RO1

FYP Final Report

Real-time Cryptocurrency Trading Suggestion System Using Machine Learning

by

TANG Marco Kwan Ho, SUEN Ka Chun and FONG Chi Chung

RO1

Advised by

Prof. David Paul ROSSITER

Submitted in partial fulfillment

of the requirements for CPEG 4901

in the

Department of Computer Science and Engineering

The Hong Kong University of Science and Technology

2018-2019

Date of submission: April 17, 2019

Abstract

This project is to generate and provide wise real-time decisions on cryptocurrencies to users with the help of our machine learning models and financial models. Users can access our price predictions and suggested portfolios. The project can be categorized into four parts: data preprocessing, machine learning models, server & financial models and a mobile application. Over 300 million cryptocurrency tick-by-tick price data has been fetched and stored by the system. The data is then normalized by our algorithm. Those normalized data are used for training the three machine learning models including Encoder-Decoder LSTM, Bidirectional Encoder-Decoder LSTM, and Attention-Based Encoder-Decoder LSTM while Encoder-Decoder LSTM gives the best performance. The prediction of our machine learning models feeds into the Markowitz Portfolio Selection Model which provides wise real-time decisions to users through our mobile application. Users can easily get updated immediately by viewing charts and receiving notifications by our cross-platform mobile application.

Table of Contents

List of Figures

1 Introduction

1.1 Overview

Cryptocurrencies have become very popular, with trading reaching millions of transactions every day. Cryptocurrency trading is famous for its high volatility compared to foreign currency and securities trading [4]. Frequent changes in exchange rates imply that there are more opportunities for investors to trade and gain from fluctuations in the market [6].

Furthermore, 24/7 cryptocurrency markets are more accessible than stock markets, since they are not restricted to certain trading hours [3] [12]. In addition, the low entry price of some cryptocurrencies enables investors to have more flexibility in building portfolios.

The drawbacks of the high volatility of cryptocurrency trading are increased risks to investors, as they have to make quick decisions in a rapidly changing market [10], and the temptation to allow human emotions to influence trading decisions, which can lead to hasty and illogical decisions that bring financial loss. Thus, taking advantage of modern Artificial Intelligence (AI) technologies [8] and the huge and accessible cryptocurrency datasets on the Internet, we can build machine learning models, automate the investment analysis, reduce risk and avoid hasty and emotional decisions.

In this project, we have implemented a real-time price prediction mobile portal with modern machine learning models in order to automate the trading analysis, provide trading advices in the cryptocurrency market and make trading strategies to assist the user to make profits in the cryptocurrency market.

1.2 Objectives

Our project includes two main goals:

1. Use machine learning models to predict future price and trends of cryptocurrencies
2. Construct a mobile portal to allow people to visit our trading suggestions and prediction result

To achieve our two main goals, we worked on the following objectives:

1. Experiment with several machine learning techniques to see which ones perform best in forecasting cryptocurrency price and trends.
2. Utilize technical analysis to make trading decisions in the cryptocurrency market
3. Provide a user-friendly application for investors of cryptocurrency to look for additional trading suggestions and indicators for short-term trading.

1.3 Literature Survey

1.3.1 Technical Analysis on Financial Market

Technical analysis is a technique that uses the patterns of historical asset price movement to forecast future price trends. If markers are not manipulated, investors can profit from technical analysis. Research conducted at Hamburg University [7] targeted nine popular patterns in technical analysis and built a corresponding trading strategy. During their testing period, they were able to use eight out of nine patterns to make significant profits. These results show that technical analysis based on various commonly patterns can help investors to make profits in financial markets. Thus, we used several of these patterns in our technical analysis.

1.3.2 Machine Learning for Time Series Analysis

Recent research on time series analysis [11] applies an Attention-based Long Short-term Memory (LSTM) model to predict future stock prices. An

Attention-based LSTM is a modern neural network which has usually been used for tackling sequence-to-sequence prediction tasks. The results show that an Attention-based LSTM is able to dig out useful features of stock price data, and an Attention-based LSTM can produce good performance in predicting the stock prices in certain scenarios. However, this research only focused on predicting stock prices but not on generating indicators from an investment point of view, which is not enough for making an all-rounded trading strategy. Thus, we used an Attention-based LSTM model in our system.

1.3.3   Machine Learning Models Incorporated with Technical Analysis

Using machine learning models embedded with technical analysis is another approach to predicting time series financial data.

A study from MARA University of Technology [9] introduced a method of using a nonlinear autoregressive exogenous (NARX) model with a technical indicator called moving average (MA) to predict the Bitcoin price. The researchers used 5,10,20,50,100 and 200-day MAs of the Bitcoin price as an input to their model after particle swarm optimization, attempting to predict short-term and long-term trends. They came up a "one step ahead prediction price" after training the models. The results were positive, as their predicted prices closely followed the actual price, as Figure 1 shows, indicating that using technical indicators can be good approach for constructing machine learning models. Despite the positive results from the study, it only suggests one step ahead prediction, which may not give investors enough time to respond when the cryptocurrency market is very volatile. Thus, we are not using this approach, although we combined technical analysis and machine learning.
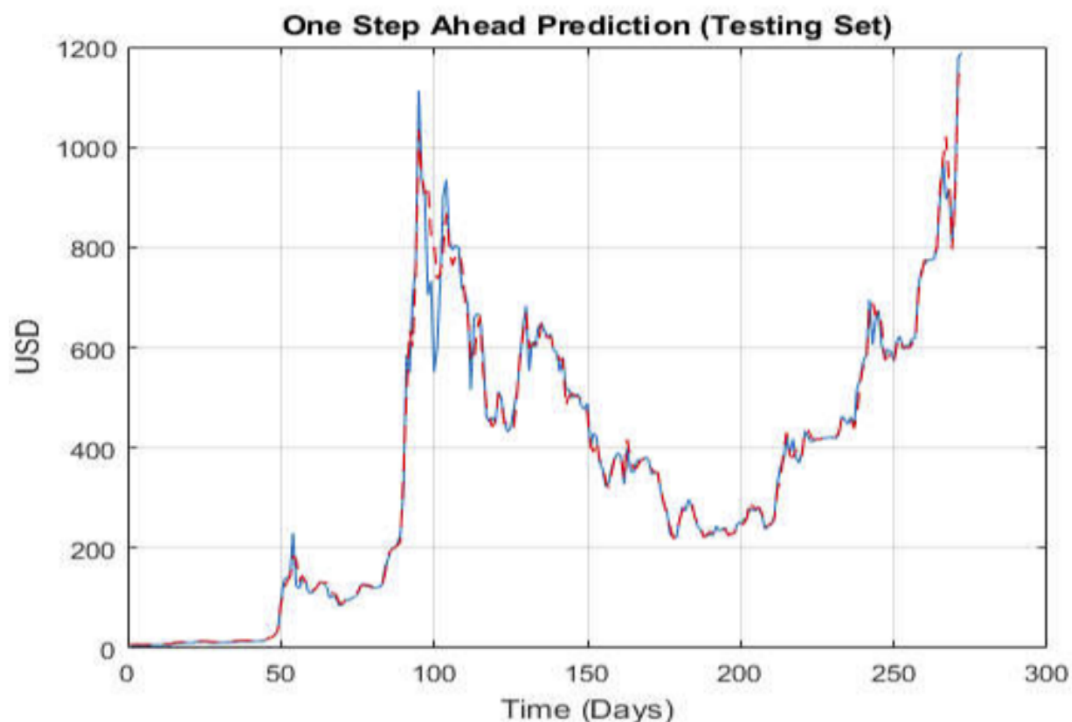
Figure 1: One Step Ahead Prediction Graph [9]

Extracting trends from historical prices is the primary idea of technical analysis. One study [1] introduced a new Multi-resolution Convergence Divergence (MRCD) indicator, which is different from but similar to the orthodox Moving Average Convergence Divergence (MACD) indicator. MRCD is more sensitive and flexible to fluctuations at any frequency interval and for more future possibilities. The study also established an MRCD-NARX neural network model to forecast the trend of a stock. In the end, the study concluded that the MRCD performs much better, as the error range was 10 times less than that of the traditional MACD in their test, while it made a more lucrative portfolio than the MACD-driven model. However, this model may have a significant time delay, as fundamental indicators still play an important role on the stock price. Nevertheless, the study did show that using MACD/MRCD-related indicators can help forecast the price of a stock or currency, fit into learning models and potentially generated profits.

2 Methodology

2.1 Design

From the literature survey above, we acknowledge that:

- The LSTM and Attention architecture can have an excellent performance in analyzing time series data.

- Pattern and technical analysis can be useful for price forecasting.

- Incorporating technical analysis ideas into machine learning can provide good results.

Thus, in this project, our team developed the following three types of LSTM models:

- Encoder-decoder LSTM

- Bidirectional encoder-decoder LSTM

- Attention-Based Encoder-Decoder LSTM model

The task of our machine learning models is by reading the previous 1 Day information, then predict the closing price 60 minutes ahead.

After training them and evaluating their performance, we picked the most suitable one for generating indicators and target price to the cryptocurrency market and developed a mobile application to suggest trading strategies and indicators based on the output of the machine learning models.

As cryptocurrency trading is available on more than one exchange, the indicators and target prices provided in our application are based on the price on the most popular cryptocurrency trading platform - Binance.

The following figure shows the structure of our project:
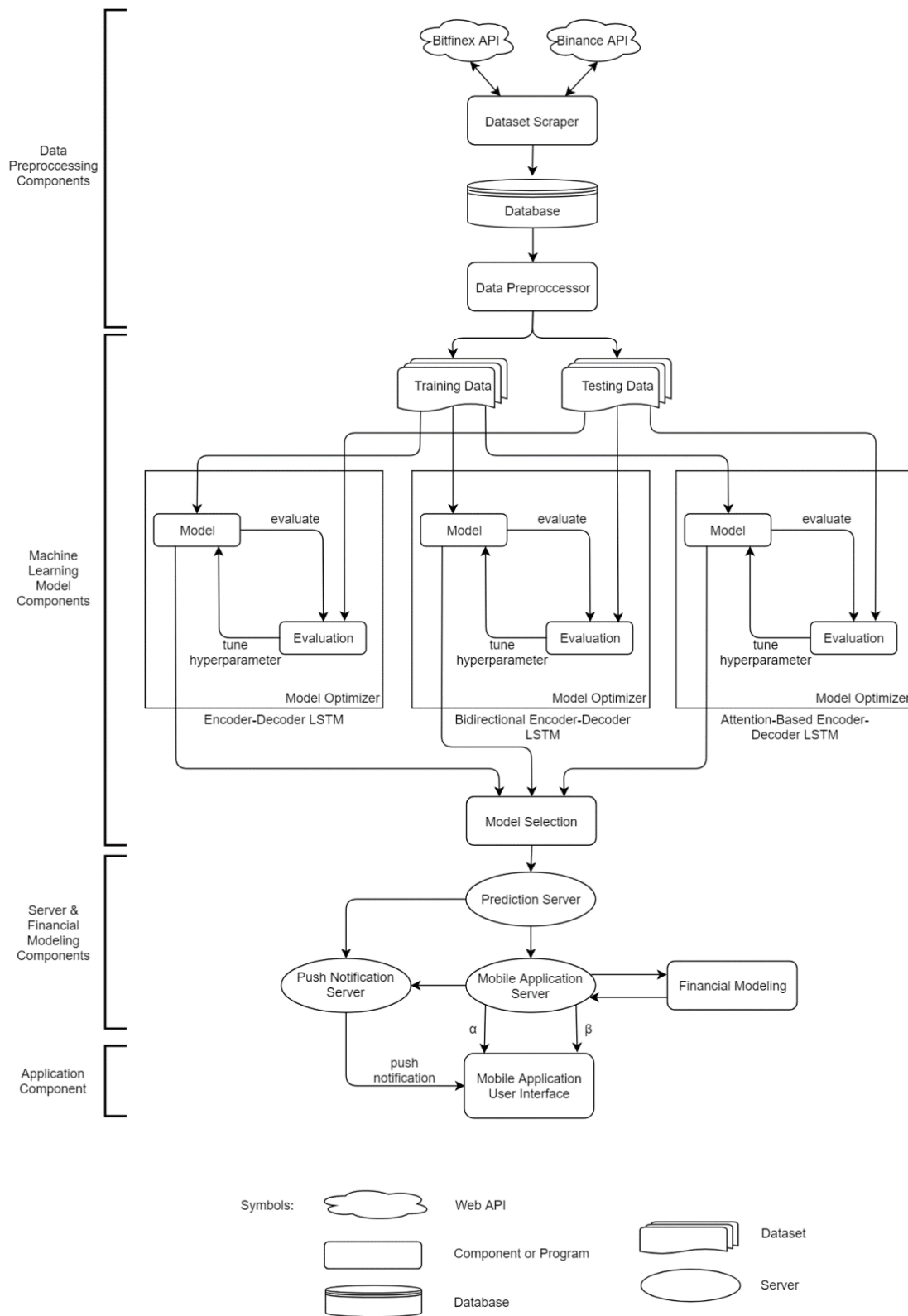
Figure 2: System Structure and Flow Chart

