Project Based Learning: Predicting Bitcoin Prices using Deep Learning

S. Yogeshwaran
Department of Engineering
Amity University Dubai
Dubai, UAE
syogeshwaran999@gmail.com

Maninder Jeet Kaur
Department of Engineering
Amity University Dubai
Dubai, UAE
mkaur@amityuniversity.ae

Piyush Maheshwari
Department of Engineering
Amity University Dubai
Dubai, UAE
pmaheshwari@amityuniversity.ae

***Abstract* — Project based learning is the methodology in which projects drive knowledge and is used in dedicated subjects without negotiating the coverage of the required technical material. This paper discusses the scheme and delivery of project based learning in computer science engineering as major project which adopts undergraduate creativities and emphasizes on real-world, open-ended projects. These projects foster a wide range of abilities, not only those related to content knowledge or technical skills, but also practical skills. The goal for this innovative undergrad project is to show how a trained machine model can predict the price of a cryptocurrency if we give the right amount of data and computational power. It displays a graph with the predicted values. The most popular technology is the kind of technological solution that could help mankind predict future events. With vast amount of data being generated and recorded on a daily basis, we have finally come close to an era where predictions can be accurate and be generated based on concrete factual data. Furthermore, with the rise of the crypto digital era more heads have turned towards the digital market for investments. This gives us the opportunity to create a model capable of predicting crypto currencies primarily Bitcoin. This can be accomplished by using a series of machine learning techniques and methodologies.**

Keywords — Project-Based Learning, Bitcoin, Blockchain, machine learning, SVM, deep learning.

I. INTRODUCTION

In a multidisciplinary education context, project based learning appears one of the most interesting instructional strategies which tries to engage students in authentic real-world tasks to enhance learning [1-5]. In project based learning students typically engage individually or in groups with an instructor or coach or mentor. Each of the project designs and implements an approach to understand practical professional environment in the field of computer science engineering. In this paper, our students have applied project based learning to develop a bit coin pricing algorithm; in addition to technical knowledge they also learned to manage resources and time execution and work in teams.

Building algorithms and models to predict prices and future events has been given significant amount of attention in the past decade. With user data being collected through various forms of paths, there has never been an abundance in raw data like there is now. Any model capable of predicting a future event whether it be to find out what the next big trend is or to predict the next behavior of a customer, most predictive models possess great potential to change opportunity into revenue. The price prediction category is no different. For years analysts and researches have been studying and trying to improve algorithms to help predict future prices. Unfortunately, the protections cannot be based purely on just previous prices, though it should still considered to have the most contribution to the model, other things such as economic growth, social and popularity of the commodity also play a significant role price predictions.

Bitcoin is the first cryptocurrency that was created back in 2009. But, it did not gain much popularity till 2012. In general, cryptocurrencies are just lines of code that can have some financial value. These codes are generated by high-performance computers. Since the creation of bitcoin, many other cryptocurrencies have been created and these are known as Altcoins. Since bitcoins are not managed by any bank and are generally open to the public to do transactions via Blockchain methodologies, it has gained popularity due to the fact of no middleman or examination by tax authorities.

Bitcoins can be thought of like gold in the early 19th century. The banks and traders were ready to exchange it for money but did not have complete control over it. Similarly, bitcoins can be exchanged for money, used to purchase goods and even complete transactions. Each coin who have to be mined and only twenty one million coins will ever exist out of which 11 million have already been mined. One of the problems that analysts and researchers faced was to implement a system capable of accurately predicting the prices. In this innovative project, our students have implemented machine-learning algorithm, for predicting changes in Bitcoin prices in the short run, from historical time series data of quantitative factors that affect Bitcoin prices. The concept was to implement a system capable to analyze real time data and give a sense of direction to investors to help in decision making. The application will take in real world data and will go through a series of data reshaping which will get the data ready to be fed into the machine learning algorithms. Using this data, we are able to predict the Bitcoin price of tomorrow.

