Cryptocurrency Price Prediction Using News and Social Media Sentiment

Connor Lamon, Eric Nielsen, Eric Redondo

Abstract— **This project analyzes the ability of news and social media data to predict price fluctuations for three cryptocurrencies: bitcoin, litecoin and ethereum. Traditional supervised learning algorithms were utilized for text-based sentiment classification, but with a twist. Daily news and social media data was labeled based on actual price changes one day in the future for each coin, rather than on positive or negative sentiment. By taking this approach, the model is able to directly predict price fluctuations instead of needing to first predict sentiment. The final version of the model was able to correctly predict, on average, the days with the largest percent increases and percent decreases in price for bitcoin and ethereum over the 67 days encompassing the test set.**

I. Introduction

As the economic and social impact of cryptocurrencies continues to grow rapidly, so does the prevalence of related news articles and social media posts, particularly tweets. As with traditional financial markets, there appears to be a relationship between media sentiment and the prices of cryptocurrency coins. While there are many causes of cryptocurrency price fluctuation, it is worthwhile to explore whether sentiment analysis on available online media can inform predictions on whether a coin's price (i.e., perceived value) will go up or down.

Specifically, in this project our team worked to answer two questions: (1) Can sentiment analysis on news article headlines and/or social media posts produce accurate predictions about the future price fluctuations of bitcoin, litecoin, and ethereum? (2) If predictions from both sets of data are possible, which is the better indicator of future coin prices?

The input of our system is text data from news headlines and tweets, aggregated by day and kept in order of occurrence to preserve the time-series nature. Traditional supervised learning binary classification algorithms were then used to assign each news headline and tweet a label of 0 or 1 for each coin, indicating predictions of a price decrease or price increase one day in the future, respectively. The majority label for each coin, on each day, was then used as the final daily prediction.

Our ultimate goal is to refine this price prediction model and incorporate it into a larger system that automatically and intelligently manages a cryptocurrency portfolio. The groundwork for such a system was completed by our team for our concurrent project in the Stanford class CS221: Artificial Intelligence - developing a reinforcement learning model that manages a portfolio of bitcoin, litecoin, and ethereum. We have experimented with feeding the output of this classifier as features for the portfolio manager, but improved prediction accuracy is necessary before the features offer value.

II. Related Work

There have been previous attempts to utilize sentiment from tweets to predict fluctuations in the price of bitcoin. Coliannni et al. [1] reported 90% accuracy in predicting price fluctuations using similar supervised learning algorithms, however their data was labeled using an online text sentiment API. Therefore, their accuracy measurement corresponded to how well their model matched the online text sentiment API, not the accuracy in terms of predicting price fluctuations. Similarly, Stenqvist and Lonno [2] utilized deep learning algorithms, on a much higher frequency time scale of every 30 minutes, to achieve a 79% accuracy in predicting bitcoin price fluctuations using 2.27 million tweets. Neither of these methods use data labeled directly based on price fluctuations, nor did they analyze the average size of price percent increases and percent decreases their models were predicting.

More classical approaches of using historical price data of cryptocurrencies to make predictions have also been tried. Hegazy and Mumford [3] achieved 57% accuracy in predicting the actual price using supervised learning methods. Jiang and Liang [4] utilized deep reinforecment learning to manage a bitcoin portfolio that made predictions on price. They achieved a 10x gain in portfolio value. Last, Shah and Zhang [5] utilized Bayesian regression to double their investment over a 60 day period. None of these methods utilized news or social media data to capture trends not apparent in the price history data.

Our project is differentiated from the projects outlined above in several key ways. First, we attempt to make price predictions using news data in addition to social media data. Second, we predict price fluctuations for ethereum and litecoin in addition to bitcoin. Third, we label our data based on actual price changes rather than traditional text sentiment. Finally and most importantly, we analyze predicted price fluctuations by size (i.e., percent change); this is a valuable data point to know given the extreme daily volatility of current cryptocurrency markets.

III. Datasets

Data input to the model comes in two forms. First, we obtained daily price data for bitcoin, ethereum, and litecoin from Kaggle [7]. Second, we built online scraping scripts to acquire ~3,600 cryptocurrency-related news article headlines from cryptocoinsnews.com [6] and ~10,000 bitcoin-related tweets, ~10,000 ethereum-related tweets, and ~10,000 litecoin-related tweets (using a customized version of J. Henrique’s project [8], which itself uses the twitter API [9]). The data acquired is from the time period January 1, 2017 - November 30, 2017.

