Using Twitter Data in Non-Fungible Token Investing

Oswaldo Olivo
UC Berkeley School of Information
oswaldo@ischool.berkeley.edu

Abstract

Non-fungible Tokens(NFTs) are digital collectibles stored in major blockchains such as Bitcoin, Ethereum, and Solana. They have become widely popular since February 2021, and are a novel asset class similar to cryptocurrencies, becoming heavily traded in exchanges such as OpenSea and FTX. Predicting future sale prices for NFTs is both useful for artists trying to determine if their project is likely to be successful, as well as collectors trying to discover valuable collections. In this project we explore the use of Twitter data and Machine Learning models to predict whether the next sale of an NFT will sell at a higher, lower, or similar price to the previous sale. We found that using features computed from tweets such as DistilBERT embeddings, sentiment polarities, and topic modeling provides a 10% improvement in accuracy and Area-Under-Curve(AUC) for sale price movement prediction with respect to the baseline model, suggesting the fact that including features from Twitter data is a promising component for an NFT sales prediction model.

1 Introduction

Non-fungible tokens (NFTs) are digital collectibles stored in major blockchains such as Bitcoin, Ethereum, and Solana. While initially NFTs were regarded as digital art, projects have evolved to offer other benefits to their holders, such as a dividend, access to a game or social community, fractional claims to app revenue, among others.

Since February 2021, where CryptoPunks, which were initially airdropped for free, started selling to the tune of millions of dollars, NFTs have become an increasingly popular alternative asset class, similar to cryptocurrencies. Having a model for predicting whether NFT sale prices for a project will go up, down, or stay flat is useful for artists trying to determine if their creation will be successful in the near term, as well as for collectors looking to discover valuable projects before they become prohibitively expensive.

In this project we extract different features from Twitter data and use them to predict whether the next sale price of an NFT will be higher, lower, or relatively the same as the previous sale price. That is, for a given sale event, we formulate a classification problem with the labels *UP*, *DOWN*, *FLAT*, corresponding to the next sale price being higher, lower, or within 5% of the previous sale price, respectively.

We collect tweets for different time windows within the sale events, and compute DistilBERT(Sanh, 2019) embeddings of the tweets, and compute cosine similarity with tweets for the different classification labels. This intuitively corresponds to determining whether a group of tweets between sale events is closer to tweets corresponding to the group of increasing, decreasing or flat sales for other pairs of sale events.

We train an Latent Dirichlet Allocation(LDA)(Hoffman, 2014) model to determine the probability of the presence of a top-10 topic in the tweets between sale events, and use these presence probabilities as features in our models.

We compute sentiment polarity of the tweets between sale events using spaCy(SpacyAPI, 2022), which yields a probability for each of the main sentiment polarities: positive, negative and neutral sentiment. We use the average sentiment probability and number of tweets per polarity as features.

We also add the total number of tweets between the sale events as a feature.

We collected over 50K tweets for the Cool Cats NFT collection, and all the sale events in OpenSea (around 1700), from the inception of the project (July 2021) to October 2021.

We compute the prediction accuracies and AUCs of the classification task for a baseline model, Decision Trees, Boosted Decision Trees, Random Forests, Naive Bayes, K-Nearest Neighbors, and an

LSTM and RNN. We obtained a 10% improvement in accuracy and AUC with respect to a baseline model of random guessing in a balanced test set. This shows that features derived from Twitter data provide insight into NFT sale price movements, and can be an important component of a forecasting tool.

2 Project Overview

2.1 Datasets Used

- **NFT sales data:** We collected all sale events for 1000 NFTs from the Cool Cats collection, from its creation (July 2021) to October 2021, using the OpenSea event API (OpenSeaAPI, 2022). There were 2420 sales in total.

- **Twitter data:** We collected all tweets referencing the Cool Cats NFT project between July 2021 and October 2021, using the Twitter search API(TwitterAPI, 2022)There were 54,453 tweets in this time interval.

- **Embeddings data:** We used word embeddings from the output layer of DistilBERT(Sanh, 2019), la ightweight version of BERT pre-trained on a collection of unpublished books (BookCorpus) and English Wikipedia articles.