The rest of the paper is organized as follows. Section II discusses the related work done in the area of bitcoin price prediction. Section III presents the prediction model with the features. Section IV gives the software requirements of the proposed project work. Section V discusses the project design and implementation. Section VI provides the testing and results followed by conclusion and future prospects.

978-1-5386-9506-7/19/$31.00 ©2019 IEEE

II. Literature Survey

Bitcoin is an electronic money system (EMS) that was created to exchange units of currency called bitcoins, sometimes referred to as BTC [6]. Many electronic money systems have existed and had success in the past but bitcoin differs in that it is a new and unique cryptocurrency with mechanisms that try to mitigate costly challenges to an EMS. A cryptocurrency uses cryptographic controls to eliminate the need for a central authority's involvement in transactions, which removes the risk that they might manipulate the supply of the currency, or feel compelled to mediate on disputes. The upper bound on the amount of cryptocurrency units is known and carefully controlled to mimic a scarce resource such as gold [7].

Cryptocurrency is a digital currency that utilizes cryptography to secure the processes involved in transactions and generation of units. Bitcoin, in particular was the world's first decentralized cryptocurrency which was created in 2009 based on a white paper written by a person with the pseudonym of Satoshi Nakamoto [8]. In centralized currencies, the government or other corporate entities have control over the supply of currency by printing new money. In contrast, bitcoin is a decentralized currency, meaning that no single entity is responsible for the creation of new units or bitcoins [9]. Bitcoin provides a secure way for people to make digital transactions with anonymity. Every time transactions of bitcoins are made, there needs to be a way to organize the orders of bitcoins. It is easy to know who made the transaction in the next line, you need to solve a certain math puzzle. Every time these math puzzles are solved, and a transaction is queued, new bitcoins are created and introduced into the system. Currently, the reward for adding a transaction to the Blockchain is 12.5 newly added bitcoins. Every single bitcoin unit in circulation has been created from these transactions. The reward for adding a transaction to the Blockchain will be halved every 4 years, reaching the limit of 0 at around year 2140. There is currently a limit of 21 million units to the amount of bitcoins in circulation. As the number of bitcoins in circulation approaches closer to the limit, bitcoins become increasingly harder to mine. This limited supply of bitcoins is presumably one reason for the high valuation of bitcoins until January 2018.

Bitcoin's price has arguably behaved like a bubble. For instance, the price was $7854 on November 15, 2017. Just a month later, the price hit an all-time high of $19,511 on December 15, 2017. Then on December 28, 2017, the price has sunk 27% from this high. The bitcoin price has been in the range of $6000 to $11000 since February 2018. With such high volatility, there is an opportunity to make great amounts of money as well as a chance to lose eons. Many people have been scrambling to find ways to make money off this volatile market. Thus, a natural question is whether returns of Bitcoin are predictable. This is non-trivial question especially given a short history of bitcoin and the fact that there are very few studies of the bitcoin return predictability.

Several studies have examined the valuation of bitcoin and other cryptocurrencies. For instance, authors in [10] develop a model of bitcoin pricing and provide mixed evidence about the ability of the model to explain bitcoin prices. In [11], authors consider the valuation of bitcoin and decentralized network assets using an equilibrium model. Several other studies analyze the implications of Blockchain and related technologies for other areas in finance. Raskin and Yermark [12] considered the implications for central banking. Work done in [13] focusses on corporate governance. Authors in [14] investigate bitcoin mining costs. Authors in [15] provides an in depth discussion of the mechanics of cryptocurrencies.

There are studies those have done large analysis on technical analysis. In [16], many studies have examined different aspects of moving average (MA) strategies in financial markets. Work done in [17-19] focus on the profitability of these strategies in equity markets. Authors in [20, 21] compare some specific MA strategies with the buy-and-hold strategy based on certain equity portfolios. The research in [22] forecast the equity risk premium using technical indicators. Work done in [23] test MA strategies using equity exchange traded funds (ETFs).