2.1.1 Data Scraper

Bitfinex.com and Binance.com were chosen to be the platforms for our team to collect the data. These two websites provide APIs for developers to scrap cryptocurrency data from specific timeframe. The reason why we need to use two online exchange API is that Binance is limited to the past year, and the data available in Bitfinex is down to 2015. Thus, Bitfinex API is used for collecting the historical dataset and Binance API is used for collecting the latest dataset. We scraped the data of the two cryptocurrencies (Bitcoin & Ripple) in the time interval of 1 minute from Bitfinex and Binance.

We are going to use the term "<u>timestep</u>" which means the time interval between the data throughout this report.

Every timestep of raw data obtained contains the following entries:
'High', 'Low', 'Open', 'Close' and 'Volume'.

2.1.2 Trend Information Data Preprocessing Design

Trends are very important features for analyzing the financial market. Our team found that the EMAs, the MACD (Moving Average Convergence/Divergence oscillator) signal line and MACD histogram are able to reflect the current trend effectively most of the time. Therefore, our team has implemented the Data preprocessing algorithm to transform the raw data to the EMA and MACD data and add them to the datasets.

MACD is based on the difference between 12-period and 26-period Exponential Moving Average (EMA):

$$MACD_{(12,26)} = \text{EMA}_{12} - \text{EMA}_{26}$$

$$EMA_{Current} = (Price_{Current} * \alpha) + [EMA_{Previous} * (1 - \alpha)]$$

$$Smoothing\ Factor = \alpha = \frac{2}{(1 + N)}$$

$$Number\ of\ Periods = N$$

The following image shows what an MACD looks like:

Figure 3: MACD illustration.

Credit: tradingview.com

The MACD data preprocessing provides 4 additional features to our models:

1.   MACD

$$MACD_{(12,26)} = EMA_{12} - EMA_{26}$$

2.   MACD signal

$$MACD_{signal} = EMA_9 \; of \; MACD_{(12,26)}$$

3.   MACD histogram

$$MACD_{histogram} = MACD_{(12,26)} - MACD_{signal}$$

4.   MACD histogram index

This shows how long the price stays in the same side on the MACD histogram.

2.1.3   Price Prediction Model Design

In this project, the task of our models is to read the data of the past day (1440 timesteps) and predict the closing price 60 minutes ahead.

As we have mentioned before, we have implemented the encoder-decoder LSTM model, Bidirectional encoder-decoder LSTM model and Attention-based encoder-decoder LSTM model to tackle the task.

Here is the design of the three types of LSTM models that we are using.
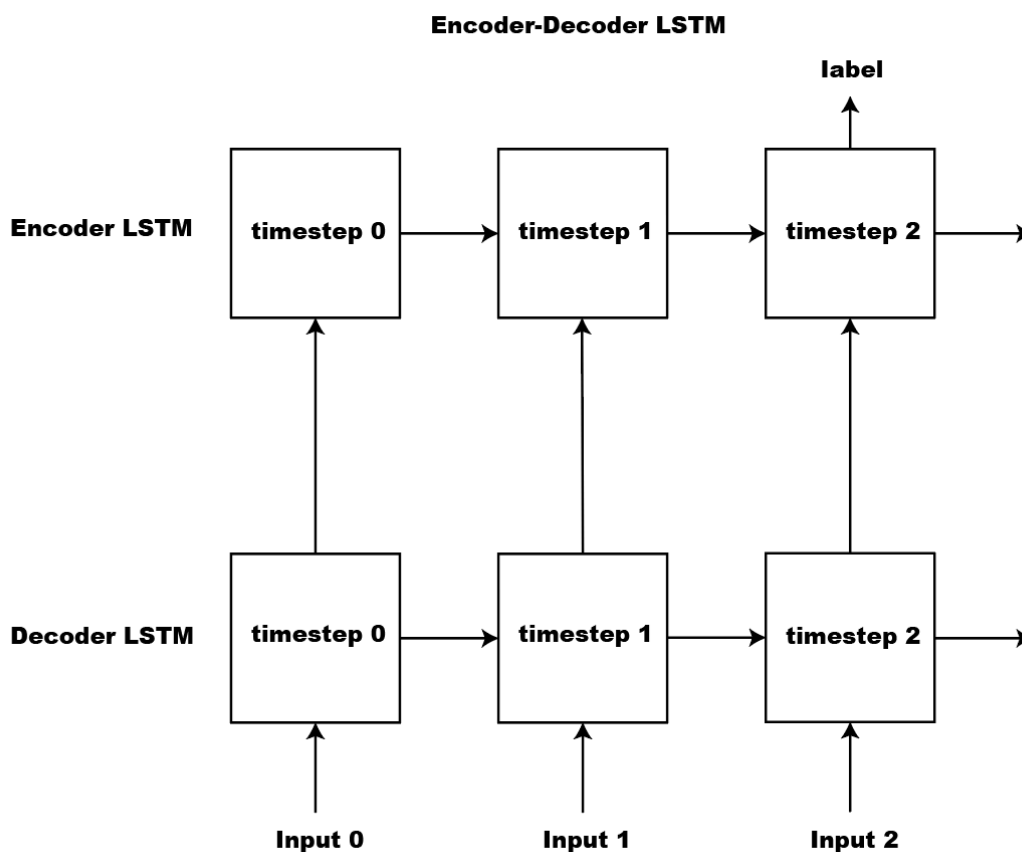
Figure 4: Encoder-Decoder LSTM

The encoder LSTM takes inputs and generates outputs in every timestep. The encoder LSTM takes the outputs from the decoder LSTM together with its own hidden states in every timestep and generates a single output at the last timestep.

Bidirectional encoder-decoder LSTM
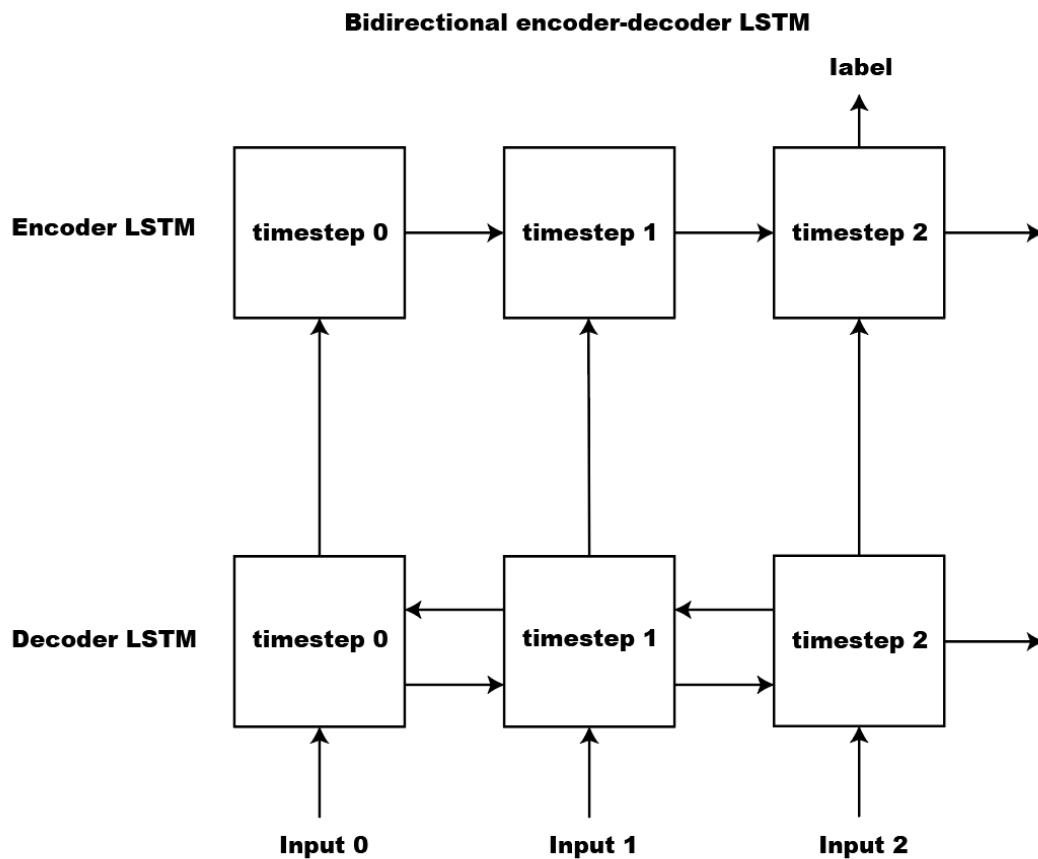
Figure 5: Bidirectional encoder-decoder LSTM

The Bidirectional encoder-decoder LSTM allows the Decoder LSTM to pass its hidden state from either direction. It could enable the model to "remember" information in either way. That allows the Encoder LSTM to consider the bidirectional "memory" to do the regression.
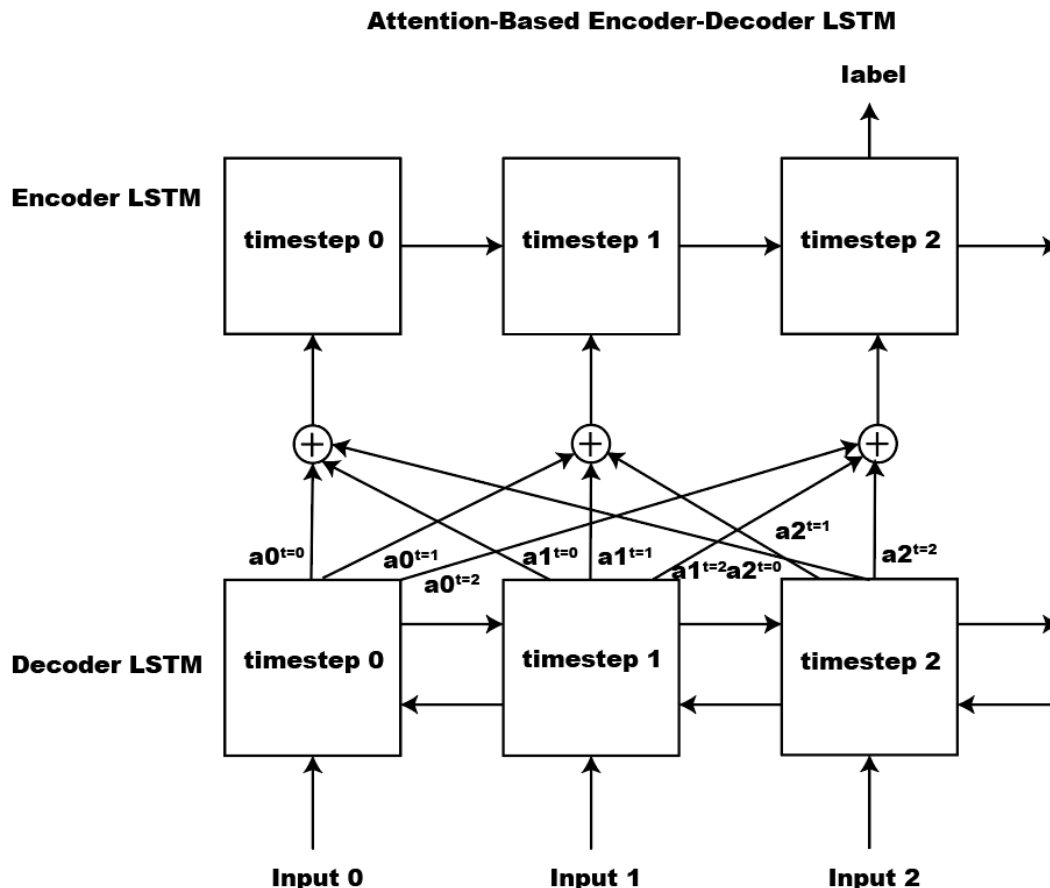
Figure 6: Attention-Based Encoder-Decoder LSTM

The Attention-based Encoder-Decoder LSTM model provides an additional attention layer which allows the Encoder to pay extra attention to the important timesteps through backpropagation.

