



Project Report

Crypto Sentiment Price Prediction

November 2021

Version 1.0

ISO2PT-PLP-GRP-11

Members: Vincent Ng, Zhou Zhe, Dong Xiaoguang



1.0 Project Summary

The cryptocurrency market is a rapidly growing area of financial interest and has seen extreme price volatility in the past decade. The price fluctuations are largely driven by retail investors who are in turn largely affected by sentiments and emotions.

This project aims to make sense of market sentiment and generate trade signals based on market sentiment.

We'll be using Bitcoin as the cryptocurrency of choice as it is the most established and commands the highest market capitalization and therefore exhibits the least volatility compared to other cryptocurrencies. We've obtained Bitcoin's historical daily price from 2017 to 2021 using the Binance price API.

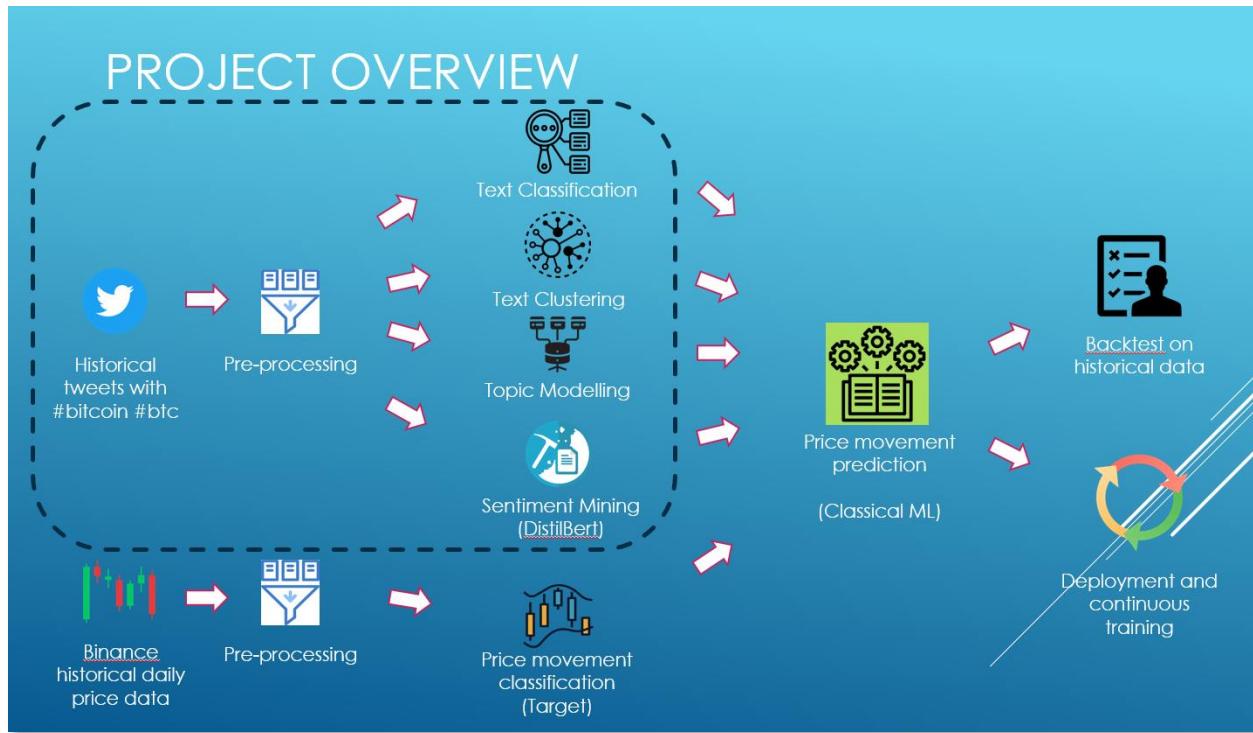
Market sentiment will be based on a third party collected Twitter tweets containing #Bitcoin and #BTC (found on Kaggle) from Feb to Sep 2021.

By extracting language features from the tweets using techniques such as Text Classification, Text Clustering, Topic Modelling and Sentiment Mining, we're able to make predictions on how Bitcoin price is going to move in the next few days with an accuracy of over 70%

We also proceeded to back test the predictions to see how well such a model will perform on a simulation portfolio and found that it outperformed a buy-and-hold strategy by a factor of 5.8



2.0 System Design



The system comprises of 2 pipelines as seen above. The language features extraction pipeline (in dotted outline) will tap on the techniques covered in the Text Analytics and Sentiment Mining courses, while the price movement classification, training and prediction will make use of concepts and techniques learnt in earlier courses in the Intelligent Systems programme.