This paper differs from the already done studies in that it is among the first to examine the predictability of bitcoin value using a predictive model using 26 features using deep learning methodology. For web development support, Django was used.

III. The Prediction Model

After the recent popularity of bitcoins, many researchers have tried to implement prediction models. Building a prediction model for machine learning problem is a difficult task, as there is no right or wrong – best fit must be found over a lot of empirical testing for each specific use case. Many parameters must be tweaked until at least some sensible outcome is generated from the algorithm. This section will go through model building steps and parameter tuning decisions.

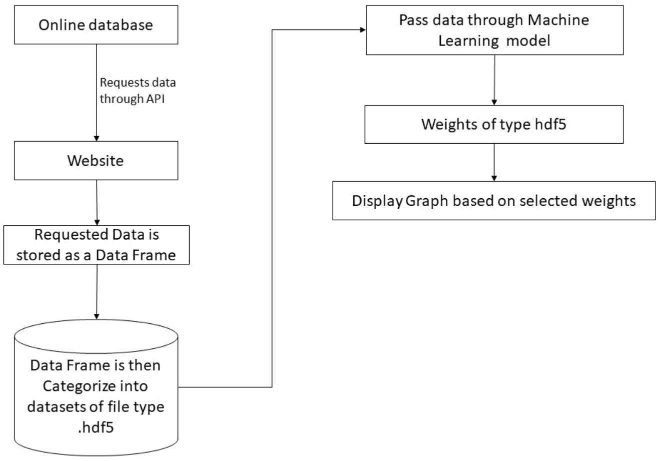

Online database
Requests data through API
Website
Requested Data is stored as a Data Frame
Data Frame is then Categorize into datasets of file type .hdf5
Pass data through Machine Learning model
Weights of type hdf5
Display Graph based on selected weights

Fig. 1. Block diagram of process flow

Support vector machine algorithms have been successfully used in the past as we studies in research works done in [24-28]. In particular, support vector machines (SVM) are suggested to work well with small or noisy data and this have been used widely in the asset returns prediction problems. SVM classification has the advantage of yielding global optimal values. In this project, a predictive model is analyzed based on the input and the accuracy of the result. The block diagram of the process flow is shown in figure 1. There are 26 features that are taken into account out of which only 16 were used to make the final input dataset [29].

The model was built using the SVM to strengthen the architecture of the model as depicted in Figure 1. Once predicted a linear regression was imposed on the predictions to give an idea of the trend. Having the data is critical to build and machine learning model and the quality of data is also important. In such a scenario, there is need to be an algorithm and procedure to check whether the given data is valid. In the project, an anomaly detection model was implemented by using unsupervised learning. K-means clustering was used to group the data into m-data points as there are no labels for the data [30]. Once the group is ready the data was fed into a unsupervised support vector machine to recognize the anomalies in the given sequence of m-data points.

IV. Software requirements of the project

Some of the frameworks used in this project are discussed below.

- This project built primarily on Python. Python is a high level programming language, which is very efficient when trying to build machine-learning algorithms. Since it is an open source language, it has a lot of open source libraries built by third party institutions such as Google for example, which can facilitate in construction of complex programs and algorithms. Complex programs can be written in shorter lines of code in python when compared to Java or other object-oriented programs due to python's modular features. It can also be used to code across wide range of platforms.
- Another language used in this project is Django. Django is the off the shelf, go-to platform for developing python based web apps. It is a framework built for python web development. As Django is also built on python, as a result it supports most functionalities when the right libraries are installed. Python programmers find it more flexible and comfortable to code in Django as the coding environment remains the same for them most part.
- Anaconda navigator is an open source distributor for Python. It focuses on providing IDE's and programming environments for data science and machine learning. Anaconda is widely used because of the custom packages that have been built. It is compatible with Windows, Linux and MacOS. Anaconda also supports development in R programming and has a wide community base surround in the development in R and python development.
- Spyder is one of the platform integrated development environment (IDE) available on Anaconda. It pulls together the powerful libraries required for data analysis and facilitates in building machine learning algorithm.
- The neural networks built on in this project were completed using the Keras libraries. Keras offers neural network API which can run on Tensorflow or Theano. Keras was selected for its user-friendly API's and its ability to support multiple CPU's as well as GPU's. Keras facilitates seamless prototyping. Like all python libraries Keras also takes advantages of the modularity concept providing users with independent configurable modules. These modules are also customizable allowing the developers to create new and more effective model to suit their requirements. Since all the code is purely written in python, python developers do not find it hard to debug or run complex modified code.