- **Sentiment data:** We used a pre-trained sentiment analysis model from the spaCy API(SpacyAPI, 2022), which was pre-trained with OntoNotes, an annotated text corpus containing telephone conversations, newsgroups, blogs, and religious texts.

- **Stopwords:** We used the set of English stopwords from the NLTK package(NLTK, 2022) as part of preprocessing the Twitter data for tour models.

2.2 Background and Problem Approach

There has been significant research in finding relationships between social media data, either by volume of tweets, sentiment, or other features, to predict stock prices and stock price movement directions(Hu, 2018)(Sawhney, 2021)(Xu, 2018).

More recently there's been a body of work applying these ideas to cryptocurrencies (Li, 2019)(Maule, 2021). In particular, our paper was influenced by previous research involving the use of word embeddings and cosine similarity of tweets, along with topic modeling, to attempt to predict whether a cryptocurrency closes below, above or about the same price from the daily open price(Maule, 2021). Similarly, there is a body of work computing average sentiment of tweets for cryptocurrencies and finding relationships between the sentiment and the closing price(Li, 2019).

One complication about our work is the smaller trading volume in NFTs: whereas the stocks and cryptocurrencies studied in literature are traded thousands of time per day, we observed some days without any NFT trades and about 2,000 trades in a period of four months (July 2021 - October 2021). Another unique feature about NFTs is that each traded token is closer to a collectible rather than a stock or a cryptocurrency. While the different shares of a company are worth the same at a given time, some NFTs within the same collection are more rare than others or are more aesthetically pleasing. Thus, only comparing the sale price of the same NFT across time is the fair comparison, rather than comparing prices of different NFTs at a specific time interval.

2.3 Problem Definition

We define our problem of interest as the following classification task:

Given a set of tweets about an NFT collection between a pair of sale events for a specific NFT. Predict whether the latter sale price of the specific NFT is higher, lower, or the same as the previous sale price.

2.4 Research Hypothesis

After defining our problem in the previous subsection, the following is the central research hypothesis of our project: **Using features extracted from Twitter data results in better NFT sale price movement prediction than random guessing**.

2.5 Problem Approach

In this project we use a combination of ideas from previous research of predicting cryptocurrency prices with Twitter data. We formulate our problem as a classification task: given Twitter data between two sale events for the same NFT, determine whether the price from the previous sale to the next sale will go up (*UP*), down (*DOWN*), or stay flat(*FLAT*). We use a 5% threshold, identical to (Maule, 2021) , to assign the labels: a decrease of more than 5% corresponds to a *DOWN* label, an increase of more than 5% to an *UP* label, and

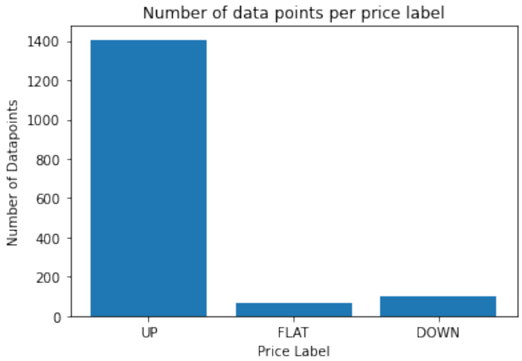

Figure 1: Number of data points per price label

a change between -5% and 5% to *FLAT*. Figure 1 shows the distribution of data points per label.

We apply the idea of computing embeddings of the tweets and computing cosine similarity to the different label groups, and use them as features(Maule, 2021). In particular, we're trying to determine whether a collection of tweets is closer to the group of tweets with the *UP*, *DOWN*, or *FLAT* labels, by using cosine similarity between the embeddings. Given the special characteristics of working with NFTs, we create a dataset entry for each pair of sale events for a specific NFT (analogous to creating an entry with Open and Close price for a stock or cryptocurrency, but taking into account the uniqueness of each NFT and low trading volume). Also due to the low trading volume, where an NFT might not be sold for weeks, we collect tweets for different time intervals preceding the next sale of an NFT, of one hour, 12 hours and 24 hours. The idea is to try to find a correlation between a sale and a short-term window of preceding tweets about the NFT collection.

