Crypto Exchange Price Prediction using Limit Order Book

Ashwin S. Padmanabhan, Ben Gilboa, Tamal Biswas

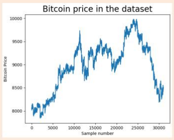
CS230 Project

Introduction

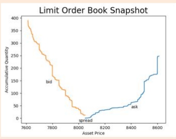
Explore and develop a deep machine learning model that predicts the future price of digital asset such as bitcoin. We intend to build a machine learning RNN (Recurrent Neural Network) that predicts the future price of a tradable and volatile digital asset such as the Bitcoin. The input to the model will be a limit order book data along with other historical indicators for demand and supply to develop our predictor. Although we chose a digital asset for this project, the principals and methods we develop are transferable to any asset that is tradable in an exchange

Dataset Characteristics and Acquisition

Bitcoin historical price



Limit order book



The limit order book snapshot represent the demand and supply in the market in a certain point in time. In the above figure, it is observed that the demand is "stronger" indicating that the price is about to increase. We look at the 500 highest bid orders and the 500 lowest ask orders in every snapshot of the order book

Along with limit order book we look at the corresponding bitcoin price which is the "last" price of a transaction at the same time when the order book was sampled. This data will serve both as features in the training examples as well as in generating the classifier for price increase or decrease

Data Acquisition

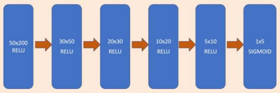
Sampling the Bittrex exchange every 1 minute using the API it provides. We have obtained 20,000 samples that represent 2 weeks worth of trading data.

Sample raw Sell Order Book data from API -
https://bittrex.com/api/v1.1/public/getorderbook?market=USDT-BTC&type=sell

Sample raw Buy Order Book data from the API -
https://bittrex.com/api/v1.1/public/getorderbook?market=USDT-BTC&type=buy

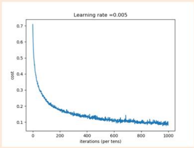
Initial Model and Results

Fully Connected Network Architecture

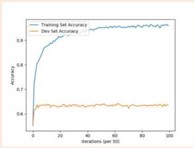


Key objective of this phase was to find correlation and validate the data from the order book as valid predictor

Cost Plot



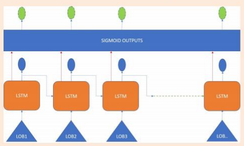
Training vs Dev Accuracy



21,000 training samples with training / dev sets as 80%/20% split. Compared bitcoin price 1min, 2min, 3min, 5min and 10min into the future to the current price. Achieved approximately 95% accuracy on the training set and approximately 64% on the dev set. Max dev accuracy at around 3100 epochs)

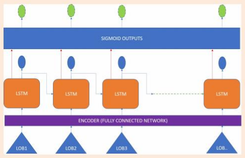
Recurrent Neural Networks

RNN Architecture with Inputs and Outputs same as Fully Connected Network



Limit order book (LOB) input into an LSTM network followed by a sigmoid output prediction of an increase or a decrease like the Fully Connected Network. Adding L2 regularization, we achieved almost a 10% increase in dev accuracy, but the training accuracy fell close to the dev accuracy

RNN with Fully Connected Network acting as encoder becomes RNN input



Limit order book data encoded using a Fully Connected Network and the activations fed from the last but one layer into the RNN. Performance similar to the previous RNN where the order book was directly fed as input

RNN with Single Limit Order Book spliced into multiple time steps

Input split into equal number of parts and fed to a time step in the LSTM network. 200 feature inputs split into 10 parts of 20 each and fed to the LSTM network with 20 time steps. Achieved 95% plus accuracy on the training set and 66% on the dev set (a 3% increase compared to the FC network)

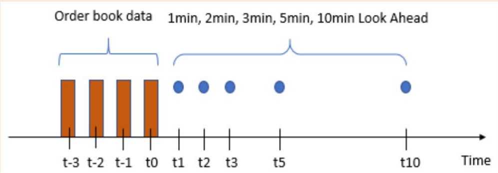
Categorical Model

Enhance output labels to 3 categories:

- Increase by more than threshold percent
- Decrease by more than threshold percent
- Did not change by more than threshold percent

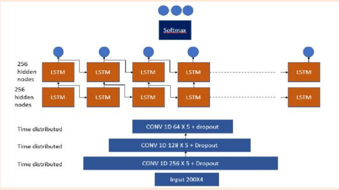
Signal created with 4 dimensions so we can tune and find the best option. One dimension is the threshold (0.1%, 0.2%, 0.3%, 0.4%, 0.5%), second dimension is the future look ahead prediction (1min, 2min, 3min, 5min, 10min), other 2 dimensions are for the RNN time step (window size) and the batch.

Categorical Model input / output preparation



Best results achieved for 2 min look ahead prediction and 0.2% threshold change

Categorical Model Architecture



Best result with 3 convolutional layers (1 dimensional) connected to 2 layers of LSTM RNN with 256 hidden nodes each and one softmax layer with 3 states for the output

25,000 samples label distribution

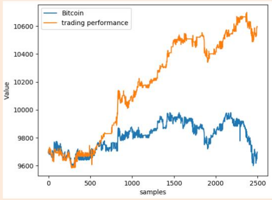
	Increase	No change	Decrease
Training Set	3126	16310	3060
Dev Set	278	1928	290

Categorical Model Results

	Increase	No change	Decrease
Ground truth	278	1928	290
Predicted increase	87	247	17
Predicted no change	181	1505	196
Predicted decrease	10	176	77

Using the network above, we achieved 70% accuracy for the training set and 67% accuracy for the dev set. If we consider the no change label as kind of "don't care" than the prediction becomes much better. Meaning for prediction of increase, there are 25% of actual increase, 70% of no change and only 5% of decrease

Prediction Simulation



To test the model we have built a simple trading algorithm that starts with value of one bitcoin. It alternates between bitcoin and dollars based on the prediction considering the current price for the conversions. We see that this trading algorithm outperform the bitcoin itself for the tie interval of our dev set achieving over 9% compared to 0%

Conclusion

We clearly establish correlation between limit order book and future price.

We saw that RNN networks perform better that fully connected networks for this kind of problem.

Using categorical model with threshold, modifying the loss function and adding convolutional layers we were able to create a predictor that outperform the bitcoin asset performance

References

[1] Tian Guo, NinoAntulov-Fantulin. Predicting short-term Bitcoin price fluctuations from buy and sell orders

[2] Huisu Jang, Jaewook Lee. An Empirical Study on Modeling and Prediction of Bitcoin prices with Bayesian Neural Networks Based on Blockchain information

[3] Muhammad J Amjad, Devarat Shah. Trading Bitcoinand Online Time Series Protection

[4] N.I. Indera, I.M.Yassinm A.Zabidi, Z.I.Rizman. Non-Linear AutoRegressive with Exegeneous Input (NARX) Bitcoin price prediction model using PSO-Optimized parameters and moving average technical indicators

[5] Justin A Sirignano. Deep Learning for Limit Order Books

Link to poster video/audio presentation in YouTube:

https://www.youtube.com/watch?v=SGRJ9Iu0RxI