V. Project Design and Implementation

A. Data Collection

We collect the historical data from poloniex.com using an REST API call. The API returns data from 2015 to the present day in time intervals of 5 mins and 2 hours. The collected data is then placed into a Data Frame.

B. Data Preprocessing

The Data Frame would contain all the columns that were required as well as a few additional columns. In order to feed relevant data into our model those extra columns will be removed and the filtered data is stored in to a CSV file. The exported CSV file is later then called into difference parts of the overall program and filtered again to get relevant data. The rows of the data frame are stored in . h5py

Code for storing in .h5py with h5py.File(file_name, 'w') as f:

```
f.create_dataset("inputs", data=B)
f.create_dataset('outputs', data=Y)
f.create_dataset("input_times", data=input_times)
f.create_dataset('output_times', data=output_times)
f.create_dataset("original_datas",
data=np.array(original_df))
f.create_dataset('original_inputs',
data=original_B)
f.create_dataset('original_outputs',
data=original_Y)
```

C. Convolutional Neural Network

Convolutional Neural Networks (CNN) is a deep learning methodology used for classification. However, here we tweak it to be used for prediction. By setting up a one-dimensional network instead of 2D or 3D, we can predict the output by feeding in a list of the close prices from our dataset [11].

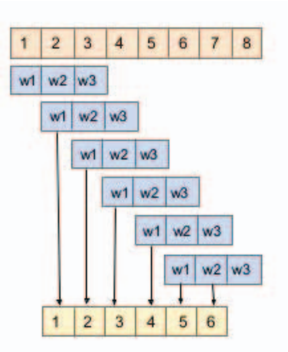


Fig. 2. CNN weights distribution in 1D [11]

There are three layers in the neural network and the output time is based on the below formula:

Output time = (Input time – Kernel size) / Strides + 1

In each of the layers a regularization is add to reduce over fitting. The regularization functions used are Dropout or LeakyReLU which allows few neurons to be removed to avoid gradient decent.

Code for CNN layers:

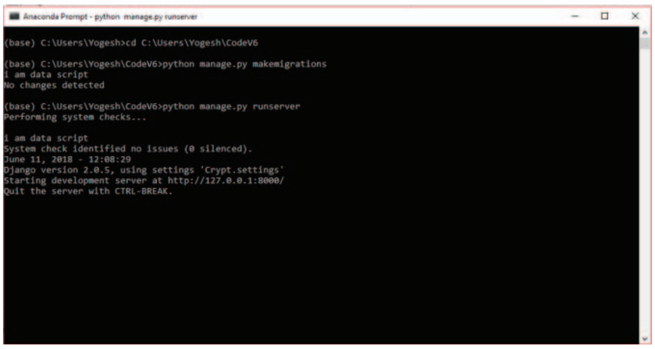
```
model = Sequential()
model.add(Conv1D(activation='relu',
input_shape=(step_size, nb_features), strides=3,
filters=8, kernel_size=8))
model.add(LeakyReLU())
model.add(Dropout(0.5))
model.add(Conv1D(activation='relu', strides=2,
filters=8, kernel_size=8))
model.add(LeakyReLU())
model.add(Dropout(0.5))
model.add(Conv1D(strides=2, filters=nb_features,
kernel_size=8))
```

Here in the project code we have added a dropout layer of 0.5 to say that around 50% of neurons can be dropped to avoid over fitting.