The daily coin price data was used to label each news headline with six values and each tweet with two values. For a news headline occurring on day d_i, three labels represent each coin’s price change on day d_{i+1} and three labels represent the change on d_{i+2}. For a bitcoin-related tweet occurring on day d_i, one label represents bitcoin’s price change on day d_{i+1} and one label represents the change on d_{i+2}. Similar logic follows for ethereum and litecoin-related tweets. The labels are binary $\in \{0, 1\}$, with a ’0’ indicating a decrease in price and a ’1’ indicating an increase. Once labeled, the news headlines and tweets were then segmented into train (60%), development (20%), and test sets (20%). These data sets correspond to the date ranges Jan. 1, 2017 - July 20, 2017, July 21, 2017 - Sept. 23, 2017, and Sept. 24, 2017 - Nov. 30, 2017, respectively.

TABLE I
Example Labeled News Headline

Date	Headline
01/01/2017	Bitcoin Starts 2017 at $1000

+1D BTC	+2D BTC	+1D ETH	+2D ETH	+1D LTC	+2D LTC
1	1	0	1	1	0

TABLE II
Example Labeled Litecoin-Related Tweet

Date	Tweet	+1D LTC	+2D LTC
11/23/2017	#Litecoin is Life at $1000	1	0

IV. Methods

Feature Extraction

Each headline and tweet was tokenized using spaCy [10] library functions and additional logic to convert all text to lowercase and to remove white space, punctuation, stop characters (e.g. ’a’, ’the’, ’and’), and several specific types of strings (discussed in Experiments section). The features selected from the headlines and tweets include all word 1- and 2-grams that appear in the text. Feature extraction was performed using the scikit-learn [11] CountVectorizer method on the tokenized text, which represents the features as a sparse matrix of token counts.

Classification

The model uses a classifier to learn feature weights that are used for predicting data labels. Our initial model was built using a simple logistic regression classifier. We chose this method initially as it is simple to understand and allowed us to perform error analysis quickly to determine how to proceed further. Ultimately, linear support vector classification, multinomial Naive Bayes, and Bernoulli Naive Bayes were also tried, but logistic regression produced the best results on the development set. All classifiers were implemented using scikit-learn library functions.

A. Logistic Regression

Logistic regression was the first classifier implemented due to its simplicity and ease of understanding. It is a classification algorithm that utilizes the sigmoid function, $\sigma(x)$,to convert the dot product of a weights vector, W, with the features, x, to output the probability of the example being a 1. A probability greater than 0.5 corresponds to a label of 1, otherwise the label is 0. The sigmoid and logistic loss function are as follows:

$$\sigma(x) = \frac{1}{1+\exp(-W^T x)} \tag{1}$$

$$l(x) = \sum_{i=1}^{m} y^{(i)} \log(\sigma(x^{(i)}) + (1-y^{(i)}) \log(1-\sigma(x^{(i)})) \tag{2}$$

B. Linear Support Vector Machine

Support vector machines are more powerful classification algorithms that create ”support” vectors to maximize the margin, or euclidean distance, between all data points and the decision boundary. The standard optimization problem that defines finding the optimal margin is as follows:

$$\min_{\gamma,w,b} \frac{1}{2}||w||^2 \tag{3}$$

$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1, i = 1, .., m$$

C. Naive Bayes

Naive Bayes is a generative classification algorithm that makes a large assumption about the data: all examples, x_i, are conditionally independent given the labels y. The model then uses maximum likelihood estimation to maximize the joint likelihood of the data:

$$L(X, Y) = \prod_{i=1}^{m} p(x^{(i)}, y^{(i)}) \tag{4}$$

Predictor

In order to make a single prediction about the change in each coin’s price on a given day, classifier output is first aggregated by day (e.g., all news headlines from January 1, 2017 are considered together) for each coin. The final prediction is then assigned based on the majority label associated with each coin on each day. Predictions are made separately based on only news headlines and only tweets (discussion of combined predictions in Results and Discussion section).

V. Experiments

Evaluation Metrics

Our solution was formulated as a binary classification problem, so we initially utilized the traditional accuracy measure to evaluate our model. However, since our final daily predictions are based on the majority of aggregated predicted labels for each day, raw classification accuracy was not the best way to evaluate the model. We determined that a superior evaluation metric was to compare the average price change for days when the model makes correct predictions vs. the average price change for days when the model makes incorrect predictions (i.e., does the model correctly predict the largest price increases and decreases). Taking this approach, we developed a modified confusion matrix that considers the of the price changes predicted correctly and price changes predicted incorrectly. We also looked at the average magnitude we predicted in either direction to see if we were predicting the majority of the largest swings.