But the Attention layer comes with a very high space complexity. The traditional encoder-decoder LSTM model with Attention architecture provides individual attention vector for the encoder LSTM in each timestep. The attention layer has a space complexity of $O(n^2)$.

As we are taking 1440 timesteps of data in every prediction (n = 1440), the attention layer is very large, and our graphic card could not manage it.

Revised Attention-Based Encoder-Decoder LSTM
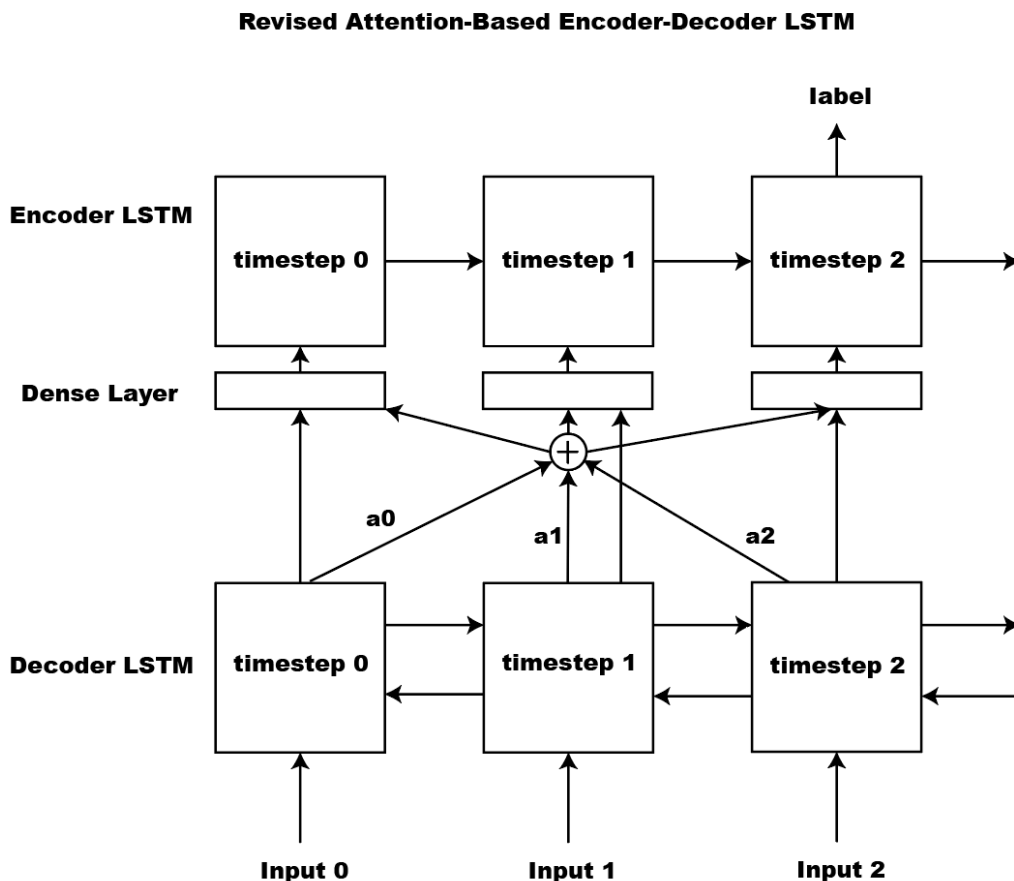
Figure 7: Revised Attention-Based Encoder-Decoder LSTM

In order to solve this problem, we changed the architecture of the attention layer and provide one general attention vector which applies to every timestep of the encoder LSTM. The revised architecture can effectively reduce the space complexity of the model. At the same time, we add an extra dense layer between the Attention layer and the Decoder layer. It is designed for our model to tune the proportion of taking the information from the decoder timesteps output and the attention layer output through backpropagation.

2.1.4 Portfolio Modelling Design

A. β Value

Volatility of a coin can be measured by β value.

1. $\beta_i > 1$: The coin is more volatile compared to the market.
2. $\beta_i = 1$: The coin reacts as same as the market.
3. $\beta_i < 1$: The coin is less volatile compared to the market.

As Bitcoin has the largest market capitalization and the settlement currency for exchanges, the price of Bitcoin is set to represent the market, meaning that the Beta of bitcoin is 1.

$$\beta_i = \frac{Coviance(P_i, \ P_{BTC})}{Varience(P_{BTC})}$$

B. CAPM

Capital Asset Pricing Model (CAPM) is used to show the correlation of systematic risk and expected return of cryptocurrencies. The expected return of the coin can be calculated by CAPM which is essential for the portfolio management.

$$ER_i = R_f + \beta_i(ER_m - \ R_f)$$

$$
\begin{aligned}
ER_i \quad &= \quad \text{Expected return of the Coin} \\
R_f \quad &= \quad \text{Risk-free rate (USDT)} \\
\beta_i \quad &= \quad \text{Beta of the Coin} \\
ER_m \quad &= \quad \text{Expected return of market} \\
(ER_m - \ R_f) \quad &= \quad \text{Market risk premium}
\end{aligned}
$$

C. Markowitz Portfolio Optimization

Markowitz Portfolio Optimization takes in the prediction server output, the CAPM and β value to generate random portfolios. By looking for the portfolios with the highest predicted return with the efficient frontier, the optimization will be output the results for further analysis.

D. Coin Portfolio

The portfolio optimization is based on the Markowitz portfolio optimization, CAPM and β value. Low-risk and high-risk portfolio will be generated to the users based on the β values. Low-risk portfolio is recommended to the users who wish to have a more defensive strategy in cryptocurrency investment and are not willing to endure huge volatility. In contrast, high-risk portfolio is tailored for users who are more advantageous in investment and willing to adopt huge changes in their portfolio.

2.1.5 Mobile Application Design

The mobile application acts a medium for our users to access and interact with our Two feature products: Coin Price Predictor α and Coin Portfolio β.

A. Coin Price Predictor α
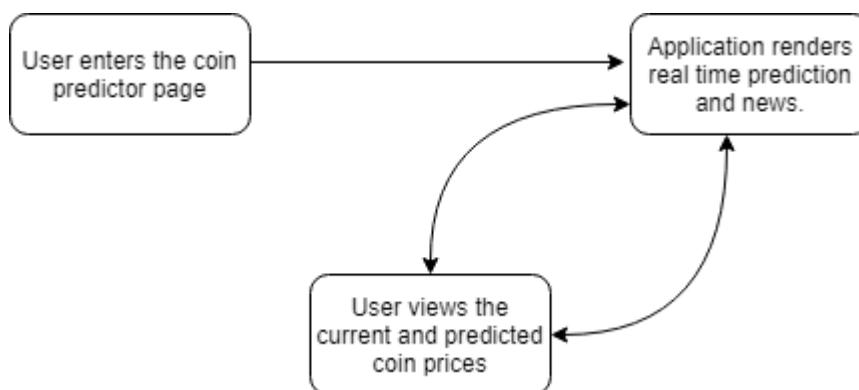
Figure 8: Flow of Coin Price Predictor α

Coin Price Predictor α shows the predicted price 60 minutes later of cryptocurrencies and the current price. Users can make trading decisions based on our predicted prices based on the above machine learning models. Also, users can access to immediate news and performance of the cryptocurrency on the screen.

20
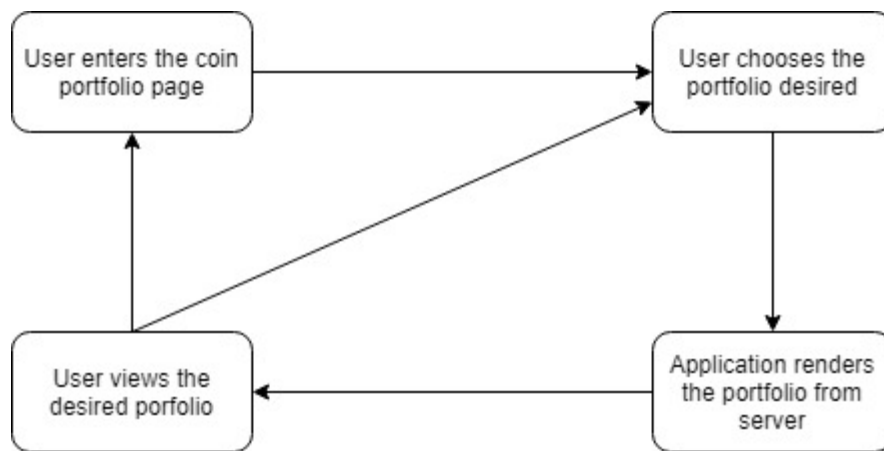
B. **Coin Portfolio β**

Figure 9: Flow of Coin Portfolio β

Coin Portfolio β shows 2 portfolios in two risk levels in the light of the β value: defensive portfolio and aggressive portfolio. Users can check the cryptocurrency components of both coin portfolios to take references to their current portfolio.

C.   Push Notification Service

The push notification is one of the significant features that mobile application can provide to its users. The message pops up on the user's mobile device and receive the message even they are not currently running the application in foreground. We design the push notification service and provide both our price prediction (Coin Price Predictor α) and trading strategy (Coin Portfolio β) to the user.

Expo push notification service is chosen to provide the notification to our end-users. Expo provides its notification backend server for the developers to develop a push notification service in an easy way. This Expo service speeds up our development of the push notification feature.
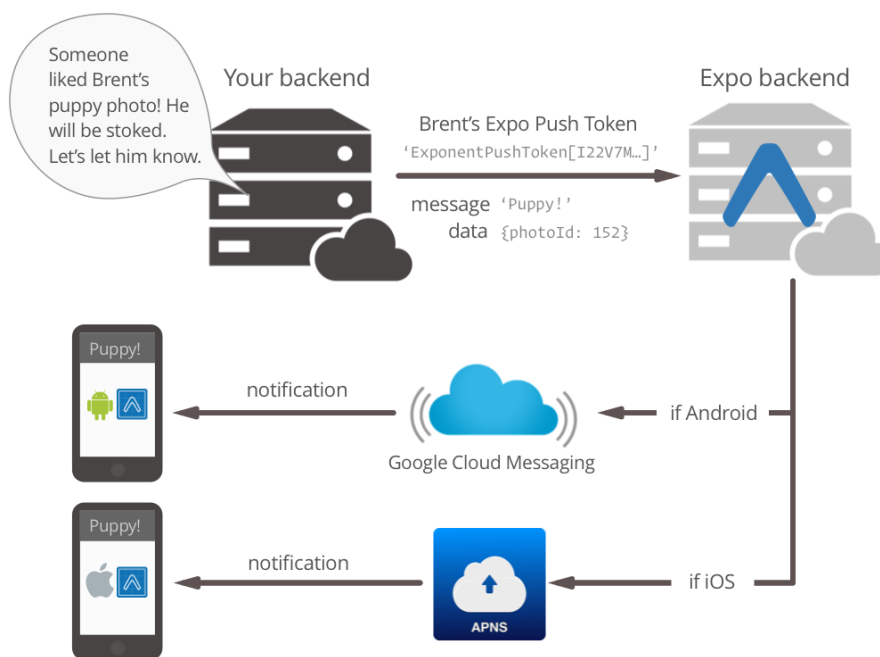
Figure 10: Expo Push Notification

Credit: expo.io

An additional backend server is implemented to access our users' mobile device token from database and send a push notification request to expo backend in attempt to send push notification to our users.