Recurrent neural networks (RNN) are also a deep learning methodology developed in the late 1980s. This neural network is best suited for sequential data [13]. It is much more efficient as it is capable of remembering the weights at each layer and inputting them to the next layer. The RNN make use of internal memory to store the sequence of data per row with the next predictable value on the adjacent upper right cell. The inputs are taken in and run through three gates the Forget Gate, Input Gate and Output Gate. In each of the gates a sigmoid function is applied in all the layers to make sure the output is a value between 0 and 1. Therefore when inputting the value to this layer we scale to transform our input data which is reshaped to fit the neural network.

D. Implementation

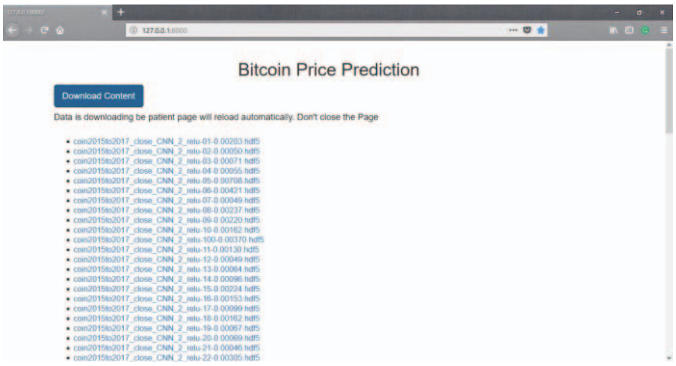
The Web application is designed on the Django web framework and has two pages for one for the CNN network and other for the LSTM (Long Short Term Memory) network. The server hast to start by entering the runserver command into the conda prompt.



```
(base) C:\Users\Yogesh>cd C:\Users\Yogesh\CodeV6

(base) C:\Users\Yogesh\CodeV6>python manage.py makemigrations
i am data script
No changes detected

(base) C:\Users\Yogesh\CodeV6>python manage.py runserver
Performing system checks...

i am data script
System check identified no issues (0 silenced).
June 11, 2018 - 12:08:29
Django version 2.0.5, using settings 'Crypt.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Fig. 3. Starting server

The server can be opened on the browser by entering 127.0.0.1:8000 as the URL.



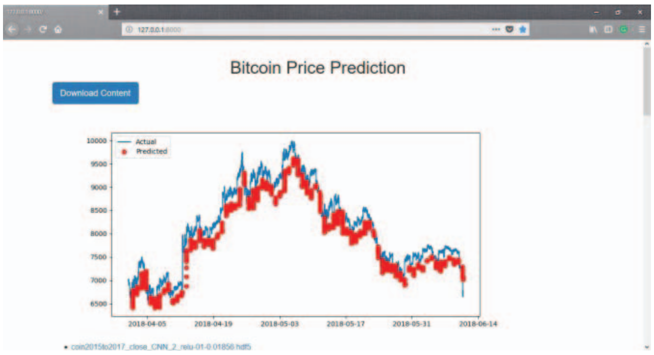
Bitcoin Price Prediction

Data is downloading be patient page will reload automatically. Don't close the Page

- coin2015to2017_close_CNN_2_relu-01-0.00203.hdf5
- coin2015to2017_close_CNN_2_relu-02-0.00050.hdf5
- coin2015to2017_close_CNN_2_relu-03-0.00071.hdf5
- coin2015to2017_close_CNN_2_relu-04-0.00055.hdf5
- coin2015to2017_close_CNN_2_relu-05-0.00708.hdf5
- coin2015to2017_close_CNN_2_relu-06-0.00421.hdf5