2.1 Language features extraction pipeline

The data source for the language features extraction pipeline is found through Kaggle. It is a collection of 1.18 million Twitter tweets that contain the hashtag #BTC or #Bitcoin spanning from February to September 2021.

2021-02-10 23:52:25	#BTC #Bitcoin #Ethereum #ETH #Crypto #cryptotrading \$RSR I know i told you guys the target was \$0....
2021-02-10 23:52:08	.@Tesla's #bitcoin investment is revolutionary for #crypto but other firms may not do the same just ...
2021-02-10 23:52:04	Annnnd #btc #Bitcoin is headed even higher now... https://t.co/EHAduI087d

After pre-processing, applying Text Analytics techniques on the tweets, and aggregating their daily values, we were able to produce feature columns that can be used by our final model for price movement prediction. We noted that there are days inside the range with no tweets captured. We'll just have to make do with the tweet data that is available.



2.2 Price movement classification pipeline

The Bitcoin daily price movement (our prediction target) can be easily downloaded using a library (python-binance) and the price movement over the next N-days were analyzed to see which offers the best performance.

timestamp	open	high	low	close	volume
8/17/2017	4469.93	4485.39	4200.74	4285.08	2812379
8/18/2017	4285.08	4371.52	3938.77	4108.37	4994494
8/19/2017	4108.37	4184.69	3850	4139.98	1508239
8/20/2017	4139.98	4211.08	4032.62	4086.29	1915636
8/21/2017	4086.29	4119.62	3911.79	4016	2770592

The final model will make use of classical Machine Learning to make predictions on price movement and we made use of TPOT library which employs Genetic Programming to find the best performing Machine Learning model and it's corresponding hyperparameters.

With the final model, we'll feed it historical data to obtain trade signals in the form of buy and sell based on predicted price change after 3 days.

Finally, these trade signals are passed into a Python back testing simulator library with Bitcoin historical prices to study how well the portfolio performs over the simulation period and compare it with a simple buy and hold strategy.



3.0 System Models

3.1 Text Classification

As the collected tweets data corpus does not contain any labels, the supervised machine learning is unable to be applied for text classification training as usual. In order to perform text feature engineering, the lexicon rule-based text classification approach is adopted as first sentiment analysis model for labeling sentiment scores. The sentiment scores will be used as features for the crypto sentiment price prediction.

The sentiment analysis library VADER (Valence Aware Dictionary and Sentiment Reasoner) is used for labeling tweets in our project, which is a lexicon and rule-based sentiment analysis tool and is specifically attuned to sentiments expressed in social media.

According to the research done by C.J. Hutto and Eric Gilbert, the VADER lexicon performs exceptionally well in the social media domain. The correlation coefficient shows that VADER ($r = 0.881$) performs as well as individual human raters ($r = 0.888$) at matching ground truth (aggregated group mean from 20 human raters for sentiment intensity of each tweet). When they further inspect the classification accuracy and see that VADER ($F1 = 0.96$) even outperforms individual human raters ($F1 = 0.84$) at correctly classifying the sentiment of tweets into positive, neutral, or negative classes.

After taking the experimentation, the common text pre-processing tasks such as lowercasing, removing punctuation, removing stop words, stemming and lemmatization will not be used for VADER sentiment analysis and only adopt some like removing URLs and hashtags. Because five generalizable heuristics features are developed for performing high accuracy in DADER.