2.2 Implementation

The Implementation Phase included the following aspects:

2.2.1 Cryptocurrency Database

We implemented the database of cryptocurrencies dataset with a NoSQL Database called MongoDB. The database server is hosted on the virtual machine on Google Cloud Platform.

The database server includes 4 databases to store cryptocurrencies data:

1. Database for Raw Dataset from Bitfinex API
2. Database for Raw Dataset from Binance API
3. Merge Database for Bitfinex API and Binance API
4. Database for Merge Database Dataset with MACDs & indicators calculated

2.2.2 Dataset Scraper and its Algorithm

We have implemented the dataset scraper with Python. The dataset scraper is used for collecting the entire dataset of the target cryptocurrencies. The collected datasets are used for training machine learning models. We used Bitfinex and Binance API to gather the entire pricing data of the 2 cryptocurrencies (Bitcoin & Ripple) in 1-minute time intervals.

The dataset scraper has two modes:

1. Historical Mode
2. Batch Update Mode

The Historical mode is used for creating new databases of cryptocurrencies and store the entire historical dataset into the database server. The data was scraped from the current time to the oldest timestamp in sequence. Only Bitfinex cryptocurrency data was scraped with this mode.

Batch update mode is used for updating the existing database with the latest data. The dataset would be crawling from now to the newest timestamp in sequence. Both Bitfinex and Binance cryptocurrency data would be crawled from this mode.

We have implemented a scheduler to run the batch update algorithm per minute in our server on Google Cloud Platform. Multi-threaded architecture has been implemented for both the historical mode and the batch update mode to enhance the performance of the algorithm.

2.2.3   Data Splitting

The data used for training and testing was from 2017-1-1 to 2018-10-22. Our team have decided not to use the data before 2017-1-1 because of that decrease the performance of our model by experiment. Our team suspected that it was due to the changes in the cryptocurrency market from time to time.

Our team has decided to use 60% of the dataset as the training set and 40% of the dataset as the test set since our task is a time series analysis. It is not appropriate to use K-fold cross-validation in our task. Therefore, we leave a slightly larger test set for testing our models.

The following diagram illustrate how we do the train test split:
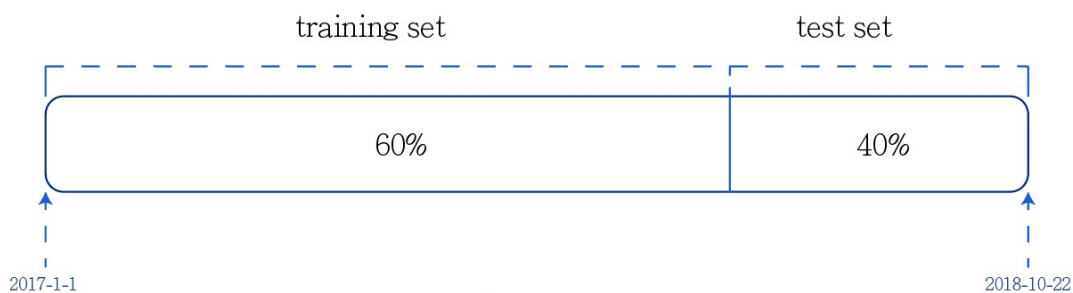

Figure 11: Train Test Split

Note that the cryptocurrency Ripple launched on 2017-05-19, so our train test split for Ripple data starts from 2017-05-19 to 2018-10-22 and following the 60-40 train test split ratio.

2.2.4 Data preprocessing

All the preprocessing below is implemented by Numpy and Pandas. They provide rich mathematical calculation libraries and row/column-wise operations to increase the preprocessing efficiency.

We have tried to put in the raw data for training the machine learning models, but it turns out that our machine learning models could not learn from the dataset and generates over 600% of MAE, this is due to the input data ranging problem. Therefore, we have made use of the following data preprocessing approach to limit the input data range to resolve the problem successfully.

As we have mentioned on part 2.1.1, we have preprocessed the raw price data to MACDs and EMAs data and added to our datasets. The MACDs and EMAs data will also apply to the normalization procedures below.

For the price columns (High, Low, Open, Close), we find the minimum and maximum around these columns within the dataset and use the following formula to perform the data normalization:

$$encoded\ price = \frac{original\ price - price_{min}}{price_{max} - price_{min}}$$

Since the Volume varies hugely, we put it into a log scale before we implement further normalization.

For all the other Non-price columns (Including Volume, MACDs), we find their minimum and maximum and apply the following formula to perform the data normalization individually:

$$encoded\ entry = \frac{original\ entry - column_{min}}{column_{max} - column_{min}}$$

The rationale of this approach is to make use of the normalization formulas above to transform the data from an unlimited range to $0 - 1$. It can boost up the training efficiency of our machine learning model.

2.2.5   Build and optimize the models

We have built and trained the three types of LSTM models by KERAS. They all share the same inputs, labels, training and optimization procedures below.

Models' input:

The input of the price prediction model is the normalized price data (high, low, open, close), normalized volume data, the 4 normalized MACDs features mentioned in 2.1.1 together with the normalized EMAs of the previous 1440 timesteps.

Models' label:

The labels of the model are the normalized high, low, open, close price data after 60 minutes.

Models training:

Since the dataset is huge, the training procedure takes a lot of time. Therefore, our team took 15 strides on the samples (take every 15 samples of the data) to train the model. It does effectively speed up our training process.

Hyperparameter tuning:

We have integrated the black box optimization framework - Scikit Optimize into our code to perform hyperparameter tuning on our machine learning models. That framework uses a statistical approach to try different combinations of hyperparameters in order to optimize the performance of models.

The hyperparameters below is optimized by the Scikit Optimize:
- number of Decoder LSTM units
- number of Encoder LSTM units
- Learning rate
- number of Dense layers (FC layers)
- number of units of the Dense Layers
- Dense Layers activation functions

We have used Scikit Optimize to run 50 rounds of hyperparameter tuning on each of our models. Then, we pick the model which gives the lowest mean absolute error (MAE) on their prediction on prices for each type of model and each type of cryptocurrency.

Pick the best model for the Mobile Application:

For each Ripple and Bitcoin, we have evaluated the performance of the three types of LSTM models on their prediction of direction and price after hyperparameter tuning, then we picked the best model among them for generating trading suggestions to our mobile application.

Finally, we decided to use the ***Two-layered Encoder-Decoder LSTM*** model for predicting both Ripple and Bitcoin in our mobile application., the detail of our evaluation is mentioned in part 2.3.1.

2.2.6   Prediction Server

The prediction server has been implemented with Python. It checks the database every minute. Whenever new data was added to the database, the prediction server will use the latest data to run the best models and generate 60 minutes ahead predictions for Bitcoin and Ripple. Then the server will broadcast the prediction results using the WebSocket implemented by the python library - tornado.

2.2.7   Mobile Application

The mobile application is composed of a coin portfolio advisor and a coin price predictor. The application is based on the React Native framework and Expo which allows the application can be run on iOS and Android operative system using the same code. React-native-elements and react-navigation are used to design the application user interface. Victory-native is used for data demonstration. As we are using different packages to enhancing development quality and speed, the packages are managed by Yarn which provides secure and convenient dependency management. Socket.io framework has been used to manage the information communication between the application frontend and the backend server.

Figure 12 shows the home page view of the mobile application.
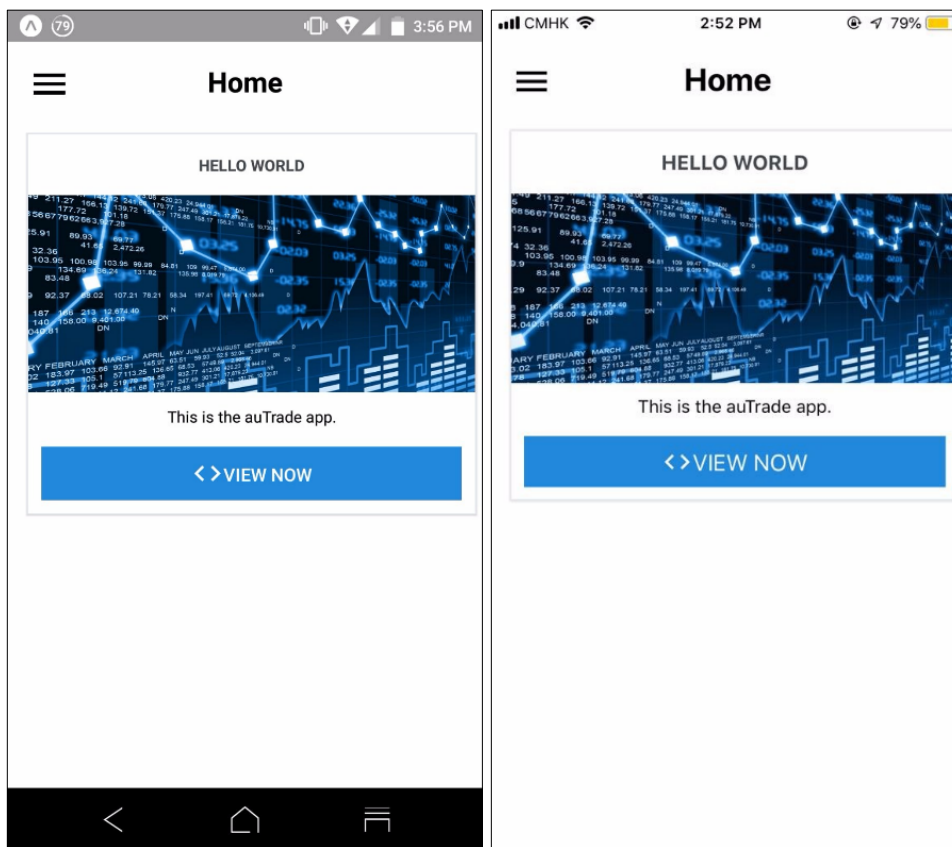
Figure 12: App view of Home Page. (Left: Android, Right: iOS)

Coin Price Predictor α

Coin Price Predictor α predicts coin prices 60 minutes later fetched from prediction server via the mobile application server. The screen also shows the current price of each predicted cryptocurrencies for comparison. An icon on right-hand side shows how the predictor foresees the coin, whether it will drop or rise 60 minutes later. These components are implemented by Socket.io and React-native-elements.