- coin2015to2017_close_CNN_2_relu-07-0.00049.hdf5
- coin2015to2017_close_CNN_2_relu-08-0.00237.hdf5
- coin2015to2017_close_CNN_2_relu-09-0.00220.hdf5
- coin2015to2017_close_CNN_2_relu-10-0.00162.hdf5
- coin2015to2017_close_CNN_2_relu-100-0.00370.hdf5
- coin2015to2017_close_CNN_2_relu-11-0.00130.hdf5
- coin2015to2017_close_CNN_2_relu-12-0.00049.hdf5
- coin2015to2017_close_CNN_2_relu-13-0.00064.hdf5
- coin2015to2017_close_CNN_2_relu-14-0.00096.hdf5
- coin2015to2017_close_CNN_2_relu-15-0.00224.hdf5
- coin2015to2017_close_CNN_2_relu-16-0.00153.hdf5
- coin2015to2017_close_CNN_2_relu-17-0.00099.hdf5
- coin2015to2017_close_CNN_2_relu-18-0.00162.hdf5
- coin2015to2017_close_CNN_2_relu-19-0.00067.hdf5
- coin2015to2017_close_CNN_2_relu-20-0.00069.hdf5
- coin2015to2017_close_CNN_2_relu-21-0.00046.hdf5
- coin2015to2017_close_CNN_2_relu-22-0.00305.hdf5

Fig. 4. User window

Once the new weights are updated the graph can be generated by selecting the appropriate weight.



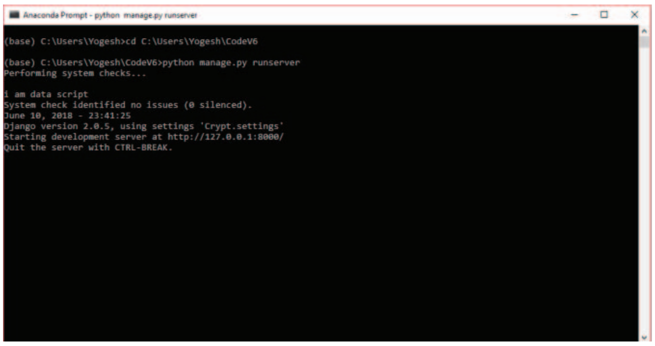
Bitcoin Price Prediction

- coin2015to2017_close_CNN_2_relu-01-0.01856.hdf5

Fig. 5. Prediction graph window

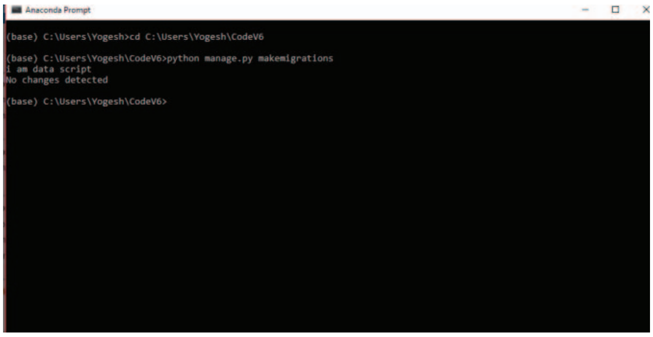
VI. Testing and Results

The website was tested by running the code on a local hosted server. Using Anaconda command prompt, “MakeMigrations” was compiled to make sure all data files lined up properly.



```
(base) C:\Users\Yogesh>cd C:\Users\Yogesh\CodeV6

(base) C:\Users\Yogesh\CodeV6>python manage.py runserver
Performing system checks...

i am data script
System check identified no issues (0 silenced).
June 10, 2018 - 23:41:25
Django version 2.0.5, using settings 'Crypt.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Fig. 6. Conda prompt window



```
(base) C:\Users\Yogesh>cd C:\Users\Yogesh\CodeV6

(base) C:\Users\Yogesh\CodeV6>python manage.py makemigrations
i am data script
No changes detected

(base) C:\Users\Yogesh\CodeV6>
```

Fig. 7. Conada prompt window for migrations

Dataset Testing: Tested running multiple variations of the same data set to identify which was gave the best result.

