Predicting bitcoin trend change using tweets

Jakob Jelencic
Artificial Intelligence Laboratory
Jozef Stefan Institute and Jozef International Postgraduate School
Ljubljana, Slovenia
jakob.jelencic@ijs.si

ABSTRACT

Predicting future is hard and challenging task. Predicting financial derivative that one can benefit from is even more challenging. The idea of this work is to use information contained in tweets data-set combined with standard Open-High-Low-Close [OHLC] data-set for trend prediction of crypto-currency Bitcoin [XBT] in time period from 2019-10-01 to 2020-05-01. A lot of emphasis is put on text preprocessing, which is then followed by deep learning models and concluded with analysis of underlying embedding. Results were not as promising as one might hope for, but they present a good starting point for future work.

1. INTRODUCTION

Twitter is an American microblogging and social networking service on which users post and interact with messages known as "tweets". Registered users can post, like, and retweet tweets, but unregistered users can only read them. Users access Twitter through its website interface, through Short Message Service (SMS) or its mobile-device application software. Tweets were originally restricted to 140 characters, but was doubled to 280 for non-CJK languages in November 2017. People might post a message for a wide range of reasons, such as to state someone's mood in a moment, to advertise one's business, to comment on current events, or to report an accident or disaster [5].

Bitcoin is a cryptocurrency. It is a decentralized digital currency without a central bank or single administrator that can be sent from user to user on the peer-to-peer bitcoin network without the need for intermediaries. Bitcoin is known for its unpredictable price movements, sometimes even to 10% on the daily basis. Bitcoin also serve as an underlying asset for various financial derivatives, which means that one can profit from knowing the future price changes.

Tweets data offer a constant stream of new information about people beliefs about Bitcoin. Since Bitcoin is very volatile asset, without any real-world value, its value is mainly driven by people's trust in it. Which means that possible up or down trends could be predicted by understanding sentiment of people tweets related to Bitcoin and other cryptocurrencies. Tweets data-set is combined with classical Open-High-Low-Close [OHLC] data-set for 5 minute time periods. OHLC data-set contain information about opening and closing price of given time period, its maximum and minimum price during observed time period and sum of volume and number of transactions made [4]. This present additional information how the market is behaving at any given point.

In financial mathematics derivatives are usually modeled with some kind of stochastic process. Most commonly some form of Brownian motion is used. In theory increment in Brownian motion is distributed as $N(\mu, \Sigma)$ independent from previous increment. This implies that prediction of a real time price change of a derivative is not possible, so the target goal should be changed accordingly. Instead of predicting the impossible, the goal of this work is to predict a change in a trend. Trend is calculated with exponential moving average, application of it can be observed in Figure 1.

Definition: Exponential moving average:

$$EMA(TS, n) = \alpha \cdot (\sum_{i=0}^{n-1} (1-\alpha)^i \, TS_{n-i}),$$

$$\alpha = \frac{2}{n+1}.$$

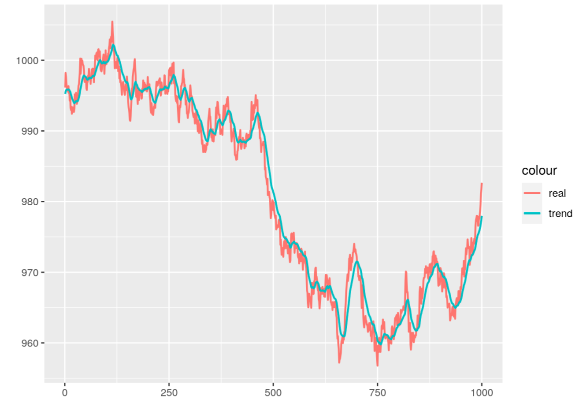
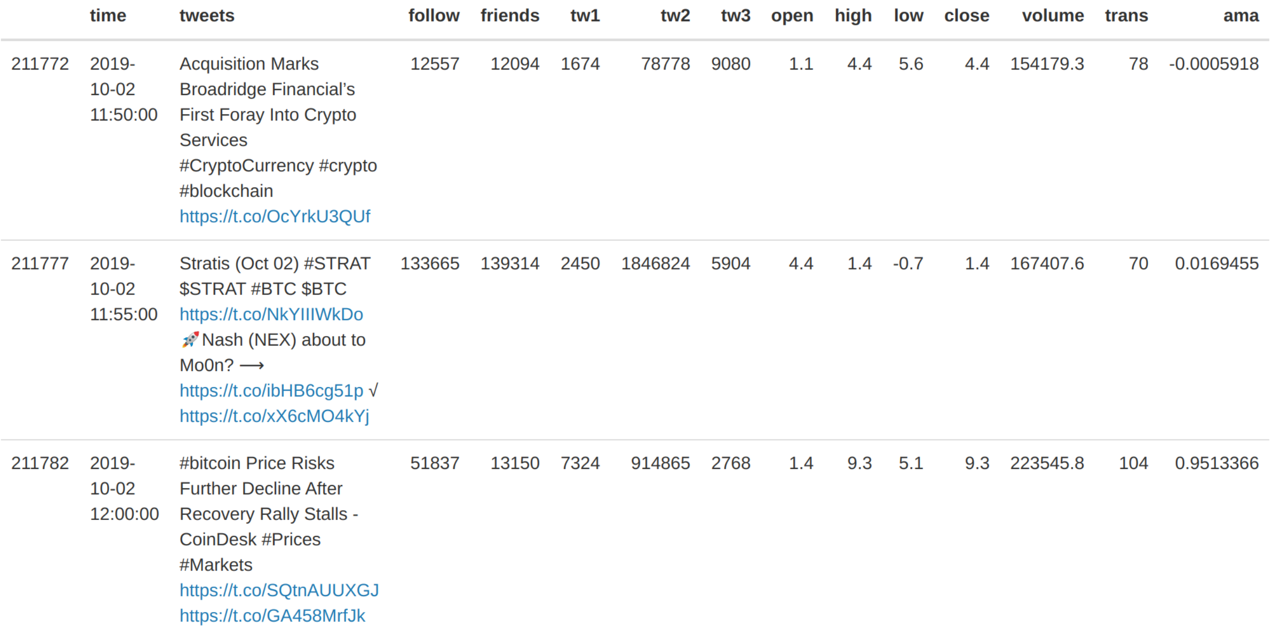


Figure 1: Example of exponential moving average



	time	tweets	follow	friends	tw1	tw2	tw3	open	high	low	close	volume	trans	ama
211772	2019-10-02 11:50:00	Acquisition Marks Broadridge Financial's First Foray Into Crypto Services #CryptoCurrency #crypto #blockchain https://t.co/OcYrkU3QUf	12557	12094	1674	78778	9080	1.1	4.4	5.6	4.4	154179.3	78	-0.0005918
211777	2019-10-02 11:55:00	Stratis (Oct 02) #STRAT $STRAT #BTC $BTC https://t.co/NkYIIIWkDo 🚀Nash (NEX) about to Mo0n? ⟶ https://t.co/ibHB6cg51p √ https://t.co/xX6cMO4kYj	133665	139314	2450	1846824	5904	4.4	1.4	-0.7	1.4	167407.6	70	0.0169455
211782	2019-10-02 12:00:00	#bitcoin Price Risks Further Decline After Recovery Rally Stalls - CoinDesk #Prices #Markets https://t.co/SQtnAUUXGJ https://t.co/GA458MrfJk	51837	13150	7324	914865	2768	1.4	9.3	5.1	9.3	223545.8	104	0.9513366

Figure 2: Example of working dataset.

2. DATA DESCRIPTION

Collected tweets range from 01-10-2019 to 01-05-2020. We have filtered tweets by crypto-related hashtags. Originally tweets contained multilingual data, but only English one were extracted. Data-set still resulted in more than 5 000 000 tweets over a little more than a half year period. Dealing with such big data-set has proven to be too difficult of a task. But since a lot of tweets are just pure noise, this data-set can be reduced. Idea is to extract the tweets with the largest target audience. Since the data-set contain number of tweet's author friends and followers, we have extracted the tweets with maximum sum of both in a 5 minute period. Unfortunately, crypto world is relatively anonymous, so there is no Warren Buffet alike personalty, to whom we could gave extra weight.