The additional information page of each cryptocurrency was implemented in the mobile application. The page shows the last hour price chart and essential data of the cryptocurrency with the latest news from the internet. User can navigate to the additional information page when user click the corresponding cryptocurrency box (Figure 13).
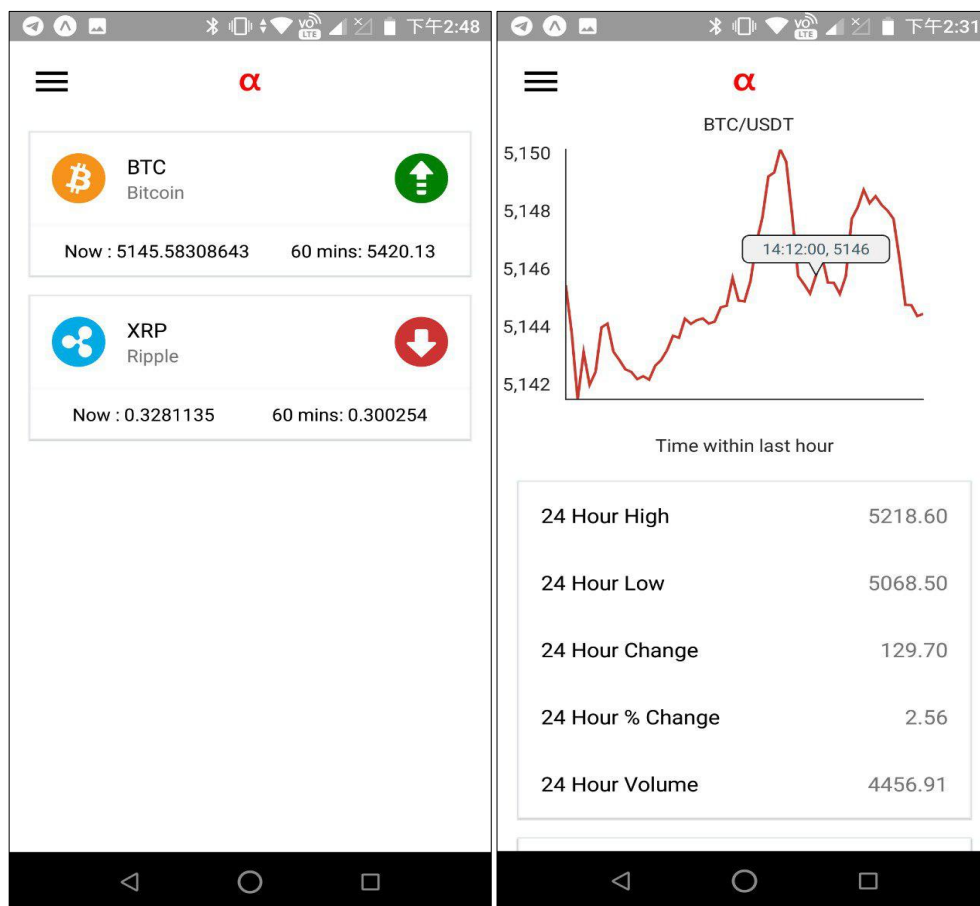
Figure 13: App view of α

Coin Portfolio β

Coin Portfolio β results mainly come from the prediction price of our machine learning models. Users can choose on low or high-risk portfolio depending on their willingness and their preferences on risks. Victory-native is used to visualize the proportions and components of cryptocurrencies in the portfolio.
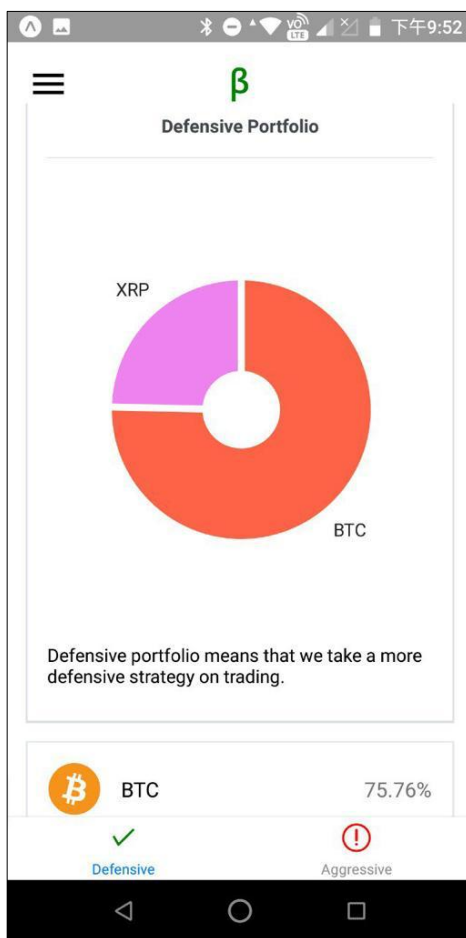
Figure 14: App view of β

Push Notification

Push notification service has been implemented by both front-end and back-end side.

The front end of push notification service has been integrated into mobile application. When users login their account via the mobile application, their mobile device token would be stored on firebase real-time database simultaneously. User can also select their expected notification time interval via mobile application setting page, their preference would be also stored on firebase real-time database.

The notification backend service was implemented with Python. The backend handles the device token and the user selected notification interval and send the push notification request to Expo backend server.

Push Notification allows the user to keep track of the coin portfolio without the need to keep focusing on their mobile devices. Our server will send two notifications to the user's mobile device in each user-defined interval. The first notification provides real-time price prediction (Coin Price Predictor α) with the percentage of price change. The next notification shows the real-time trading strategy (Coin Portfolio β) in both aggressive and defensive portfolio.

Figure 15: Push Notification View in Mobile Device

31

Authentication Component

The authorization of the application was implemented with Firebase, a Backend as a Services (BaaS) platform from Google. Users can create its own account via email and login to use our service. The Login page was developed with authentication component in using React Native Firebase module (Figure 16).
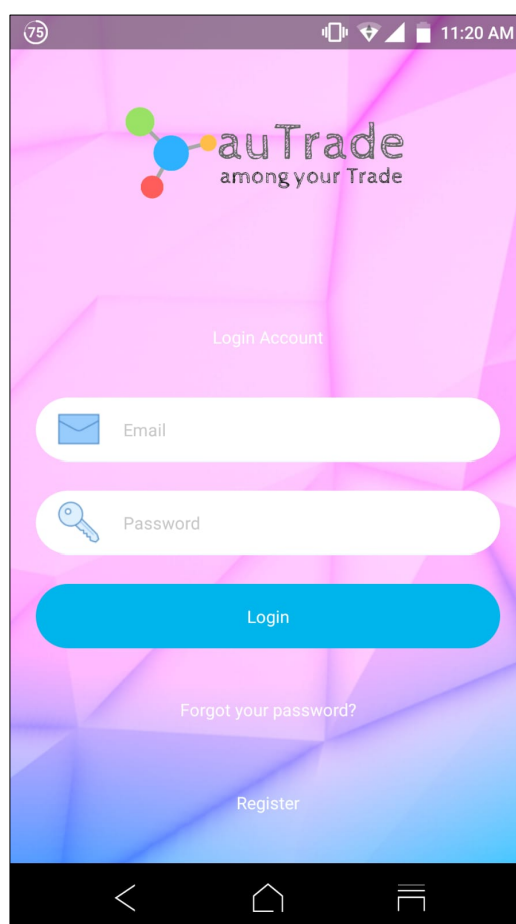
Figure 16: Login UI in Mobile Device

Mobile Application Server

The Mobile application server is managed to collect data from the prediction server and calculate the Coin Price Predictor α and Coin Portfolio β in every timestep as well as broadcast the result to all connected clients.

The Coin Price Predictor α is directly from the output of the prediction server.

The coin portfolio β comes from the concept by Markowitz Portfolio Optimization. An efficient frontier showing the optimal portfolio options to achieve the best returns with respect to their expected returns and volatility.
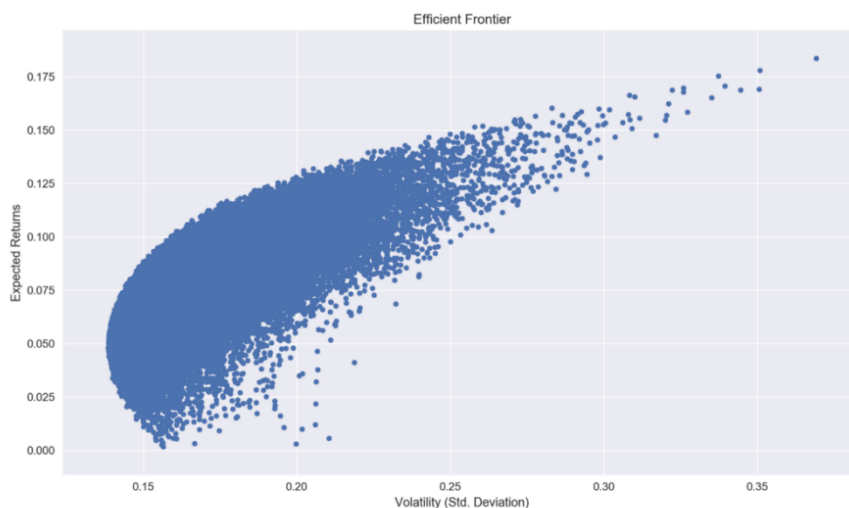
Figure 17: Efficient Frontier

Credit: Medium

Two-coin portfolio in defensive and aggressive strategy is generated by our script. The script is written in Python. Pandas, Numpy and Scipy are used to find out the optimal portfolio. Random Portfolios with different diversification are generated and a Markowitz Bullet is shaped. The script picks a portfolio based on the expected returns and volatility of cryptocurrencies and obtains the β.

Socket API for the frontend

The backend of our system sends the calculated αs and βs to the connected client in every timestep. The socket API has implemented in socket.io and tornado.

2.3 Evaluation & Testing

2.3.1 Evaluation of the LSTM models' result

As we have mentioned in part 2.2.3. We have evaluated the performance of our models on both Ripple, and Bitcoin after hyperparameter tuning.

The following tables show the performance of the three LSTM models on the test set data of Ripple (250678 samples in total) and Bitcoin (370132 samples in total):

	Mean absolute error (MAE) on price prediction	accuracy of directions
Encoder-Decoder LSTM	1.36%	50.9%
Bidirectional Encoder-Decoder LSTM	1.91%	50.8%
Attention-Based Encoder-Decoder LSTM	2.29%	50.7%

Table 1: Models' Performance on Ripple

	Mean absolute error (MAE) on price prediction	accuracy of directions
Encoder-Decoder LSTM	0.77 %	50.6%
Bidirectional Encoder-Decoder LSTM	0.71%	50.3%
Attention-Based Encoder-Decoder LSTM	1.37%	49.8%

Table 2: Models' Performance on Bitcoin

The Encoder-Decoder LSTM out-performed the other two LSTM models on predicting the future price of Ripple in both MAE and accuracy of directions.

Especially for the MAE, the performance of the Encoder-Decoder LSTM is significantly better than the other two models.

The Encoder-Decoder LSTM also has the best performance on the accuracy of directions when evaluating the future price of Bitcoin. But the MAE of the Encoder-Decoder LSTM is slightly larger than the Bidirectional Encoder-Decoder LSTM.

Since the Encoder-Decoder LSTM has the best performance on the accuracy of directions for both Ripple and Bitcoin, also it has the best MAE when making predictions on Ripple with just slightly larger MAE comparing to the Bidirectional Encoder-Decoder LSTM when making predictions on Bitcoin. The Encoder-Decoder LSTM is the best-performed model throughout our experiment.

Therefore, our team has chosen the Encoder-Decoder LSTM model to be the model to generate predictions for our mobile application for both Ripple and Bitcoin.