Different classifiers

Four different classification algorithms were utilized, as previously discussed. The first attempt utilized logistic regression, which gave promising results. Keeping all other model components the same, we experimented with three other classifiers (Linear SVM, Multinomial NB, and Bernoulli NB) to determine if there would be an improvement in the ability to detect large percent changes in prices. We chose to try an SVM and Naive Bayes models because both seemed to have a chance to create better classifiers on our textual based, high dimensional data vs. logistic regression.

Hyperparameters

Due to the time series nature of our data, we were unable to use standard cross-validation. Fortunately, the model had a small number of hyperparameters to evaluate. The first is the size of n-grams to use as features. We found that using an n-gram range of 2 performed the best, allowing the model to capture important tokens without adding too much noise. The second hyperparameter is the regularization for the various classifiers. We found that using an inverse regularization parameter of 0.9 provided the best trade-off.

Investigating token weights

The most positive and most negative token weights were investigated to determine possible causes of inaccurate predictions. Initially, we found that numbers not related to coin prices often appeared among the most negative tokens, particularly for tweets. We also found that various-sized strings of periods (e.g., '..', '...') appeared among the most positive tokens, again particularly in tweets. We updated the text cleaner to remove numbers and strings of periods from token consideration, and these updates improved model accuracy.

Sample final token weights are shown in Tables III and IV. These show the largest positive-weighted tokens and smalled negative-weighted tokens for headline-based predictions on bitcoin price one day in the future (using the logistic regression classifier).

TABLE III
Largest Token Weights, Headline-based +1d BTC

Token	Weight
break	1
continue	0.85
time high	0.82
spike	0.8
consider	0.79

TABLE IV
Smallest Token Weights, Headline-based +1d BTC

Token	Weight
begin	-0.88
datum	-0.87
litecoin price	-0.82
german	-0.8
bitcoin hard	-0.75

Combining with CS221 project

For our team's project in CS221 - Artificial Intelligence, we utilized the same cryptocurrency price data to develop a reinforcement learning model that makes daily trades for a portfolio comprised of bitcoin, ethereum, and litecoin. We experimented with using the output of our prediction model as features for the RL algorithm, but it ended up simply adding more noise. We believe this was due to two reasons: (1) limited prediction accuracy, and (2) the use of binary {0,1} prediction labels. Given time constraints, complete integration of the two projects has been postponed for now. Once prediction accuracy is increased, we plan to utilize the raw prediction probabilities, aggregated and weighted appropriately to account for the difference in the daily volumes of news headline data vs. tweet data (the model uses many more tweets than headlines each day), as features in the reinforcement learning model.

VI. Results and Discussion

To give context to our results, it is important to understand the general price behavior of each coin during the test set time period.

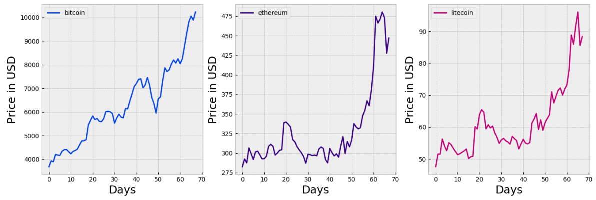

Fig. 1. BTC, ETH, LTC Prices from Sept. 24, 2017 - Nov. 30, 2017

All coin prices were generally increasing over this time period. The model uses general price trends in logic to combine headline-based and tweet-based price predictions for each day (as described earlier, these predictions are made separately). If the recent price trend for a coin is positive, the model is biased toward price increase predictions (i.e., if either the headline-based or tweet-based prediction is a price increase, the combined prediction is a price increase). The opposite is true if the recent price trend for a coin is negative. All results shown are for combined predictions.

The optimal classifier varied by coin, as a result of the datasets having different properties. We favored classifiers that produced the most balanced results (i.e., able to correctly predict both price increases and decreases) and that correctly predicted the largest (magnitude) price changes in both directions. When combining headline-based and tweet-based predictions for a coin, the same classifier is used for both.

Finally, all results shown are for price predictions one day in the future. Though the model also makes predictions for two days in the future, we are currently focused on one day predictions as they are more relevant to daily trading decisions.