Apart from computing cosine similarity with tweet embeddings, we also perform topic modeling on the tweets through the Latent Dirichlet Allocation implementation in sklearn(Hoffman, 2014). LDA creates a probabilistic distribution of a fixed number of topics n in a text dataset, based on word co-occurrences. Similar to (Maule, 2021), we compute TFIDF vectors from tweets, and we train and apply an LDA model with $n = 10$, to obtain the probabilities of the presence of a top-10 topic in the TFIDF vectors of a sequence of tweets between sale events, and use these probabilities as features in our model.

We also add the number of tweets as a feature, which can be used as a weighting factor for features related to text analysis.

Finally, similar to previous research(Li, 2019), we perform sentiment analysis over the tweets between sale events using the spaCy framework(SpacyAPI, 2022). The model is pre-trained on OntoNotes, and provides a probability score for three sentiments (positive, neutral, negative). We use the average probability of each of the sentiments across all tweets between sale events, as well as the number of tweets per sentiment, where a tweet is considered to have a specific sentiment according to the largest probability provided by the sentiment analyzer.

2.6 Parsing tools used

We queried the Twitter(TwitterAPI, 2022) and OpenSea(OpenSeaAPI, 2022) APIs to obtain tweets and NFT sale events, respectively. The endpoints return JSON data, so we implemented some functions to flatten the JSON into CSV rows, extract the subset of fields of interest (tweet id, author id, created at, text for tweets; NFT id, sale date, and sale price for NFT sales), and generate CSV files to be read as a Pandas dataframe.

2.7 Data preprocessing

We cleaned the tweets by applying the following pre-processing steps, as was done in (Maule, 2021):

1. Conversion to lowercase.
2. Removal of extra spaces.
3. Removal of URLs.
4. Word tokenization using the NLTK tokenizer(NLTK, 2022).
5. Removal of stopwords, using the set of NLTK stopwords(NLTK, 2022).
6. Applied stemming using the NLTK Porter stemmer(NLTK, 2022).

We created one dataset entry for each pair of consecutive NFT sale events, where the starting price is the price of the previous sale, and the ending price is the price of the next sale. Then, we computed the target label for each entry: a decrease of more than 5% corresponds to an *DOWN* label, an increase of more than 5% corresponds to an *UP* label, and a change within 5% of the starting price is the *FLAT* label.

We divided the dataset into 80% training and 20% testing. Similar to previous work, we extracted a sample of the dataset that is balanced in terms of classification labels(Maule, 2021).

3 Features

In this section we describe all the features computed from the Twitter data that are fed as inputs to our classification model.

3.1 DistilBERT Embeddings and Cosine Similarity to Tweet Groups

Previous research in using Twitter data to determine direction of closing prices of cryptocurrencies used DistilBERT(Sanh, 2019) word embeddings as a feature(Maule, 2021). DistilBERT is a lightweight version of BERT, pre-trained on a collection of unpublished books (BookCorpus) and English Wikipedia articles. It has 40% less parameters and runs 60% faster than BERT, while preserving 95% of the performance on the GLUE natural language understanding benchmark, according to(Sanh, 2019), which makes it a popular alternative for computing word embeddings.

For each dataset entry, we concatenated all the tweets for the last t hours before the end sale (where $t = 1, 12, 24$). Then, we compute the embeddings using the pre-trained DistilBERT output layer, which returns 768 language features for the input text. After computing the embeddings for all entries in the dataset, we concatenate all the tweets with the same label, which results in three groups of concatenated tweets (*UP*, *DOWN*, and *FLAT*). Then, for each entry, we compute the cosine similarity between the embeddings of the tweets for the data point and the embeddings of the three groups of concatenated tweets for the labels. We use the cosine similarity to each group as a feature, as well as an indicator variable of which group has the greatest cosine similarity to the data point.

3.2 Latent Dirichlet Allocation Topic Modeling

Latent Dirichlet Allocation(LDA)(Hoffman, 2014) topic modeling has been used in predicting the direction of end-of-day closing prices in cryptocurrencies(Maule, 2021).