The following diagrams show the 60 minutes ahead predictions of the Encoder-Decoder LSTM -Decoder LSTM model on the Ripple and Bitcoin price:
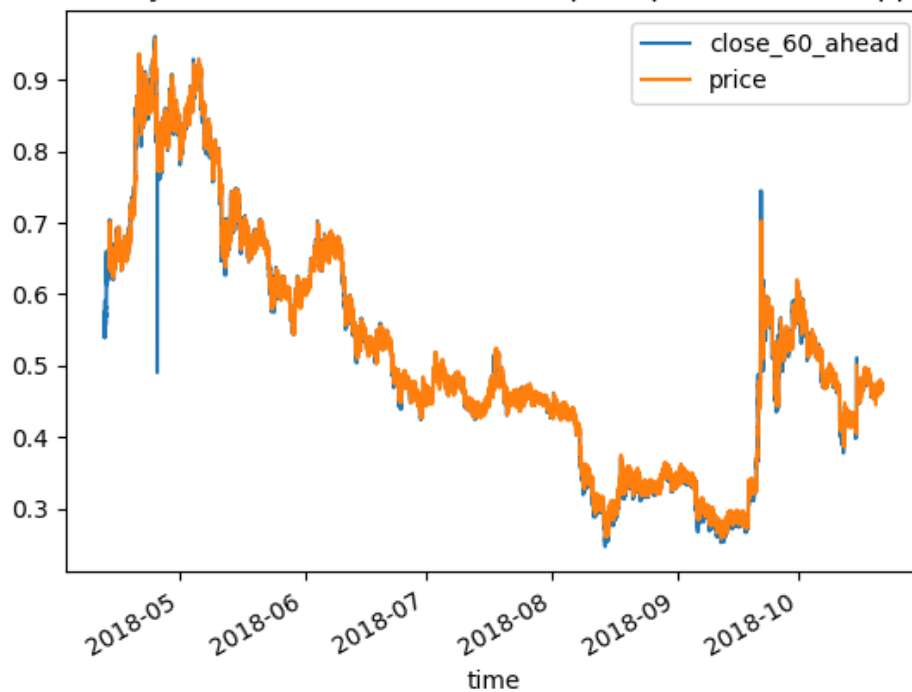
Figure 18: Encoder-decoder LSTM's price prediction

2.3.2 Evaluation of the LSTM models' performance on the cryptocurrency price data

The following three diagrams show the three models' 60 timesteps ahead price prediction on the Ripple test set (250678 samples in total):

Figure 19: Attention-based Encoder-decoder LSTM's price prediction

37

Figure 20: Bidirectional LSTM's price prediction
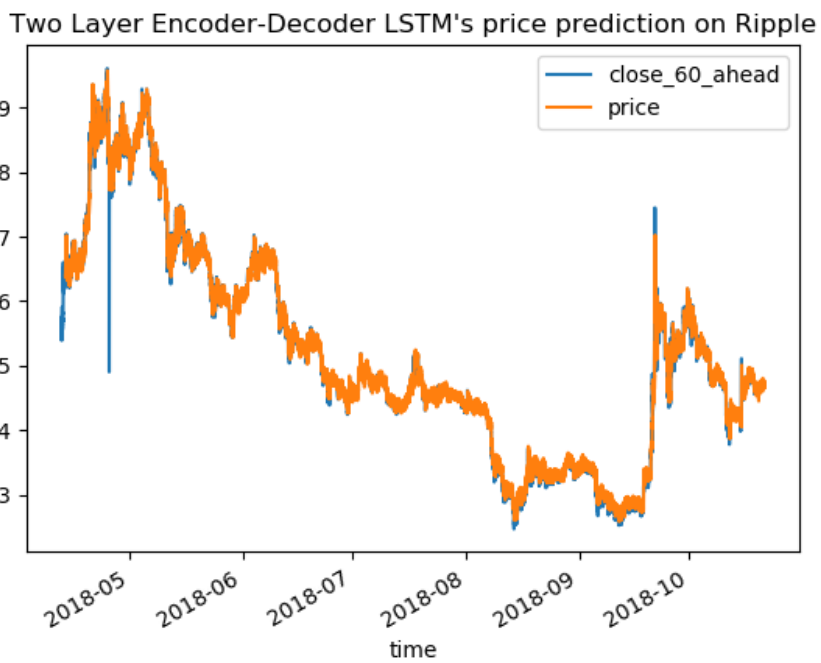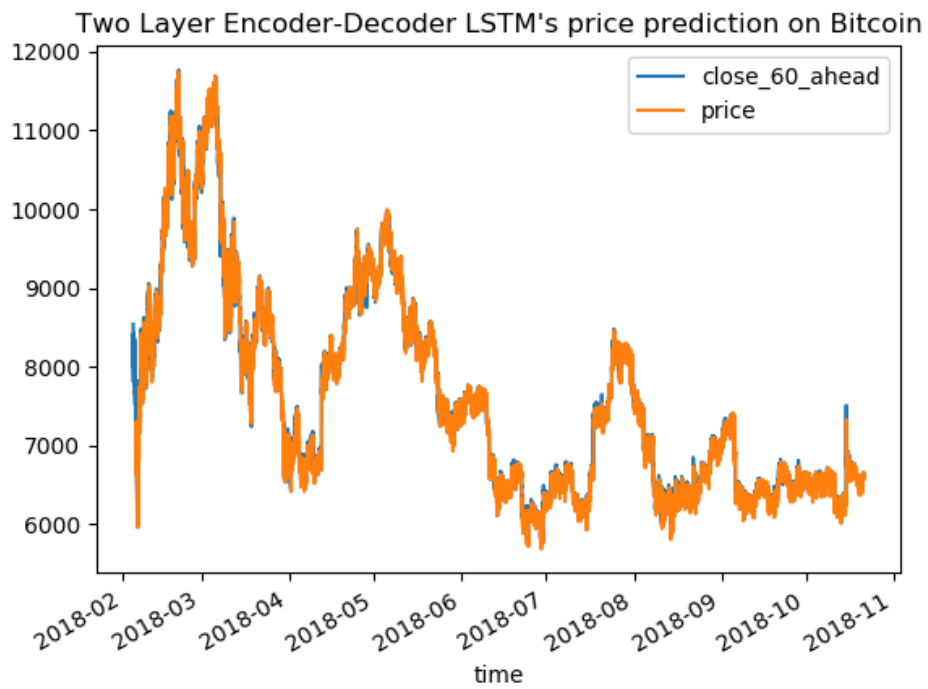
Figure 21: Encoder-decoder LSTM's price prediction

The diagrams above show that all the three models have very decent predictions on predicting the cryptocurrencies' price. Also, according to table 1 and 2, all of them are having a low MAE 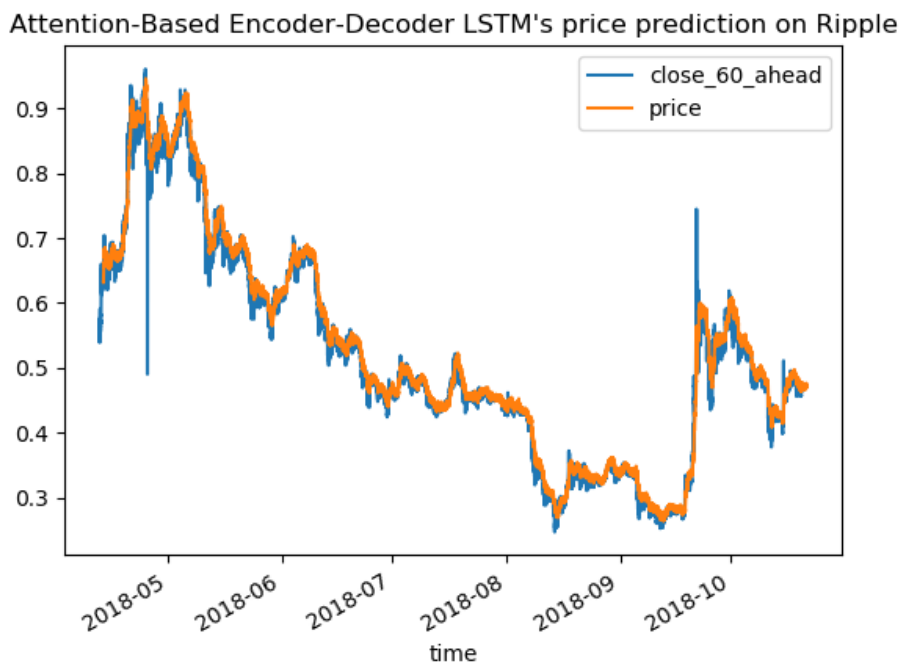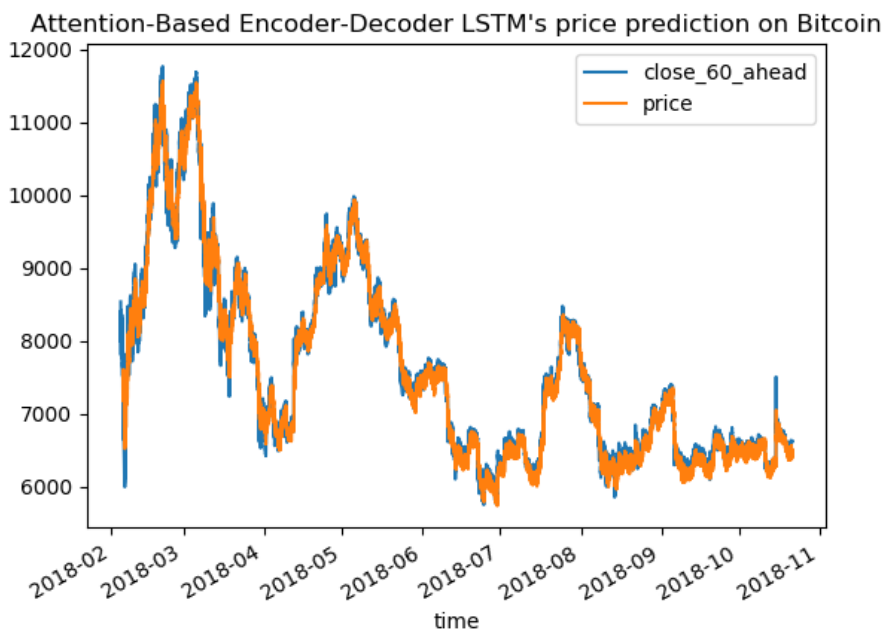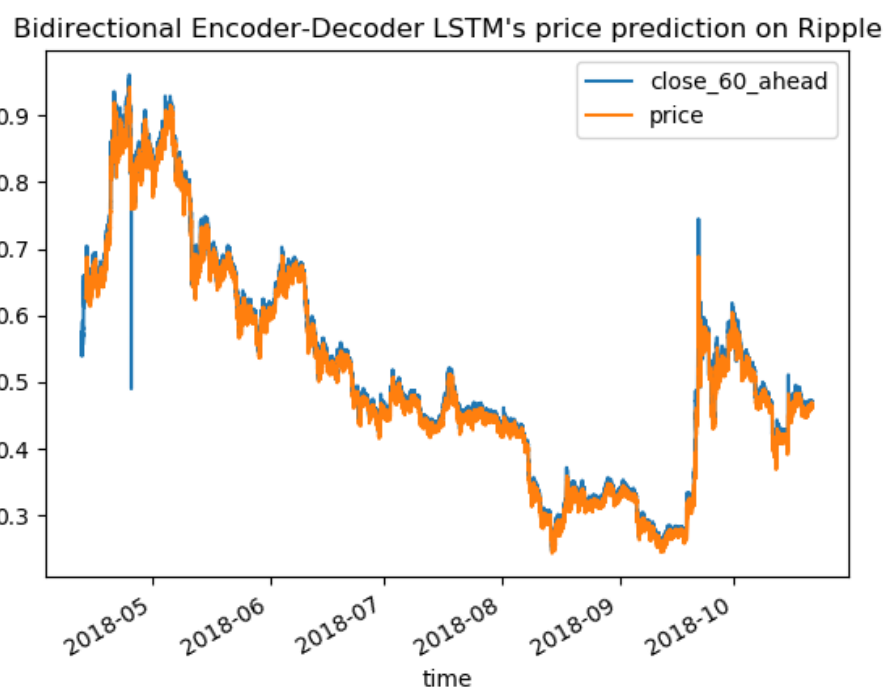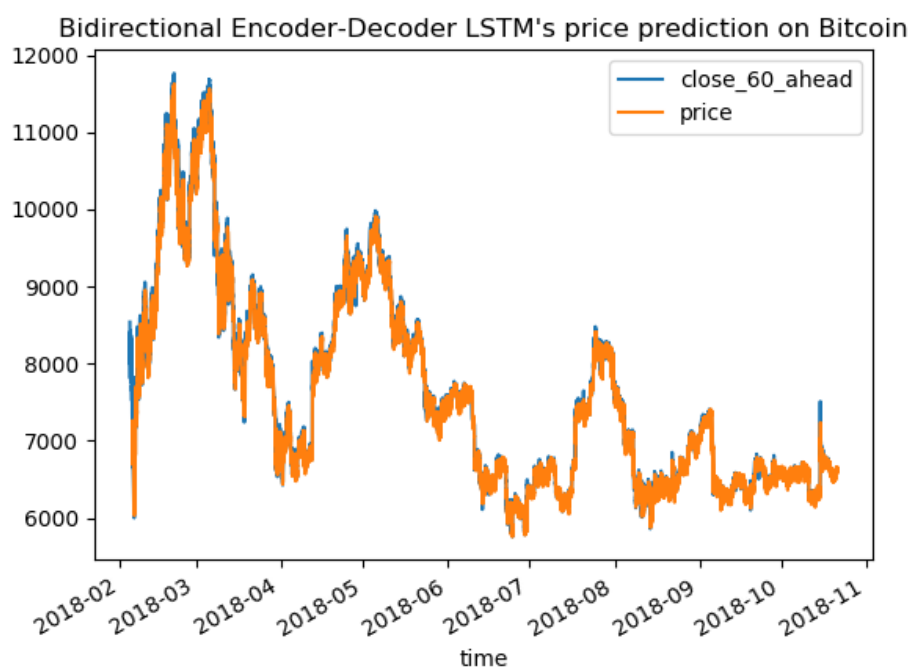on predicting the cryptocurrency price. The result reveals that all the three types of LSTM models are capable of responding to the price change and predict the price of cryptocurrencies with good accuracy.