Then we concatenated the reduced tweets with 5-minute OHLC data-set. Snapshot can be observed in Figure 2. Column names should be pretty self-explanatory, expect for "tw1","tw2","tw3", which stands for metadata information about tweets and "ama", which stand for current movement of trend. Continuous features are then normalized, "ama" is shifted one step into the future so it forms the target variable. Regression task has the most success with predictions.

3. TWEETS PROCESSING

Aim of this chapter is to focus on processing tweets. Tweets differ from regular text data, since many of them consist hyperlink, hashtags, abbreviations, grammar mistakes and so on. This excludes any pre-build preprocessing tools, like the one available in deep learning library Tensorflow [1] which is used for building deep learning models. In the Figure 2 we can see an example of some tweets. The cleaning process was executed in the same order as it is stated below. For each tweet the following process was executed:

- Escape characters were removed.
- Tweet was split by " ".
- All non alphanumeric characters were removed, including "#".
- All characters were converted to lower case.
- Usual stop-words were removed.

At this point data-set contain over 200000 different tokens, which is way to sparse for so limited data-set. At this point empirical cumulative distribution function was calculated and all tokens that have less than 50 appearances were removed. The dictionary size is now 2150.

Another thing to consider is how to process numbers that appear in between text. Obviously a separate token for each number is not acceptable, since it would negate all the work it was done so far. The following function was applied to process numbers. 5 more tokens were created and then numbers from a certain interval were assigned corresponding token.

- Small number: $X < 1000$.
- Medium number: $X \in [1000, 10000)$.
- Semi big number: $X \in [10000, 100000)$.
- Big number: $X \in [100000, 1000000)$.
- Huge number: $X \geq 1000000$.

Additional masking token were assigned for missing data. This wrap up dictionary, final length of dictionary is 2156.

Last thing in processing tweets is to handle their length. Not all tweets have the same length. One idea is to take the maximum length of all tweets, then mask the others so they all have the same length. Unfortunately this would take a lot of unnecessary space, which is a problem. Also long tweets does not mean informative tweet. In Figure 3 is plotted the empirical cumulative distribution function of tweets' length.

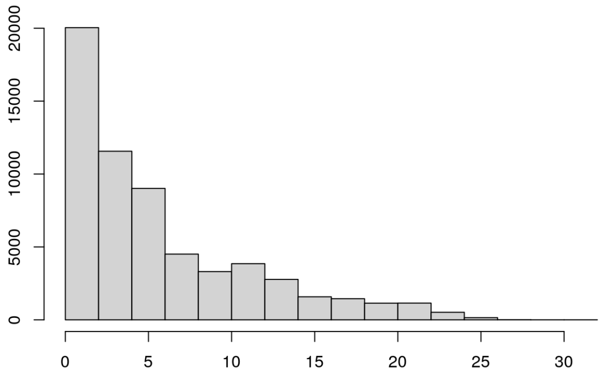


Figure 3: Histogram of tweets' length.

No additional manipulation of tokens were done. It is known that tokens "bitcoin" and "btc" means the same, and they could be join into one token, but they are left intact and the deep learning model will decide either they are the same or not.

4. DEEP LEARNING MODELS

Obvious choice for text models are recurrent neural networks, more specifically Long-Short-term-Memory [LSTM] recurrent networks [2]. They are usually combined with embedding layers, which transform singular token to vector of arbitrary size [6].

Since the task at hand is predicting the future, there is no good benchmark metric or model which could serve as a threshold for our model performance. So in order to see if the tweets can contribute anything, we have decided to build a shallow neural network of just OHLC data which would serve as a benchmark model. 80% of the data-set was taken as a training set, remaining was left out for validation. Split was the same in both models. Both time we used Adam optimizer [3] and mean-squared error [MSE] as a loss function. Training was stopped as soon as validation loss did not improve for 10 epochs. Batch size was 256.

Structure of a benchmark model:

- Input dense layer with 32 neurons.
- Stacked dense layer with 32 neurons.
- Stacked dense layer with 32 neurons.
- Output dense layer with 1 neuron.