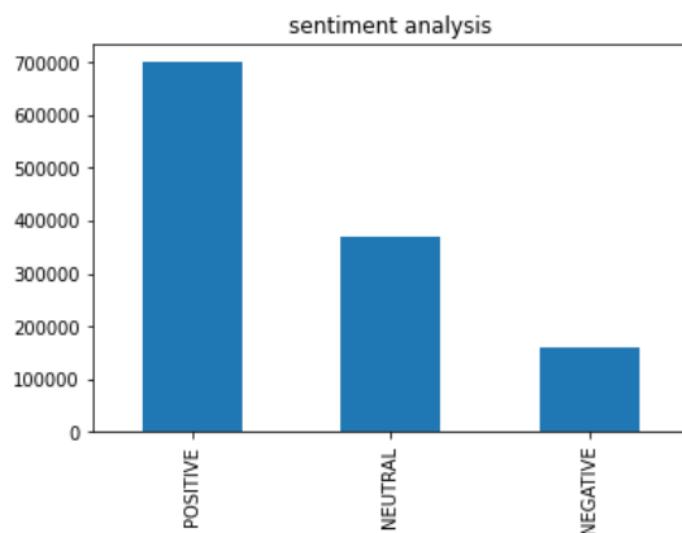


- Punctuation: This can increase the sentiment intensity without modifying the underlying sentiment. (e.g., “Good!!!”)
- Capitalization: This also signals emphasis with affecting the underlying mood of the text. (e.g., using all CAPS for words or phrases)
- Degree modifiers: These can decrease or increase sentiment intensity (e.g., “very” or “kind of”)
- Emoticons: It understands many sentiment-laden emoticons (e.g., :) and :D)
- Analyzing tri-grams preceding a lexical feature: this allows VADER to identify shifts where the negation flips the polarity of the text.

VADER takes in a string and returns a dictionary of scores in each of four categories: negative, neutral, positive and compound.

Sample result: {'neg': 0.0, 'neu': 0.208, 'pos': 0.592, 'compound': 0.5404}

The compound score is used as our feature column for subsequent price prediction, which is computed by summing the valence scores of each word in the lexicon, adjusted according to the rules, and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive). The following diagram illustrates the distribution of sentiment scores:



3.2 Text Clustering

3.2.1 Tweets Preprocessing

Before feeding into the model, the tweets are preprocessed. The preprocess function takes a string as input and returns the processed text. There are several steps of pre-processing.

First, all the tweets are converted to lowercase. Followed by which, the system removes all the URLs in the data as URLs contain no semantic or lexical information normally. Then all the ticker symbols are replaced with spaces. The ticker symbols are any stock symbol that starts with \$. Besides, there are also many usernames which are words starting with @. These usernames are also replaced with spaces. Furthermore, the preprocessing method replaces everything not a letter or apostrophe with a space. After which, single letter words and any leading / trailing whitespaces are removed. Finally, stop words ('crypto', 'cryptocurrency', 'amp', 'get', 'blockchain') are removed from the data.

3.2.2 Text Clustering Model

With all the input data tokenized, the word tokens are fed into the process of creating TF-IDF matrix. After which, the matrix is forwarded into the Clustering model.

The clustering model is using KMeans. After trial and error, the number of clusters is defined as 5. The first cluster contains key words of 5 clusters are as below.

```
Cluster 0: btc ethereum eth binance buy bnb doge dogecoin market money
Cluster 1: btc price usd moon current last unknown hours btcusd hour
Cluster 2: stakes heco chain invite distribution entry referral position date signals
Cluster 3: project airdrop bsc good airdrops bnb great defi team future
Cluster 4: coinhuntworld vault location playing found awesome user join blue green
```

The sample distribution within 5 clusters is: 1st cluster with 947114 samples, 2nd cluster with 110883 samples, 3rd cluster with 24111 samples, 4th cluster with 140493 samples and 5th cluster with 10204 samples.



Some sample clustering with labels is shown below:

	date	text	label
0	2021-02-10 23:59:04	blue ridge bank shares halted by nyse after bitcoin atm announcement	1
1	2021-02-10 23:58:48	today that's this thursday we will do take with our friend btc wallet security expe	1
2	2021-02-10 23:54:48	guys evening have read this article about btc and would like to share with you all	1
3	2021-02-10 23:54:33	big chance in billion price bitcoin fx btc crypto	2
4	2021-02-10 23:54:06	this network is secured by nodes as of today soon the biggest bears will recognise btc in too big to fail	1
...
95	2021-02-10 23:10:02	bitcoin btc current price hour hours days btc bitcoin	2
96	2021-02-10 23:09:59	saylor advertising tactic called linking trying to get people to link bitcoin with buffet before it wa	1
97	2021-02-10 23:09:39	prices update in hour	1
98	2021-02-10 23:09:18	just added episode insane eth genesis land sale by axie chat defi btc eth bitcoin crypto	1
99	2021-02-10 23:08:14	is looking super amazing should easily do bitcoin btc dogecoin	1

3.3 Topic Modeling

For topic Modeling, the preprocessing steps are as same as Text Clustering, except it is with different stop word list ('crypto', 'cryptocurrency', 'amp', 'get', 'blockchain'). Within the dictionary, the system filtered off any words with document frequency less than 2, or appearing in more than 90% documents.