Bitcoin results

Best classifier: Logistic Regression

Simple logistic regression with an inverse regularization term of 0.9 resulted in the optimal classifier for bitcoin price predictions. Table V shows the model was able to predict 43.9% of price increases correctly and 61.9% of price decreases correctly. More notably, Table VI shows that although the model predicted less than half of the bitcoin price increases correctly, it tended to correctly predict the days with the larger percent increases - on average correctly predicting days with a 4.9% price increase, and incorrectly predicting days with a 2.83% price increase. Similarly for price decreases, the model correctly predicted on average, the days with the larger (magnitude) percent decreases. Overall, bitcoin price predictions were the most balanced of any coin (i.e., a decent number of both price increases and decreases were predicted correctly). Daily bitcoin prediction results can be qualitatively seen in Fig. 2.

TABLE V

Bitcoin Prediction Confusion Matrix

	Pred. Increase	Pred. Decrease
Actual Increase	**43.9%**	56.1%
Actual Decrease	38.1%	**61.9%**

Ethereum Results

Best classifier: Bernoulli Naive Bayes

Unlike for bitcoin, a Bernoulli Naive Bayes classifier provided the best ethereum price predictions. Table VII shows that the model was very effective in predicting price increases, with 75.8% of days predicted correctly. Prediction accuracy for price decreases was much lower, at only 16.1%. However, Table VIII shows that the model was able to correctly predict the day with the largest (magnitude) price decrease. very effective in predicting the largest percent changes in both directions. Daily ethereum prediction results can be qualitatively seen in Fig. 3.

TABLE VI

Bitcoin Prediction Avg. Price Changes

	Avg. % Increase	Avg. % Decrease
Correct Pred.	**4.90%**	**-2.71%**
Incorrect Pred.	2.83%	-1.64%

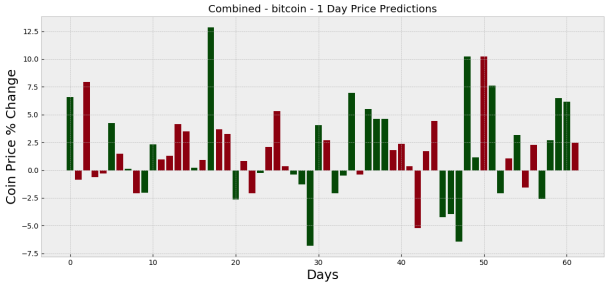

Fig. 2. Daily bitcoin percent price changes during test set time period. Height and direction of each bar represents price change; green and red coloring represents correct and incorrect predictions, respectively.

TABLE VII

Ethereum Prediction Confusion Matrix

	Pred. Increase	Pred. Decrease
Actual Increase	**75.8%**	24.2%
Actual Decrease	83.9%	**16.1%**

TABLE VIII

Ethereum Prediction Avg. Price Changes

	Avg. % Increase	Avg. % Decrease
Correct Pred.	**3.96%**	**-1.95%**
Incorrect Pred.	2.05%	-1.76%

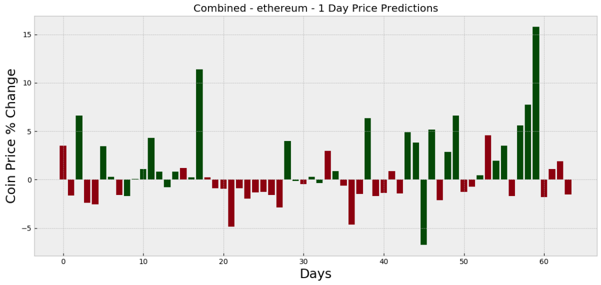

Fig. 3. Daily ethereum percent price changes during test set time period. Height and direction of each bar represents price change; green and red coloring represents correct and incorrect predictions, respectively.

Litecoin Results

Best classifier: Logistic Regression
Litecoin price prediction performance was the worst of the three coins. A logistic regression classifier with an inverse regularization term of 0.9 performed the best, however the model almost exclusively predicted price decreases. This is likely due to the fact that the increase in litecoin price during the test set time period was unprecedented compared to previous months. Likewise, litecoin popularity among the general public also increased drastically during the time period. As a result, it is likely that the distribution of the types of text data for litecoin-related headlines and tweets was different from the train and development sets to the test set. Quantitative and qualitative results can be seen in the tables and figure below.