Structure of a tweets model:

- Input embedding layer of size 64 (tweets).
- Stacked LSTM layer with 128 neurons.
- Stacked LSTM layer with 128 neurons.
- Second input layer with 64 neurons (OHLC).
- Concatenation.
- Stacked dense layer with 64 neurons.
- Output dense layer with 1 neuron.

Loss process of benchmark model can be observed in Figure 4, while loss process of tweets model can be observed in Figure 5. Orange color represent training set, while blue validation set. It is clear that the tweets model behaved a lot worse on training set than benchmark model, but on test set it has slightly lower MSE (benchmark: 13.78, tweets: 13.74). This implies that there is a lot of reserve in fitting of the tweets model, since the difference between the train and validation loss is so big. That is good since otherwise it seems that tweets do not contribute much for prediction. It is also worth noting that tweets model took way longer to learn, around 380 epochs compared to benchmark's model 40.

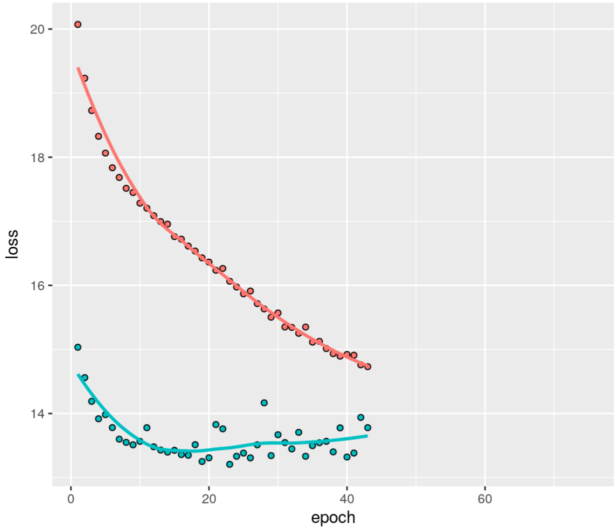


Figure 4: Loss process of benchmark model.

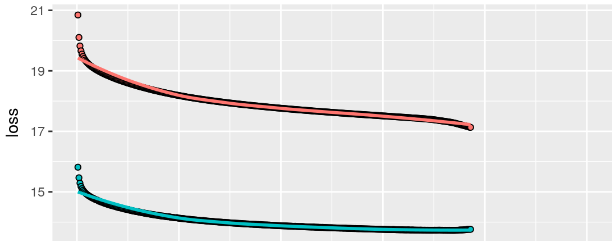


Figure 5: Loss process of tweets model.

5. ANALYSIS OF UNDERLYING EMBEDDING MATRIX

We have extracted underlying embedding matrix from tweets model. Since the model tried to minimize mean-squared error