For the LDA Model, number of topics is set as 5 as well. The distribution of 5 topics is 1st topic with 293406 samples, 2nd topic with 99191 samples, 3rd topic with 277432 samples, 4th topic with 436166 samples and 5th topic with 126610 samples.

Key words of 5 topics are as:

```
[(),  
 '0.129*"bitcoin" + 0.012*"n\t" + 0.009*"people" + 0.008*"world" + 0.007*"first" + 0.007*"need" + 0.006*"bank" + 0.  
006**"fix" + 0.006*"salvador" + 0.005**"work"),  
(1,  
 '0.112*"airdrop" + 0.066*"bsc" + 0.039*"btc" + 0.027*"binancesmartchain" + 0.023*"airdropinspector" + 0.022**"one" +  
0.021**"chain" + 0.020**"join" + 0.019**"stake" + 0.017**"heco"),  
(2,  
 '0.066**"project" + 0.055**"bitcoin" + 0.018**"good" + 0.017**"future" + 0.017**"great" + 0.015**"nft" + 0.015**"team" +  
0.013**"money" + 0.013**"like" + 0.012**"petg"),  
(3,  
 '0.068**"btc" + 0.056**"bitcoin" + 0.033**"ethereum" + 0.025**"price" + 0.025**"bnb" + 0.024**"binance" + 0.024**"eth" +  
0.015**"market" + 0.015**"trading" + 0.015**"buy"),  
(4,  
 '0.077**"bitcoin" + 0.018**"coin" + 0.015**"see" + 0.015**"every" + 0.014**"new" + 0.013**"year" + 0.012**"always" + 0.012  
**"news" + 0.012**"think" + 0.011**"let"))]
```



Some sample topic modeling outcome with labels are seen below:

	date	text	label
0	2021-02-10 23:59:04	blue ridge bank shares halted by nyse after bi...	5
1	2021-02-10 23:58:48	today that's this thursday we will do take wit...	2
2	2021-02-10 23:54:48	guys evening have read this article about btc ...	3
3	2021-02-10 23:54:33	big chance in billion price bitcoin fx btc crypto	2
4	2021-02-10 23:54:06	this network is secured by nodes as of today s...	2
...
95	2021-02-10 23:10:02	bitcoin btc current price hour hours days btc ...	4
96	2021-02-10 23:09:59	saylor advertising tactic called linking tryin...	1
97	2021-02-10 23:09:39	prices update in hour	4
98	2021-02-10 23:09:18	just added episode insane eth genesis land sal...	4
99	2021-02-10 23:08:14	is looking super amazing should easily do bitc...	5

3.4 Sentiment Mining

The Natural Language Processing tasks have been significantly improved when the state of art pre-trained model BERT (Bidirectional Encoder Representations from Transformers) was proposed by researcher at Google since 2018 that is a bidirectional transformer pre-trained using a combination of masked language modeling objective and next sentence prediction on a large corpus comprising the Wikipedia and Toronto Book Corpus.

In our system, we use distilled version BERT (DistilBERT) as sentiment mining tool. The numerical features from output of DistilBERT that will be used as one of feature columns for subsequent crypto sentiment price prediction. DistilBERT has 40% less parameters than bert-base-uncased, runs 60% faster while preserving over 95% of BERT's performances as measured on the GLUE language understanding benchmark. It is more suitable for our use case as it trains on unlabeled text corpus.

The following training process steps are used for feature extraction:

1. Load text data: 1.2m tweets are loaded into our feature extraction program.



2. Text preprocessing: as the tweets may contains those irrelevant ticker symbols, URLs and hashtags. We have performed the common text-preprocessing as
 - a. Lowercasing text
 - b. Removing URLs
 - c. Removing username and hashtags
 - d. Removing special characters
3. Load DistilBERT tokenizer: The distilBERT model is loaded with pre-trained model file distilbert-base-uncased as well as below steps:
 - a. Calculate maximum length of text by iterating each tweet
 - b. Tokenize each tweet
 - c. Add padding to those length of tweets is less than maximum
 - d. Create attention mask to ignore those positions with padding zero
4. Run DistilBERT training:
 - a. Loop the training data with batch size 1000 and pass in corresponding padded input ids and attention masks
 - b. Get first vector in last hidden state from each batch and ignore all others ([CLS] token)
5. Save feature vectors to csv
 - a. Save the feature vector to csv file as we use it as input to subsequent crypto sentiment price prediction model