According to table 1 and 2 again, none of the models have significantly high accuracy on predicting the directions. The major reason might due to the time intervals of our data. Since our data are in a very short time interval (1 minute), our models are considered to evaluate those very short-term changes. Those minute-wise changes are affected by a lot of uncertain issues like the instant buying behavior, the random oscillations and the actions of the institutional investors. Those uncertain issues contribute a lot of noise to our minute-wise data and negatively affects the performance of our prediction models.

Although our models' accuracy on predicting the directions is not high, they are all greater than 50% except the Attention-Based Encoder-Decoder LSTM on Bitcoin. Also, our sample space is huge (250678 test set samples for Ripple, 370132 test set samples for Bitcoin). Which shows that our models' predictions on the cryptocurrencies price bear reference value.

Another possible cause of the low accuracy on predicting the directions is the models themselves are regression models on predicting the cryptocurrencies' price but not classification models on the future price direction. So, we cannot ensure the models to learn about predicting the trend through training and backpropagation.

By comparing the performance of the three models, we discovered that the Encoder-Decoder LSTM works best, and the Attention-Based Encoder-Decoder LSTM and Bidirectional Encoder-Decoder LSTM have worse performance comparing to the Encoder-Decoder LSTM. Which implies that the attention layer and the bidirectional architecture does not help the machine learning models to achieve higher directional accuracy and a smaller MAE on the cryptocurrency price data.

2.3.3   Testing on the Mobile Application

Our mobile application provided a platform for people to access our prediction and portfolio results for making trades, thus user testing is crucial for evaluating the usability of our application. Our group found a group of testers to provide feedback on the mobile application by rating with the following item on scale 1 to 10. 1 stands for the least favorable and 10 stands for the most favorable.

Usability Testing	Average Rating
Usability of Login page	7.8 / 10.0
Usability of Coin page	8.1 / 10.0
Usability of Coin Info & News page	7.6 / 10.0
Usability of Portfolio page	8.0 / 10.0

Jest has been used for mobile application testing. Snapshot testing has been run to ensure the UI does not change unexpectedly. It is essentially important as there are a variety of components have been used which may be easily altered their position.

3 Discussion

Those are the challenges and limitations we met in the project:

Hardware Resources Problem

The hardware resource is a big limitation to our project. Since our data is very dense and very frequent. The input data sequence would be huge if we want to look back the data from a long period. Precisely, there are 1440(minutes) input data sequences in our experiment. But if we want to look back for a longer period like 10 days it will take 1440 * 10 input sequences.

And those numbers are huge for LSTM models since the machine learning backend have to store every hidden state between the input sequences. And we do not have a graphic card which is able to manage that much of data (We use a graphic card with 8GB memory in this project). That has limited our machine learning model building in a way that we cannot look back to the data for a long period since our graphic card could not manage it.

Time Limitation

Since our project focus on providing useful information based on minute-wise data, the minute-wise data gives us a huge dataset for training. As the dataset is huge, we need to spend a lot more time to train a model. As a result, we need 2 hours to finish one round of hyperparameter optimization of a single model. The long training time limited us from trying some more data-processing techniques or making some more machine learning model enhancement.

Google Cloud Data Disappearance

The unexpected situation on Google Cloud would also restrict our project progress. Our group has faced that the entire dataset in MongoDB Database has disappeared. The reason why the database is clear is that the Google Cloud Platform trial period expired. Google stopped our virtual machine mandatorily when our data scraper is still working and it might cause data loss of the database. Thus, we needed to take more time to recover the data and reset our prediction server.

4 Conclusion

In this project, we have implemented three types of LSTM models which are Encoder-Decoder LSTM, Bidirectional Encoder-Decoder LSTM and Attention-Based Encoder-Decoder LSTM. Then we have evaluated their prediction performance with the minute-wise data on Bitcoin and Ripple. We found that the Encoder-Decoder LSTM model works the best in general. Our team have built a mobile application on top of our machine learning models' predictions. The mobile application is based on react native framework which allows us to implement a cross-platform application. The mobile application consists of three main features including Coin Price Predictor, Coin Portfolio and Push Notification. Coin Price Predictor allows user to access our 60-minute predicted price while they can access to our suggested coin portfolio by our Coin Portfolio feature. Users can easily get updated by our notification feature in different time intervals to make wise decisions.

For further development, we could improve the machine learning models by fitting in a longer period of the data as the machine learning models' input or examine the models by fitting in data in different time interval to generate a more comprehensive analysis on the cryptocurrency market.

5 References

[1] A. Chakrabarty, A. De, and R. Dubey, "A Flexible Approach Towards Multi-frequency Re-engineering of the Moving Average Convergence Divergence Indicator," Global Journal of Flexible Systems Management, vol. 15, no. 3, pp. 219–234, Nov. 2014.[Accessed: 15- Sep- 2018].

[2] A. Tsantekidis, N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Forecasting stock prices from the limit order book using convolutional neural networks", In *Business Informatics (CBI), 2017 IEEE 19th Conference,* Vol. 1, pp. 7-12.[Accessed: 16-Sep-2018]

[3] Burns, B. (2018). *Advantages of Day Trading*. [online] dummies. Available at: https://www.dummies.com/personal-finance/investing/advantages-of-day-trading/ [Accessed 17 Sep. 2018].

[4] Hacker Noon. (2018). *6 Reasons Why Cryptocurrency is Here to Stay – Hacker Noon*. [online] Available at: https://hackernoon.com/6-reasons-why-crypto-is-here-to-stay-ec9bd62c096 [Accessed 16 Sep. 2018].

[5] L.D. Persio and O. Honchar, "Artificial neural networks architectures for stock price prediction: comparisons and applications", *International Journal of Circuits, Systems and Signal Processing*, *10*, 2016, pp. 403-413.[Accessed: 16-Sep-2018]

[6] L. Sprung, "Here's how an investor can take advantage of market volatility", *CNBC*, 2018. [Online]. Available: https://www.cnbc.com/2018/04/02/how-investors-can-take-advantage-of-market-volatility.html. [Accessed: 17- Sep-2018].

[7] K. Michniuk, "Pattern recognition applied to chart analysis. Evidence from intraday international stock markets", 2018.

[8] K. Reddy, "Advantages and Disadvantages of Artificial Intelligence - WiseStep", *WiseStep*, 2018. [Online]. Available: https://content.wisestep.com/advantages-disadvantages-artificial-

intelligence/. [Accessed: 17- Sep- 2018].

[9] N. Indera, I. Yassin, A. Zabidi, and Z. Rizman, "Non-linear Autoregressive with Exogeneous input (narx) bitcoin price prediction model using PSO-optimized parameters and moving average technical indicators," Journal of Fundamental and Applied Sciences, vol. 9, no. 3S, p. 791, Jan. 2018.

[10] Pollock, D. (2018). *Volatility: The Necessary Evil Of Cryptocurrency And How To Handle It*. [online] Cointelegraph. Available at: https://cointelegraph.com/news/volatility-the-necessary-evil-of-cryptocurrency-and-how-to-handle-it [Accessed 16 Sep. 2018].

[11] Shengnan Yu, "Financial Time Series Analysis of Stock Data", 2018. [Online]. Available: https://www.cse.ust.hk/~rossiter/independent_studies_projects/lstm_stock_prediction/lstm_stock_prediction.pdf. [Accessed: 16- Sep- 2018].

[12] "Top 15 Advantages of Internet Marketing for Your Business", *EDKENT MEDIA*, 2018. [Online]. Available: https://edkentmedia.com/top-15-advantages-internet-marketing-business/. [Accessed: 17- Sep- 2018].

6 Appendix A: Meeting Minutes

6.1 Minutes of the 1st Project Meeting

Date: June 1, 2018

Time: 17:00

Place: Starbucks Coffee Shop

Present: TANG Marco Kwan Ho, FONG Chi Chung, SUEN Ka Chun

Absent: Prof. David Paul ROSSITER

Recorder: FONG Chi Chung

1. Approval of minutes

    This was the first formal group meeting, so there were no minutes to approve.

2. Report on progress

    2.1 Every team members have read all the Final Year Project documents and know about the overall workflow.

    2.2 Ka Chun and Kwan Ho have read through the machine learning details on implementing on predicting financial price.

    2.3 Chi Chung has gone through some financial books, concluding technical analysis should be useful in this project.

    2.4 All members try to be familiar with the goal and the objective of Final Year Project

3. Discussion items

    3.1 We have decided to do on some volatile commodities while cryptocurrency and stocks are preferred.

    3.2 We have discussed the model for machine learning, while CNN and LSTM are on the list.

    3.3 Preparation for presenting ideas to Prof. Rossiter has been done.

    3.4 We have made a plan of reading references and online materials.

4. Goals for the coming meeting

   4.1 The group should meet with Prof. Rossiter this week.

   4.2 To explore more categories of machine learning models

   4.3 Narrow down the goal of our Final Year Project

5. Meeting adjournment and next meeting

   5.1 The meeting was adjourned at 18:30

   5.2 The next meeting will be at 19:00 on June 2, 2018.

6.2   Minutes of the 2nd Project Meeting

Date: June 2, 2018

Time: 11:00

Place: Room 3554, HKUST

Present: Prof. David Paul ROSSITER, TANG Marco Kwan Ho,

FONG Chi Chung, SUEN Ka Chun

Absent: None

Recorder: FONG Chi Chung

1. Approval of minutes

    The minutes of the last meeting were approved without amendment.

2. Report on progress

    2.1 Kwan Ho explain the idea of our Final Year Project Goal: Build a algo-
    trading system through machine learning generated indicators

    2.2 Ka Chun and Kwan Ho have read through the machine learning details on
    implementing on predicting financial price.