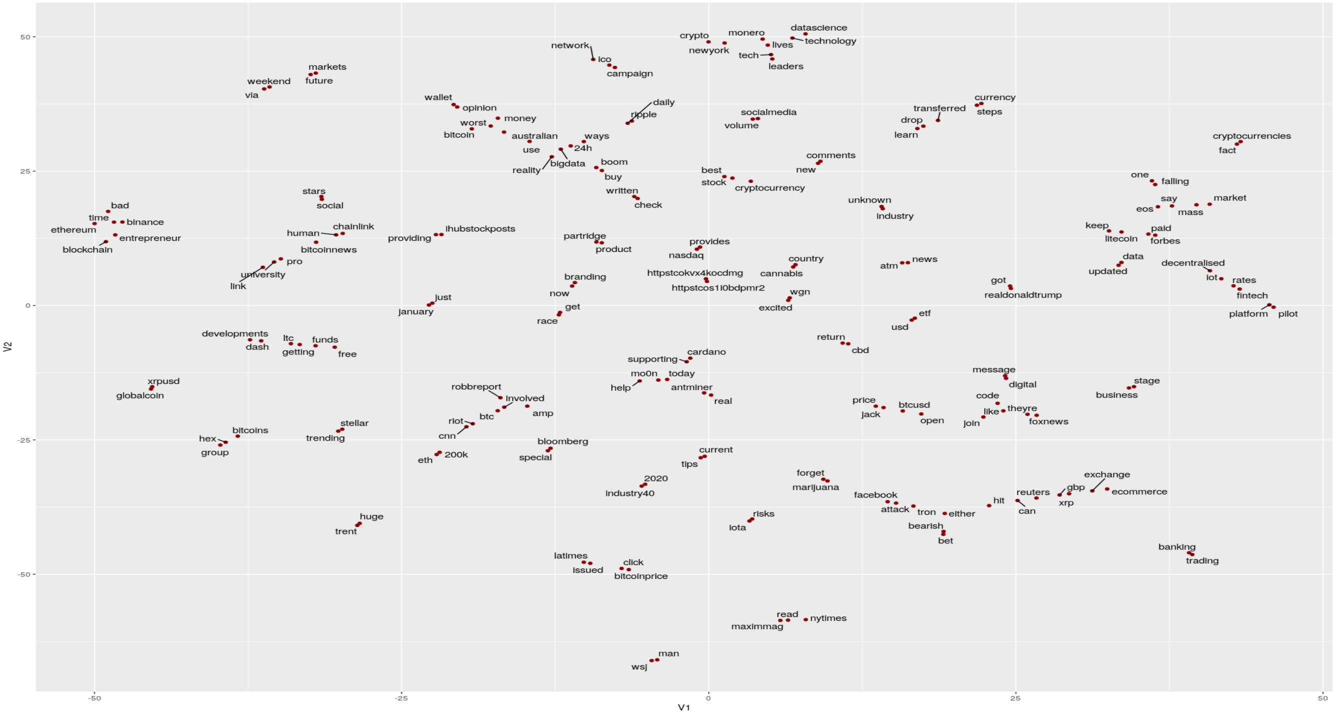


Figure 6: TSNE projection of embedding matrix.

[MSE] of predicted trend and actual trend, the embedding matrix accordingly to MSE derivative. For analysis we will use cosine similarity as a metric. If 2 words are close in the embedding matrix, this does not mean that they are semantically similar in concept of everyday language, but it means that they are similar in concept of Bitcoin trend prediction. For example if model converged perfectly, and tokens "bitcoin" and "eth" have cosine similarity near 1, that would mean that they both have similar impact on Bitcoin trend. Which is not so hard to believe since it is known that all crypto-currencies are heavily correlated with one another. On Table 1 it can be seen cosine similarity of some of the most common tokens in the dictionary.

Table 1: Cosine similarity pairs of most common tokens.

Tokens Pair	Similarity
bitcoin, crypto	0.472
blockchain, entrepreneur	0.561
crypto, cryptocurrency	0.519
cryptocurrency, blockchain	0.560
volume, social media	0.508
ethereum, blockchain	0.557

We cannot be completely satisfied with results, but for such limited data-set they are not that bad. As it is with any embedding evaluation, it comes to certain amount of subjectivity what is good and what is not.

In order to gain the better perspective of obtained embedding we did a T-distributed stochastic neighbor embedding projection to 2 dimension and plotted 100 nearest pairs. Projection can be observed in Figure 6.

6. CONCLUSION

While the obtained model cannot be served as production model for automatic trading, it presents a nice future work opportunity. We will continue to collect tweets, and hopefully with time build a more accurate data-set and with some hyper-tuning of tweets models achieve improved prediction.

7. ACKNOWLEDGMENTS

This work was financially supported by the Slovenian Research Agency.

8. REFERENCES

[1] *TensorFlow*. https://www.tensorflow.org/.

[2] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[3] D. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2014. https://arxiv.org/abs/1412.6980.

[4] J. J. Murphy. *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance Series. New York Institute of Finance, 1999.

[5] R. Nugroho, C. Paris, S. Nepal, J. Yang, and W. Zhao. A survey of recent methods on deriving topics from twitter: algorithm to evaluation. *Knowledge and Information Systems*, pages 1–35, 2020.

[6] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Series in Artificial Intelligence. Prentice Hall, Upper Saddle River, NJ, third edition, 2010.