4.0 Features integration and training

Each of the 4 language processing models features score are aggregated into a single row representing the daily feature scores as follows:

Text clustering

Datetime	c1	c2	c3	c4	c5
2021-02-05	0.853601	0.143447	0.000000	0.002952	0.000000
2021-02-06	0.841929	0.148917	0.000000	0.009155	0.000000
2021-02-07	0.865962	0.129416	0.000000	0.004622	0.000000
2021-02-08	0.873893	0.122565	0.000000	0.003542	0.000000
2021-02-09	0.865747	0.129655	0.000000	0.004598	0.000000
...
2021-08-23	0.835000	0.054556	0.012187	0.084452	0.013806
2021-08-24	0.770489	0.056532	0.011819	0.151793	0.009366
2021-08-25	0.605727	0.049765	0.125741	0.213835	0.004932
2021-08-26	0.717127	0.071874	0.034520	0.171424	0.005055
2021-09-10	0.801642	0.060307	0.008591	0.120444	0.009016

Topic modelling

Datetime	t1	t2	t3	t4	t5
2021-02-05	0.243211	0.063164	0.149351	0.415584	0.128689
2021-02-06	0.195911	0.057370	0.133659	0.444919	0.168142
2021-02-07	0.190492	0.060416	0.121162	0.427204	0.200726
2021-02-08	0.178888	0.059157	0.139214	0.406305	0.216436
2021-02-09	0.196782	0.065517	0.133103	0.415632	0.188966
...
2021-08-23	0.281539	0.074074	0.230601	0.307436	0.106351
2021-08-24	0.257015	0.078721	0.255566	0.282141	0.126556
2021-08-25	0.182628	0.201059	0.271954	0.267788	0.076572
2021-08-26	0.196522	0.106751	0.269188	0.338048	0.089491
2021-09-10	0.310764	0.055331	0.261472	0.270701	0.101731



Text Classification Score (Vader)

Datetime	score_mean	score_median	score_min	score_max
2021-02-05	0.14992585596221972	0.0	-0.9225	0.9186
2021-02-06	0.1573604026845636	0.0	-0.8979	0.9571
2021-02-07	0.16986075907590706	0.0	-0.9524	0.9545
...
2021-09-10	0.28659397354655824	0.2732	-0.9918	0.9905

Sentiment Analysis (DistilBert)

Datetime	bert_mean	bert_median	bert_min	bert_max
2021-02-05	-0.12871159409327035	-0.13060267	-0.5050843	0.34367937
2021-02-06	-0.13639183024252755	-0.1401386	-0.60410464	0.45204908
2021-02-07	-0.13300393550891146	-0.1401386	-0.62808716	0.42128685
...
2021-08-17	-0.12570059894734673	-0.12891066	-0.73567265	0.41971326
2021-08-18	-0.1259489952350499	-0.12910349999999998	-0.665828	0.6135774
2021-08-19	-0.1257721236067639	-0.12923138	-0.67615247	0.5162844
2021-08-20	-0.125082522313123	-0.12829626	-0.9406885	0.49151444
2021-08-21	-0.13471741014778552	-0.13192567	-0.7363487	0.4151647

As the Text and Sentiment Analysis scores return single value per tweet, they were aggregated using mean, median, min and max after being grouped by date.

These 4 feature sets and the day+3 price change labels are combined using the date as index to arrive at the final feature set ready for training.



