Capstone Report

Time Series Forecasting Using Sequence to Sequence Model For The Value of Bitcoin

Parinaz Farajiparvar

May 2020

1 Definition

1.1 Domain Background

In time series prediction, the goal is to predict the future values based on past observation which consists of intrinsic patterns. To determine a model that expresses the pattern of the time series, we need a model to describe the important features of the time series pattern and explain how the past affects the future or how two-time series can "interact". In [2], different deep learning methods to predict the future prices of Bitcoin mentioned. He developed a Recurrent Neural Network with a block of LSTM to do this task. In this project, we want to predict the future value of Bitcoin using Sequence to Sequence model (Seq2Seq) which is introduced in [3] as a learning method for natural language processing.

1.2 Problem Statement

The goal of this project is to predict the future price of Bitcoin for one week based on Bitcoin's historical price data. In this project I develop Seq2Seq and Stack LSTM for this task, and compare them in terms of accuracy. Based on my result, Seq2Seq has a better performance and could have a average MAPE=1

2 Analysis

2.1 Datasets and Inputs

The historical data for the USD and EUR bitcoin value, from `http://api.coindesk.com/v1/bpi/historical/close.json?start=2010-07-17&end=2017-03-03`, as it is shown in Fig. 1 we use the more recent data around 1400 time steps (each day).

2.1.1 Data Specification

I used both the value of Bitcoin in terms of dollar and euro and concatenate them create a data set with 2-dimension, we can add more dimension to this data to improve the result by have more information about the economy and word in general.
I used the data from '2013-04-13' to '2017-03-03', so in total I have 1421 samples of data for both USD and EUR. This is the example of the data: [('2013-04-13', 93), ('2013-04-14', 90, 70.9501), ('2013-04-15', 82.386, 68.6995), ('2013-04-16', 68.3557, 62.8801), ('2013-04-17', 93.07, 52.3838), ('2013-04-18', 109.01, 70.6092), ('2013-04-19', 118.48, 83.3899), ('2013-04-20', 126.6155, 90.8012), ('2013-04-21', 119.2, 96.999), ('2013-04-22',127.4, 91.3043)], which the first item in each tuple is time step and second one is the value of this item in USD and third one is the value of Bitcoin in term of EUR.
In order to prepare the data for learning process with neural network, we divide it into train 80%, validation 10% and test set 10% of data. For each part then we will window the data as a past and future part like the thing that we did in the time series assignment in the course.

2.2 Algorithm and Techniques

A Seq2Seq model was first introduced in [3] as a learning method for natural language processing. This translation application is like time series prediction because it has a sequential model and it also has sequential input-output pairs. In their method, they use a multi-layered LSTM to map the input sequence to a vector with a fixed dimension. Then this vector is used by another deep LSTM to decode the target sequence from that vector. The sequence to sequence model consists of two blocks of LSTMs, which are incorporated into an encoder and decoder block as shown in Fig 2. Their

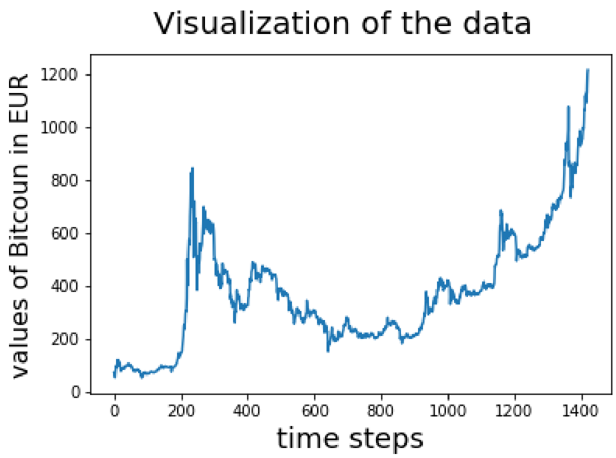

Figure 1: Visualization of the data, Bitcoin value in terms of USD

main result was that on an English to French translation tasks from the WMT-14 dataset produced a good result in comparison with other methods. Seq2Seq models are also used extensively for the dynamic, spatial-temporal characteristic of multivariate time series date.

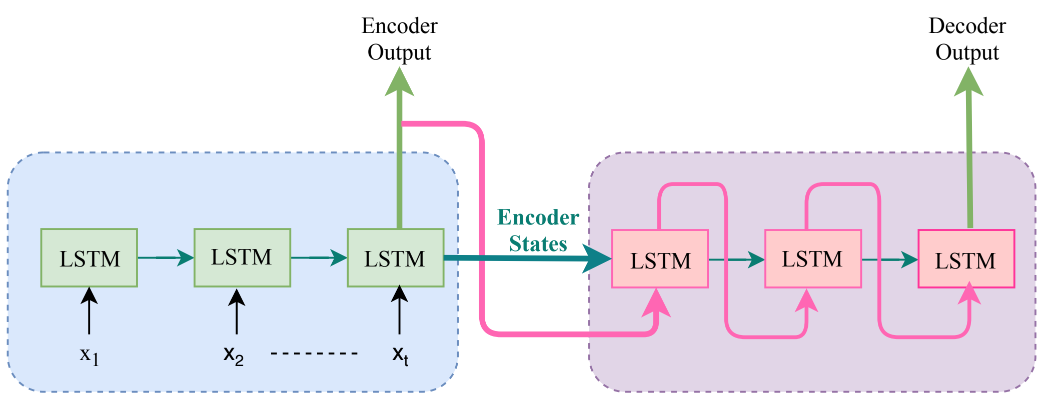

Figure 2: Sequence to Sequence model: This mode uses ”encoder-decoder” and maps input sequence with ”encoder states” and use it as initial state of the decoder.