Neural Network Tuning: Testing efficiency of outputs of the neural network by tuning the number of neurons, increasing and decreasing the number of layers.

Gap Analysis: Testing code on different environments to detect potential gap and scenarios of failure. Customize the code to suit the need of the hosting environment.

Testing with different layers CNN

- Testing with 2 layers - Two layers did not give satisfactory results as the model had predicted values with with more than 30% difference rate form the original values. This caused due to in sufficient number of neurons and layers.
- Testing with three layers - The three-layered approach had displayed promising results with less than 5% difference rate between the predicted and actual values.
- Testing with Four layers and LeakyReLU - The four-layered approach has significantly better results than the 2-layered approach but could not outperform the 3-layered approach

Testing Recurrent Neural Network

- Testing with 2 layers: The 2 layered approach results are not satisfactory. As the same with CNN the predicted values and actual values the difference it much larger than expected.
- Testing with 3 layers - The three layered approach as once again shown improvement in accuracy but still has to be tested against four layers.
- Testing with 4 layers - The four-layer approach is similar to CNN. The four-layer approach is better than 2- layer approach but still not as good at 3 layered approach.

VII. Conclusion and Future Prospects

Predicting the future will always be on the top of the list of uses for machine learning algorithms. Here in this project we have attempted to predict the prices of Bitcoins using two deep learning methodologies. This work focuses on the development of project based learning in the field of computer science engineering, by taking into account the problem definition, progression, student assessment and use of hands on activities based on use of deep learning algorithm to develop application which can predict bitcoin prices.

Django was able to support the deep learning model and bring to life a graphical web app. The convolutional Neural Network though was primarily designed to study and classify images, was customized to predict a sequence of numbers. Though it fell short by having only 5% buffer amount, it was still able to compete with the LSTM model. Through this project, students were able to learn and understand the entire lifecycle of App development. They also gained experience in building machine learning modes and Web development using Django. Prediction models are going to get more complex and effective in the future due to the increase in data collection and development of stronger data analytic strategies. The only factor that might be holding us back is the need for more computational power.

There is always room for improvement and, with the rate at which deep learning is growing, these improvements will surely be possible:

- Train the model on a larger data set to increase prediction accuracy.
- Design model with high number of neurons and run on a supercomputer or a cluster of system.
- Include more features to the feature map and integrate the model with other model which can learn based on customers’ interest to a certain commodity.

Having a well-rounded approach towards prediction is important thus further study is required to find other promising features. The dataset can be broken up onto sequential patterns and a linear regression model to be used on the patterns of data to predict the results, or use K-means clustering to group the data points. These grouped data points can be then used with a deep learning model. New students can pick a few topics from the above exploratory ideas.

REFERENCES

[1] G. Solomon, "Project-Based learning: A Primer," Technology Learning, volume 23, Jan. 2003.

[2] M. Hedley, "An undergraduate microcontroller systems laboratory," IEEE Transactions in Education, vol. 41(4), pp. 345–353, Nov. 1998.

[3] H. Markkanen, G. Donzellini, and D. Ponta, "NetPro: Methodologies and tools for project based learning in internet," in Proceedins of World Conference on Educational Multimedia, pp. 1230–1235.

[4] D. Ponta, G. Donzellini, and H. Markkanen, "NetPro: Network based project learning in internet," in Proceedings of European Symposium of Intelligent Technologies, pp. 703–708, 2002.

[5] S. A. Ambrose and C. H. Amon, "Systematic design of a first-year mechanical engineering course at Carnegie-Mellon University," Journal of Engineering Education, vol. 86, pp. 173–182, Apr. 1997.

[6] "Vocabulary - Bitcoin." [Online]. Available: https://bitcoin.org/en/vocabulary#btc.