    2.3 Chi Chung keep a focus on financial analysis materials

3. Discussion items

    3.1 Professor suggest us in a different way for the Final Year Project

    3.1.1   What is the aim of our project: Professor suggested us to think more
    board to consider what actually we want to find out in the Final Year
    Project

    3.1.2   Where we can get the data for training model: The most important
    thing of machine learning is getting the training data, the professor
    reminded us that we should check the platform of stock price API and
    compare which one is the best for our Final Year Project

    3.1.3   Social Media and News Detection: In general, news are one of the
    factors to make the fluctuating price, take the advantage of news may
    help us to build a precise model

4. Goals for the coming meeting

4.1 To evaluate and reflect the points that the professor suggested

4.2 Kwan Ho tries to use some companies' API and test the performance of collecting data.

4.3 Have more research on our topics

5. Meeting adjournment and next meeting

5.1 The meeting was adjourned at 12:00

5.2 The next meeting will be at 19:00 on August 3, 2018.

6.3 Minutes of the 3rd Project Meeting

Date: August 3, 2018

Time: 19:00

Place: Pacific Coffee Shop

Present: TANG Marco Kwan Ho, FONG Chi Chung, SUEN Ka Chun

Absent: Prof. David Paul ROSSITER

Recorder: SUEN Ka Chun

1. Approval of minutes

   The minutes of the last meeting were approved without amendment.

2. Report on progress

   2.1 Kwan Ho is focusing on stock market API.

   2.2 Ka Chun is finding some useful reference for machine learning.

   2.3 Chi Chung is trying to find some financial analysis techniques.

   2.4 All group members have thought about ways to develop the financial system
       and some suggestions were mentioned.

3. Discussion items

   3.1 The group considered if this project is suitable for the whole period of the
       FYP.

   3.2 Kwan Ho tested Futu API, which is a security company.

   3.3 Possible financial themes were discussed.

   3.4 Fong Chi Chung tried to teach the groupmate how to use Bloomberg.

4. Goals for the coming meeting

   4.1 All group members will read more information related to the project topic, e.g.
       machine learning models, trading indicators.

   4.2 All group members will need to study and compare the languages and library
       being considered for implementation of the project.

   4.3 All group members will consider the suitable financial themes.

4.4 Chi Chung will try to start to analysis which trading indicators are suitable for our project.

5. Meeting adjournment and next meeting

5.1 The meeting was adjourned at 21:30.

5.2 The next meeting will be at 19:00 on September 3, 2018.

6.4   Minutes of the 4th Project Meeting

Date: September 3, 2018

Time: 19:00

Place: Pacific Coffee Shop

Present: TANG Marco Kwan Ho, FONG Chi Chung, SUEN Ka Chun

Absent: Prof. David Paul ROSSITER

Recorder: SUEN Ka Chun

1. Approval of minutes

    The minutes of the last meeting were approved without amendment.

2. Report on progress

    2.1 Kwan Ho is developing the algorithm and trying to use Futu API to collect HSI financial data.

3. Discussion items

    3.1 All the teammate point out their interested topic on machine learning

    3.2 Evaluate the different machine learning model

    3.3 Cryptocurrency is discussed and we want to focus on it. The reason for using cryptocurrency is that we want to use fluctuating price pattern as one of the machine learning model data sources.

    3.4 Fong Chi Chung tried to teach the groupmate how to use Bloomberg again.

    3.5 Start to do the proposal

4. Goals for the coming meeting

    4.1 Finalize our Final Year Project topic

    4.2 Chi Chung will try to start to analysis which trading indicators are suitable for our project.

    4.3 Finish the most important part of the proposal

5. Meeting adjournment and next meeting

    5.1 The meeting was adjourned at 21:30.

5.2 The next meeting will be at 19:00 on September 18, 2018.

6.5   Minutes of the 5th Project Meeting

Date: September 18, 2018

Time: 17:00

Place: Pacific Coffee Shop

Present: TANG Marco Kwan Ho, FONG Chi Chung, SUEN Ka Chun

Absent: Prof. David Paul ROSSITER

Recorder: SUEN Ka Chun

1. Approval of minutes

    The minutes of last meeting were approved without amendment.

2. Report on progress

    2.1 We are working on the proposal

3. Discussion items

    3.1 Kwan Ho presented our Final Year Project goal to professor

    3.2 Professor pointed out our mistakes on the proposal, the most important part was that our proposal was the lack of logic. This is the weakness if we just represent the Final Year Project to people.

    3.3 Ka Chun jotted down the advice of professor and prepared to provide a clear objective for the teammate to correct the proposal

4. Goals for the coming meeting

    4.1 Finalize the Final Year Project proposal

    4.2 Review the proposal and submit the proposal to the department

5. Meeting adjournment and next meeting

    5.1 The meeting was adjourned at 18:00.

    5.2 The next meeting will be arranged through email

6.6 Minutes of the 6th Project Meeting

Date: October 18, 2018

Time: 17:00

Place: Pacific Coffee Shop

Present: TANG Marco Kwan Ho, FONG Chi Chung, SUEN Ka Chun

Absent: Prof. David Paul ROSSITER

Recorder: SUEN Ka Chun

1. Approval of minutes

   The minutes of last meeting were approved without amendment.

2. Report on progress

   2.1 We are working on the proposal

3. Discussion items

   3.1 Kwan Ho presented our Final Year Project goal to professor

   3.2 Professor pointed out our mistakes on the proposal, the most important part
   was that our proposal was the lack of logic. This is the weakness if we just
   represent the Final Year Project to people.

   3.3 Ka Chun jotted down the advice of professor and prepared to provide a clear
   objective for the teammate to correct the proposal

4. Goals for the coming meeting

   4.1 Finalize the Final Year Project proposal

   4.2 Review the proposal and submit the proposal to the department

5. Meeting adjournment and next meeting

   5.1 The meeting was adjourned at 18:00.

   5.2 The next meeting will be arranged through email

6.7 Minutes of the 7th Project Meeting

Date: November 30, 2018
Time: 14:10
Place: Room 3554, HKUST
Present: Prof. David Paul ROSSITER, TANG Marco Kwan Ho, FONG Chi

Chung, SUEN Ka Chun

Absent: None
Recorder: SUEN Ka Chun

1. Approval of minutes
   The minutes of the last meeting were approved without amendment.

2. Report on progress
   2.1 Kwan Ho is showing the machine learning result to Prof. David Paul
       ROSSITER

3. Discussion items
   3.1 Discuss on how to come up with a good metrics to measure the performance
       of our model
   3.2 Illustrate the result of our machine learning model
   3.3 Discuss on the progress report

4. Goals for the coming meeting
   4.1 Came up with a way to do performance evaluation
   4.2 Having a plan to improve the machine learning model

5. Meeting adjournment and next meeting
   5.1 The meeting was adjourned at 14:40.

6.8   Minutes of the 8th Project Meeting

Date: January 11, 2019

Time: 14:10

Place: Room 3554, HKUST

Present: Prof. David Paul ROSSITER, TANG Marco Kwan Ho, FONG Chi
                 Chung, SUEN Ka Chun

Absent: None

Recorder: SUEN Ka Chun

1. Approval of minutes

    The minutes of the last meeting were approved without amendment.

2. Report on progress

    2.1 Kwan Ho is showing the progress on the database and machine learning
        models Prof. David Paul ROSSITER

    2.2 Show what have we done on the attention model

3. Discussion items

    3.1 Discuss alternative topic which could enrich our FYP

    3.2 Discuss on the performance of our machine learning model

4. Goals for the coming meeting

    4.1 Came up with a alternative way to make the FYP to be better

    4.2 Having a plan to improve the machine learning model

5. Meeting adjournment and next meeting

    5.1 The meeting was adjourned at 14:40.

6.9 Minutes of the 9th Project Meeting

Date: February 14, 2019

Time: 18:00

Place: Room 3554, HKUST

Present: Prof. David Paul ROSSITER, TANG Marco Kwan Ho, FONG Chi Chung, SUEN Ka Chun

Absent: None

Recorder: SUEN Ka Chun

1. Approval of minutes

    The minutes of the last meeting were approved without amendment.

2. Report on progress

    2.1 Show the plan of the mobile application development

    2.2 Evaluate the progress report

3. Discussion items

    3.1 Discuss how improve the progress report

    3.2 Discuss the upcoming progress of FYP

4. Goals for the coming meeting

    4.1 Finalize the FYP and show the results of FYP

    4.2 Having a mock presentation of FYP

5. Meeting adjournment and next meeting

    5.1 The meeting was adjourned at 18:30.

6.10 Minutes of the 10th Project Meeting

Date: April 15, 2019

Time: 17:00

Place: Room 3554, HKUST

Present: Prof. David Paul ROSSITER, TANG Marco Kwan Ho, FONG Chi

                  Chung, SUEN Ka Chun

Absent: None

Recorder: SUEN Ka Chun

1. Approval of minutes

   The minutes of the last meeting were approved without amendment.

2. Report on progress

   2.1 Show the final result of our FYP

   2.2 Evaluate the final report

3. Discussion items

   3.1 Discuss how improve the final report

   3.2 Discuss the improvement of the presentation

4. Goals for the coming meeting

   4.1 Present the FYP in demo presentation

5. Meeting adjournment and next meeting

   5.1 The meeting was adjourned at 17:30.

7 Appendix B: Project Planning

7.1. Distribution of Work

Task	TANG Marco Kwan Ho	FONG Chi Chung	SUEN Ka Chun
Do the Literature Survey	○	●	○
Gather data from dataset scraper	○	○	●
Design and set up database	○	○	●
Design machine learning models	●	○	○
Data preprocessing	●	○	○
Input normalization	●	○	○
labels preprocess on machine learning models	●	○	○
Implement the machine learning models	●	○	○
Implement the mobile applications backend	○	●	○
Implement the authorization of mobile application	○	○	●
Implement the layout of mobile application	○	●	○
Test and evaluate the performance of dataset scraper	○	○	●
Test and evaluate the performance on machine learning models	○	●	○
Test and evaluate the performance of the mobile application	●	○	○
Write the Proposal	○	●	○

Write the Monthly Reports	○	●	○
Write the Progress Report	●	○	○
Write the Final Report	●	○	○
Prepare for the Presentation	○	○	●
Design the Project Poster	○	○	●

● Leader ○ Assistant

7.2. GANTT Chart

Task	July	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr
Do the Literature Survey	■	■								
Gather data from dataset scraper					■	■	■			
Design and set up database		■	■							
Design machine learning models			■	■	■					
Data preprocessing			■	■	■	■				
Input normalization				■	■	■	■			
labels preprocess on machine learning models				■	■	■				
Implement the machine learning models				■	■	■	■	■		
Implement the mobile applications backend								■	■	■
Implement the authorization of mobile application						■	■			
Implement the layout of mobile application						■	■			
Test and evaluate the performance of dataset scraper						■	■			
Test, evaluate and optimize the performance of machine learning models							■	■	■	

Test and evaluate the performance of the mobile application							▓	▓	▓	▓
Write the Proposal			▓							
Write the Monthly Reports			▓	▓	▓	▓		▓		
Write the Progress Report								▓		
Write the Final Report										▓
Prepare for the Presentation										▓
Design the Project Poster										▓

▓   the tasks which have been done

8 Appendix C: Required Hardware & Software

8.1 Hardware

Machine learning deployment PC: Ubuntu with Nvidia RTX 2070

8.2 Software and Libraries

Python 3.6	Programming Language
TensorFlow 1.2.1	Machine Learning Framework
Keras	Machine Learning Framework
VSCode	IDE
NumPy	Data processing Library
pandas	Data processing Library
matplotlib	Python Plotting Library
scikit-learn	Machine learning library
ta-lib	Technical analysis library

Mobile Application development package:

HTML5	Web Markup Language
CSS3	HTML Style Language
React.js	User Interface Library
Material.UI	User Interface Library
Redux	React storage Library
React Native	Mobile Application Framework
React Navigation	Mobile Application Framework
Victory-Native	Data Visualization Library