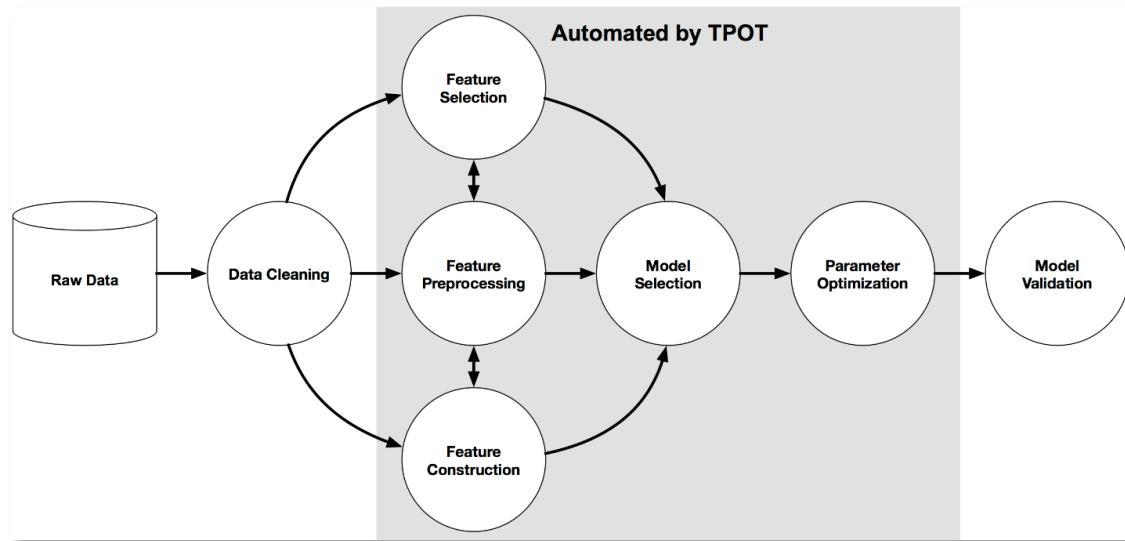
Combined Features Set for Training

	open	high	low	close	Volume USDT	price_change_label	c1	c2	c3	c4	...	t4	t5
date													
2021-02-05	36936.65	38310.12	36570.00	38290.24	2.509278e+09		1.0	0.853601	0.143447	0.000000	0.002952	...	0.415584 0.128689
2021-02-06	38289.32	40955.51	38215.94	39186.94	3.922095e+09		1.0	0.841929	0.148917	0.000000	0.009155	...	0.444919 0.168142
2021-02-07	39181.01	39700.00	37351.00	38795.69	3.256521e+09		1.0	0.865962	0.129416	0.000000	0.004622	...	0.427204 0.200726
2021-02-08	38795.69	46794.45	37988.89	46374.87	5.881537e+09		1.0	0.873893	0.122565	0.000000	0.003542	...	0.406305 0.216436
2021-02-09	46374.86	48142.19	44961.09	46420.42	5.386255e+09		1.0	0.865747	0.129655	0.000000	0.004598	...	0.415632 0.188966

bert_mean	bert_median	bert_min	bert_max	score_mean	score_median	score_min	score_max
-0.12871159409327035	-0.13060267	-0.5050843	0.34367937	0.14992585596221972		0.0	-0.9225 0.9186
-0.13639183024252755	-0.1401386	-0.60410464	0.45204908	0.1573604026845636		0.0	-0.8979 0.9571
-0.13300393550891146	-0.1401386	-0.62808716	0.42128685	0.16986075907590706		0.0	-0.9524 0.9545
-0.1280233506341424	-0.13286781	-0.60410464	0.38223177	0.18379461661059224		0.0	-0.9442 0.9678
-0.12788942554735708	-0.13189440000000002	-0.60410464	0.33967122	0.17676144827586338		0.0	-0.8689 0.9678



Using TPOT to find the best performing ML model with Genetic Programming



TPOT is a Python Automated Machine Learning tool that optimizes machine learning pipelines using genetic programming. TPOT will automate the most tedious part of machine learning by intelligently exploring thousands of possible pipelines to find the best one for the data provided.

```
Generation 1 - Current best internal CV score: 0.8072727272727273
Generation 2 - Current best internal CV score: 0.8072727272727273
Generation 3 - Current best internal CV score: 0.8272727272727272
Generation 4 - Current best internal CV score: 0.8272727272727272
Generation 5 - Current best internal CV score: 0.8272727272727272
Generation 6 - Current best internal CV score: 0.8272727272727272
Generation 7 - Current best internal CV score: 0.8272727272727272
Generation 8 - Current best internal CV score: 0.8272727272727272
Generation 9 - Current best internal CV score: 0.8272727272727272
Generation 10 - Current best internal CV score: 0.8272727272727274

Best pipeline: DecisionTreeClassifier(RandomForestClassifier(input_matrix, bootstrap=False, criterion=entropy, max_features=8, min_samples_leaf=17, min_samples_split=4, n_estimators=100), criterion=gini, max_depth=8, min_samples_leaf=18, min_samples_split=2)
```