2.3 Benchmark Model

To imporve the result from traditional time series prediction methods like *ARMA*, *ARIMA* and *SARIMA* we use the deep learning method for this problem. In addition, we select the Seq2Seq model since it can capture the far past effect of the data and do the prediction as a mixture of most recent data in the past and far past. In addition, to have a better comparison, I also developed a Stack LSTM model. I compare the result of these two techniques and I found the Seq2seq have a better result. I expect to see even better result for long term prediction.

3 Methodology

3.1 Pre-processing

To use the neural network(NN), we need to perform specific prepossessing operations to prepare the data for the NN. First, we normalized the data, then we divide the data into three parts: Training, Validation and Testing. Furthermore, we reconstruct the data in a way o have a supervised learning problem. We can do this using previous time steps as input variables and use the next time steps as the output variables. This method is called sliding window. By doing this sliding window, the total size data-set (which has 7 steps in the past and 7 steps in future) is 1407, and we train the Stack LSTM and Seq2Seq models using these processed data.

3.2 Implementation

In this project, I used two different methods for time series prediction, for the Stack LSTM (which I used GRU instead of LSTM), I used **keras** Libarary to implement it and I created a data generator to feed the the network, I used "*dataset_bitcoin.py*" and train the network and then predict the network using model.predict.
Next implementation, was Seq2Seq model. This network was a complicated network and I used **tensorflow** to create this model. This model consists of two parts, encoder and decoder both of these blocks have block of LSTM/GRU, which I used GRU (it is really similar to LSTM).

3.3 Refinement

During the training process I hypertune the parameters of the networks, like number of past window size, hidden layers, learning rate, batch size learning rate and so on. Based on my results the best value for past window size = 7, number of stack LSTM in Seq2Seq is 2 and number of hidden neurons = 100, learning rate = 0.007, and batch size = 100.

4 Results

I improved the prediction result for both method and I improved the result from stack lstm and got better with Seq2Seq model. In Fig. 4, you can see the prediction from Stack LSTM. In Fig. we can see the prediction with

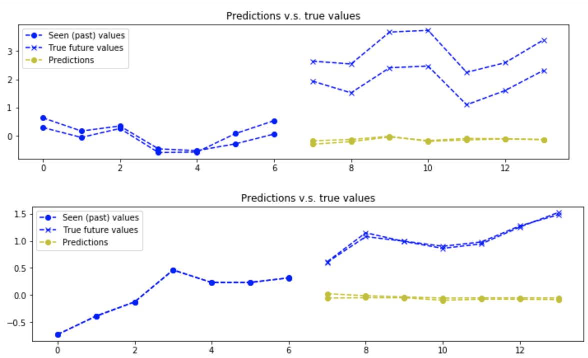

Figure 3: Prediction using stack lstm for Bitcoin value in terms of both USD and EUR.

Seq2Seq model.

4.1 Evolution Metrics

For this problem we use the mean absolute percentage error (MAPE) metric like [1] which is defined as:

$$\frac{100\%}{N}\Sigma_{n=1}^{N}|\frac{y_{pred}-y_{true}}{y_{true}}| \tag{1}$$

where the N is the number of prediction steps. I used this matrix to compare the result of these two methods, I compete the MAPE for value of batch data

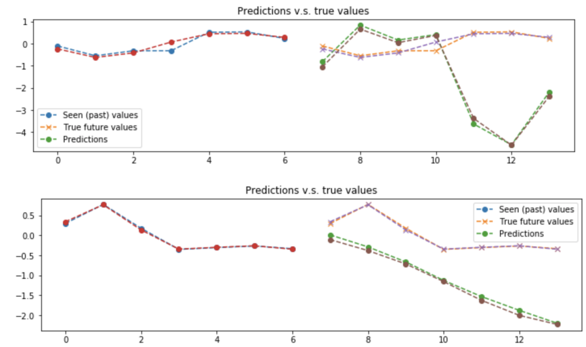

Figure 4: Prediction using Seq2Seq model for Bitcoin value in terms of both USD and EUR.

and then I report the average of this value by computing the mean of MAPE for each batch size. As a result the MAPE from Seq2Seq was the half of MAPE from stack LSTM.
Finally, I got 0.9% as the MAPE in training process with Seq2seq and the average of MAPE in test process is 1.0

References

[1] Suhwan Ji, Jongmin Kim, and Hyeonseung Im. A comparative study of bitcoin price prediction using deep learning. *Mathematics*, 7(10):898, 2019.

[2] Marco Santos. Predicting bitcoin prices with deep learning, 2019.

[3] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.