[7] "PayPal." [Online]. Available: http://en.wikipedia.org/wiki/PayPal.

[8] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf

[9] Harvey, C. (2014). Bitcoin myths and facts. Working paper, Duke University. Available at http://ssrn.com/abstract=2479670

[10] Athey, S., I. Parashkevov, V. Sarukkai, and J. Xia (2016). Bitcoin pricing, adoption, and usage: Theory and evidence. Working paper, Stanford University.

[11] Pagnotta, E. and A. Buraschi (2018). An equilibrium valuation of bitcoin and decentralized network assets. Available at SSRN: https://ssrn.com/abstract=3142022.

[12] Raskin, M. and D. Yermack (2016). Digital currencies, decentralized ledgers, and the future of central banking. Working paper, National Bureau of Economic Research.

[13] Yermack, D. (2017). Corporate governance and blockchains. Review of Finance 21(1), 7–31.

[14] Huberman, G., J. D. Leshno, and C. C. Moallemi (2017). Monopoly without a monopolist: An economic analysis of the bitcoin payment system. Columbia Business School Research Paper No. 17-92.

[15] Harvey, C. (2016). Cryptofinance. Working paper, Duke University. http://ssrn.com/abstract=2438299.

[16] Brock, W., J. Lakonishok, and B. LeBaron (1992). Simple technical trading rules and the stochastic proper-ties of stock returns. Journal of Finance 47(5), 1731–1764.

[17] Bessembinder, H. and K. Chan (1998). Market efficiency and the returns to technical analysis. Financial Management 27(2), 5–17.

[18] LeBaron, B. (1999). Technical trading rule profitability and foreign exchange intervention. Journal of International Economics 49(1), 125–143.

[19] Sullivan, R., A. Timmermann, and H. White (1999). Data-snooping, technical trading rule performance, and the bootstrap. Journal of Finance 54(5), 1647–1691.

[20] Han, Y., K. Yang, and G. Zhou (2013).A new anomaly: The cross-sectional profitability of technical analysis. Journal of Financial and Quantitative Analysis 48(5), 1433–1461.

[21] Shynkevich, A. (2012). Performance of technical analysis in growth and small cap segments of the us equity market. Journal of Banking and Finance 36(1), 193–208.

[22] Neely, C. J., D. E. Rapach, J. Tu, and G. Zhou (2014). Forecasting the equity risk premium: the role of technical indicators. Management Science 60 (7), 1772–1791.

[23] Huang, J.-Z. and Z. J. Huang (2018). Testing Moving Average Trading Strategies on ETFs. Available at http://dx.doi.org/10.2139/ssrn.3138690.

[24] K. J. Kim, "Financial time series forecasting using support vector machines," Neurocomputing, 55, 307 – 319, 2003.

[25] W. Hunag, Y. Nakamori, S. Y. Wang, "Forecasting stock market movement direction with support vector machine," Computers and Operations Research 32, 2513-2522, 2005.

[26] J. Patel, S. Shah, P. Thakkar, K. Kotecha , "Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning technques," Expert Systems with Applications, 42, 259-268, 2015.

[27] M. C Lee, "Using support vector machine with a hybrid feature selection method to the stock trend prediction," Expert Systems with Applications, 36 (8), 10896 -10904, 2009.

[28] H. Ince, T.B. Trafalis, "Short term forecasting with support vector machines and applications to stoke price prediction," Inernational Journal of General Systems, 37, 677-687, 2008.

[29] htttp://cs229.stanford.edu/proj2014/Isaac%20Madan,%20Shaurya%20Saluja,%20Aojia%20Zhao,Automated%20Bitcoin%20Trading%20via%20Machine%20Learning%20Algorithms.pdf

[30] http://cs229.stanford.edu/proj2014/Phillip%20Pham,Steven%20Li,%20Anomaly%20Detection%20in%20Bitcoin%20Network%20Using%20Unsupervised%20Learning%20Methods.pdf