TABLE IX
LITECOIN PREDICTION CONFUSION MATRIX

	Pred. Increase	Pred. Decrease
Actual Increase	**0%**	100%
Actual Increase	0%	**100%**

TABLE X
LITECOIN PREDICTION AVG. PRICE CHANGES

	Avg. % Increase	Avg. % Decrease
Correct Pred.	**0%**	**-2.77%**
Incorrect Pred.	4.26%	0%

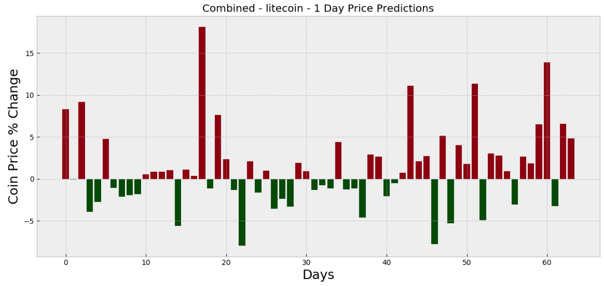

Fig. 4. Daily litecoin percent price changes during test set time period. Height and direction of each bar represents price change; green and red coloring represents correct and incorrect predictions, respectively.

Discussion

The model works relatively well for identifying general trends in coin prices, but struggles to accurately predict daily price fluctuations that are not in line with the general trend. Specifically, there is a general increase in both bitcoin and ethereum prices during the test set time period, and the model correctly picks up on this via the text input and most often predicts additional price increases. For litecoin, however, the train set time period primarily consisted of small price increases and many price decreases. As a result, the final model was not able to predict the very large increase in price during the test set time period.

VII. Conclusion and Future Work

Though the model still has much room for improvement, our two primary project objectives were completed – a functioning model was developed that makes cryptocurrency price predictions using non-technical data, and the model is generally able to predict the largest (magnitude) price increases and decreases correctly.

Additional experiments and model updates to improve performance will be made in the future, including:

- Training the model with a combination of news and twitter data so that classification and prediction will be more robust to different trends in both data sets
- Integrating additional types of media (e.g., from other news sources, Slack channels, subreddits) and larger volumes of input
- Investigating different strategies for labeling training data (e.g., using a neural network to label based on text sentiment).
- Updating the baseline model to perform more robust text pre-processing and to potentially consider additional features (e.g., news source, author)

VIII. Contributions

a) Connor Lamon:
Developed news headline scraping script. Implemented the baseline pipeline consisting of the initial text pre-processing, vectorizer and initial model selection. Performed error analysis of train set error vs development set error, and tried regularization to fix it. Performed final hyperparameter tuning to find optimal classifiers.

b) Eric Nielsen:
Developed scripts to obtain tweets and to label both news headlines and tweets. Developed functions to print / plot results. Worked to analyze and apply experimental results for tweets, including improvements to text pre-processing. Developed function to combine headline-based and tweet-based predictions. Optimized model structure to streamline experimentation (compartmentalizing functionality and enabling the saving / loading of model results).

c) Eric Redondo:
Implemented initial custom evaluation metric for evaluating daily performance of classifiers. Ran experiments to compare daily performance of different models using both news headline and tweet data. Worked on and experimented with integration of model outputs into reinforcement learning algorithm for cryptocurrency portfolio management.

References

[1] Stuart Colianni, Stephanie Rosales, and Michael Signorotti. *Algorithmic Trading of Cryptocurrency Based on Twitter Sentiment Analysis.* CS229 Project, 2015.

[2] Evita Stenqvtst, Jacob Lnn *Predicting Bitcoin price fluctuation with Twitter sentiment analysis*

[3] Kareem Hegazy and Samuel Mumford. *Comparitive Automated Bitcoin Trading Strategies*. CS229 Project, 2016 http://www.diva-portal.org/smash/get/diva2:1110776/FULLTEXT01.pdf

[4] Zhengyao Jiang and Jinjun Liang *Cryptocurrency Portfolio Management with Deep Reinforcement Learning* https://arxiv.org/abs/1612.01277v5

[5] Devavrat Shah and Kang Zhang *Bayesian regression and Bitcoin* https://arxiv.org/pdf/1410.1231v1.pdf

[6] CryptoCoins News, https://www.cryptocoinnews.com

[7] SJ Kumar, Cryptocurrency Historical Prices, [Data files] retrieved from https://www.kaggle.com/sudalairajkumar/cryptocurrencypricehistory

[8] J Henrique, Get Old Tweets Programatically, [Github repository] retrieved from https://github.com/Jefferson-Henrique/GetOldTweets-python

[9] Twitter API, https://developer.twitter.com/en/docs/tweets/search/overview

[10] spaCy, https://spacy.io/

[11] scikit-learn, http://scikit-learn.org/stable/