We compute TFIDF vectors from the concatenation of the tweets for the last $t = 1, 12, 24$ hours preceding the end sale of an NFT, and apply the LDA topic modeling from sklearn with $n = 10$ topics. We add 10 features to our datapoint, corresponding to the probabilities for each of the top-10 topics in the concatenation of the tweets preceding the NFT sale.

3.3 Sentiment Analysis

Sentiment analysis has been a key component in many research papers that use Twitter data in predicting stock(Hu, 2018)(Sawhney, 2021)(Xu, 2018) and cryptocurrency(Li, 2019) prices movements. We use the spaCy sentiment analyzer(SpacyAPI, 2022), which has been used in previous research for predicting price fluctuations in cryptocurrencies(Li, 2019).

For each datapoint, we compute the sentiment probability distribution (positive, neutral, and negative) for each tweet in a given time window of $t = 1, 12, 24$ hours before the sale. Then we add the following six features to our data point: the average probability for each sentiment polarity across all tweets in the time window, and the number of tweets per polarity (e.g. a tweet is positive if the probability of positive sentiment is greater than the neutral and negative sentiment, etc.).

4 Model

In this section we discuss the models that we used for our classification task.

4.1 Baseline Model

Given that we have a balanced test dataset, our baseline model corresponds to randomly guessing one of the three candidate labels, which results in a 33% accuracy and 50% Area Under Curve (AUC).

4.2 Advanced Models

Similar to previous research in cryptocurrency price direction, we ran out-of-the box models, one custom RNN and one custom LSTM(Maule, 2021). We used the out-of-the-box sklearn implementations of *Random Forests*, *Decision Trees*, *K-Nearest Neighbors* with $k = 3$, *Boosted Decision Trees (AdaBoostClassifier)* and *Gaussian Naive Bayes*. We implemented a custom neural network with a 16-node LSTM layer, three RELU dense layers with 16, 8 and 4 nodes, respectively, and an output softmax dense layer. We implemented a custom neural network identical to the previous network, but with an initial simple RNN layer (instead of an LSTM layer). These last two custom neural networks have the same architecture as previous work in cryptocurrency price movements(Maule, 2021).

Model	Accuracy	AUC
Baseline	**33%**	**50%**
Random Forest	34.6%	56.7%
Decision Tree	37.3%	52.9%
Naive Bayes	**42%**	**59.5%**
KNN	36.6%	54.2%
Boosted Decision Trees	36.6%	57.3%
LSTM	33%	50%
RNN	33%	50%

Table 1: Accuracy and AUC for different models using the last hour of Twitter data.

5 Results

In this section we discuss running the models with different features and different time windows of Twitter data.

5.1 Results for Different Feature Sets

The main features are cosine similarity to tweet groups, topic modeling, and sentiment analysis.

See table 1 for the accuracy and AUC for different models.

We obtained our best results with Naive Bayes, with an almost 10% improvement in accuracy and AUC with respect to the baseline. The results in table 1 correspond to a one-hour time window of features from Twitter data for the hour preceding end sale of a datapoint. Interestingly, the LSTM and RNN models did not improve the baseline. We found that the models overfit the traning data (more than 96% training accuracy), and tested using less layers, early stopping, and dropout layers, without observing any meaningful improvement. We suspect having a larger dataset or a simplification of the architecture might yield improvements, but so far were unable to reproduce the success of the custom neural networks from research in cryptocurrencies(Maule, 2021).

5.2 Unusual Data Patterns

As mentioned before, due to the uniqueness of NFTs, we doní have a "central" daily closing price as we have for stocks or cryptocurrencies, so the approach we took was to collect all tweets between pairs of sale events, and try to predict the direction of the movement from one sale price to the next. However, concatenating all tweets between sales did not scale for some NFTs that had no sales for weeks: we run out of memory when trying to compute DistilBERT embeddings across weeks of Twitter data. Therefore, we decided to concatenate tweets for different time windows preceding the last sale of the NFT: last hour of tweets, last twelve hours, and last 24 hours.

We didn't find a significant increase in the number of tweets when going from one to twenty four hours windows in our dataset, which explains why we're only reporting the results of using the last hour of tweets.

