models on eight different NFT collectibles' datasets with RMSE and R^2 metrics and perform quantitative analysis based on the results. Experimental results show that stacked LSTM based forecasting model outperforms other models in majority of datasets which proves that stacked RNN networks effectively learn multivariate features and predict the best accurate price. We found that increasing model complexity and excessive training to extract more accurate performance does not operated well on our current time-series experiment setting.

- Contributes
  - Donggyu Lee - Data collection, Data preprocessing
  - Jieon Yoon - Linear Regression
  - Seogyeong Jeong - Stacked LSTM
  - Jeongyoub Cha - Reinforced Learning

- References

NFT data - https://nonfungible.com/market-tracker

Ethereum price data -https://cointelegraph.com/ethereum-price

NASDAQ data - https://kr.investing.com/indices/nasdaq-composite-historical-data

Dollar index data - https://kr.investing.com/indices/usdollar

Kai Chen, Yi, Zhou, Fangyan Dai. "A LSTM-based method for stock returns prediction :A case study of China stock market"(2015)

David M. Q. Nelson, Adriano C. M. Pereira, Renato A. de Oliveira, “Stock Market’s Price Movement Prediction With LSTM Neural Networks”(2017)

OpenAI Gym - https://www.gymlibrary.ml/

Stable-Baseline 3 - https://stable-baselines3.readthedocs.io/en/master/guide/algos.html

SB3 algorithms - https://stable-baselines3.readthedocs.io/_/downloads/en/master/pdf/

Feature Selection with sklearn and Pandas - https://towardsdatascience.com/feature-selection-with-pandas-e3690ad8504b