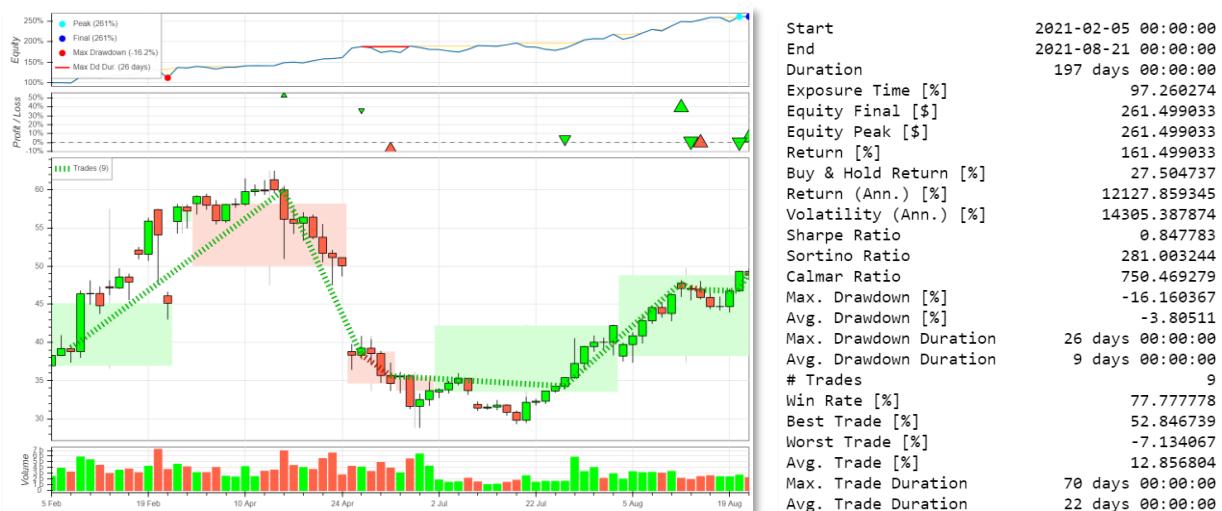
With the complete features set, we used the TPOT library to help us find the best performing ML model for our data. As seen above, over 10 generations, it found that a Decision Tree model works best with a cross validation score of 0.827.



5.0 Back testing

Using a python back testing library, we simulated how a portfolio of \$1000 will perform over the period of twitter data using the trade signals generated by the Decision Tree prediction model.

The trading rules are based on the day+3 price prediction. If the prediction is positive, the simulator will buy Bitcoin with all its funds at the current price. When the prediction is negative, the simulator will immediately sell off all Bitcoin at the prevailing market price.



As visualized in the generated graph above, the portfolio ran through historical simulation from February to August 2021 and performed a total of 9 trades. The final equity is 261% which means a profit of 161%. Compared with the buy and hold strategy that returns a 27% profit, the trade signals generated by the prediction outperformed by a factor of 5.8 times.



6.0 Findings and conclusion

From the evaluation of system performance above, we observe 2 findings as listed down below.

- Reacting to market sentiment and trading in and out of Bitcoin vastly outperforms a simple buy and hold strategy.
- Due to the volatility of the market, it pays off to be cautious by selling Bitcoin even when market sentiment is slightly poor. This helps to avoid heavy losses and preserve portfolio capital for suitable buying opportunities.

Our project still has much room for improvement, particularly so for gaps in our twitter data and optimizing our Text Analytics feature extraction methods. This sentiment model can also be used as part of a more complex model that operates on price technical analysis to become even more powerful in producing features for trade signals generation.

In conclusion, we are glad to be able to apply most of the text analytics and sentiment mining techniques covered in our course. We have come to appreciate the challenge in developing an accurate sentiment analysis model and the value it will bring forth.



References

Bitcoin tweets (1.18M records / 518 MB)

<https://www.kaggle.com/kaushiksuresh147/bitcoin-tweets>

Python-Binance (Binance API)

<https://github.com/sammchardy/python-binance>

TPOT library

<http://epistasislab.github.io/tpot/>

Backtesting library

<https://kernc.github.io/backtesting.py/>

Vader Sentiment

<https://github.com/cjhutto/vaderSentiment>

<https://ojs.aaai.org/index.php/ICWSM/article/view/14550/14399>

BERT

<https://arxiv.org/pdf/1810.04805.pdf>