The original dataset of sale events for Cool Cats is skewed towards price increases, with 89% of labels being *UP*. We could either balance the entire dataset, which reduces the number of entries, or balance the test set, which results in training with an unbalanced dataset and testing with a balanced dataset. Due to the low trading volume in NFTs, we decided to preserve as much training data as possible and test on a balanced set. In the future we would like to run our models on larger historical NFT data that shows both bear and bull markets.

We noticed that there are many tweets created by bots or that could be classified as spam, such as mentioning Cool Cats but promoting another project, or simply creating unrelated sweepstakes. Spam is also a problem seen in cryptocurrency research(Maule, 2021)(Li, 2019). Previous research in cryptocurrencies used a curated list of influential Twitter accounts as a feature to mitigate this problem(Maule, 2021), but computing this list is also a problem on its own, and can be biased against legitimate tweets from less known accounts. Other research discovered that by using sentiment analysis, we could discover spam-like tweets(Li, 2019), so we took the approach of including sentiment analysis, and counts of tweets by polarity for this reason. A promising approach might be to train a spam classifier as part of a pre-processing step for Twitter data.

6 Comparison with Existing Research

In the most closely related work to ours, the researchers were able to obtain an 61-88% accuracy in predicting the closing price direction labels for cryptocurrencies for individual tweets, which is a 28-55% improvement over the baseline(Maule, 2021). In our case, we were able to obtain an improvement of roughly 10% over the baseline in terms of both accuracy and AUC. However, our problem is more complicated than the one addressed in the cited study, due to low trading volume and lack of a central closing prices – we resorted to using pairs of sale events to make a fair price comparison due to NFTs within the same collection being different. We can say that adding text analysis of Twitter data to an NFT trading algorithm clearly outperforms random guessing, and can be used as a signal in a more sophisticated strategy.

7 Conclusion

In this project we have studied for the first time the use of Twitter data in predicting the direction of sale prices of NFTs. This task is more complicated than previously studied problems such as stock and cryptocurrency price predictions due to low trading volume, scarcity of historical data for such a novel asset class, and the lack of a central price metric for an NFT collection (NFTs are different from each other, thus they sell at different prices within the same time intervals). However, we were able to significantly outperform random guessing in a balanced dataset, by roughly 10% in terms of both accuracy and AUC. Thus, the use of NLP techniques over Twitter data provides a valuable signal as part of a more sophisticated trading strategy. There is promise in further refining our models, such as adding spam detection to filter out noisy tweets, gathering data during bull and bear markets, and collections with more diverse price movements.

References

Hoffman, M.D., Blei, D.M., Wang, C., and Paisley, J. 2014. Stochastic Variational Inference *In Journal of Machine Learning Research*, 14:4, 1303–1347.

Hu, Z., Liu, W., Bian, J., Liu, X. and Liu, T. 2018. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction *WSDM '18*, 261–269.

Li, T. Chamrajnagar, A. S., Fong, X. R., Rizik, N. R., and Fu, F. 2019. Sentiment-based prediction of alternative cryptocurrency price fluctuations using gradient boosting tree model *Frontiers in Physics*, 7:98.

Maule, A.P.P. and Johnson, K. 2021. Cryptocurrency Day Trading and Framing Prediction in Microblog Discourse *In Proceedings of the Third Workshop on Economics and Natural Language Processing*, 82–92.

NLTK https://www.nltk.org/

OpenSea API https://docs.opensea.io/reference/api-overview

Sanh, V., Debut, L., Chaumond, J., Wolf, T. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. https://arxiv.org/abs/1910.01108

Sawhney, R., Wadhwa, A., Agarwal, S., Shah, R. R. 2021. FAST: Financial News and Tweet Based Time Aware Network for Stock Trading *In Proceedings of the Sixteenth Conference of the European Chapter of the Association for Computational Linguistics*, 2164–2175.

spaCy API https://spacy.io/api

Twitter API https://developer.twitter.com/en/products/twitter-api

Xu, Y. and Cohen, S.B. 2018. Stock movement prediction from tweets and historical prices. *In ACL '18*, 1970–1979.