FORECASTING FUTURE DAILY RETURNS WITH LSTM MODELS

Shawn E Samudra

September 15, 2021

ABSTRACT

This is an attempt to forecast future daily returns with lstm models. In this attempt, we will discuss:

- How we will approach the problem.
- The ASX historical prices used for our model.
- Interpret and compare the results from our models.
- Discuss the possible strategies from the models.
- Make recommendations for future improvements of the models.

1 Section 1 - Introduction

This report is an attempt to find out how good can LSTM models forecasts daily future returns. In this report, we will discuss two approaches to this problem, modeling with both daily close prices and future daily returns as the target variable. There will be three sections to our solution attempt; data processing, exploratory data analysis (EDA), and forecasting. However, you will notice that there are some EDA in the forecasting section as well. This is due to our solution attempt in which the first two sections; data processing and EDA are solved for the whole dataset. As for the forecasting, we will be focusing on one ticker that has been processed in the first two steps. Which will then be used to model our data.

2 Section 2 - Data

The data set for this attempt are ASX stock prices from Jan 2015 to June 2018. Each file contains end of day (EOD) data in the following format: Ticker, Date, Open, High, Low, Close, and Volume.

- Ticker: Code referring to a particular stock
- Date: the trading day corresponding to this row
- Open / High / Low / Close: Prices in dollars corresponding to the opening / highest / lowest / closing price for specified ticker on the specified trading day
- Volume: Total number of shares traded on the specified trading day

3 Section 3 - Data Processing

In this section, we will discuss the data processing steps and other feature engineering steps that have been taken for the whole dataset.

We first convert the EOD data into five separate time series data frames. One for each attribute that was mentioned in Section 2. In each data frame, rows are indexed by date and columns by ticker. Each column of the original dataset was pivoted to solve this task.

Next, we then create a dataframe with the future close returns as defined by the equation:

$$r_{t,t+1} := \frac{P^c_{t+1}}{P^c} - 1, \tag{1}$$

Where $r_{t,t+1}$ is the future 1 future close return at time t and P^c_{t+1} is the close price at time t.

We also create a data frame of the intraday close return with the following equation:

$$r^I_t := \frac{P^c_t}{P^o_t} - 1, \tag{2}$$

Where r^I_t is the intraday close return at time t, P^c_t is the close price at time t and P^o_t is the open price at time t.

Following the same manner from the intraday close return, we also created one more data frame intraday volume change with a lag of one day. We will assume that giving a lag of one day will give the model information on the previous' day volume spike or drop.

Finally, we create another data frame with the daily high and low ratio $\frac{High}{Low}$, by dividing the daily high and daily low prices for each day.

In total we have nine data frames from this data processing section.

4 Section 4 - Exploratory Data Analysis

In this section, we will discuss on how we will handle data coverage, data quality and how to handle time series data.

4.1 Coverage

We first check the original dataset and found no missing values. This means that in the original dataset all the entries are good. We can now assume that for every row, if any data in one particular column is present, the data on the other remaining columns will also be present. Thus, coming back to the nine data frames that we have obtained, we only need to check on one data frame to ensure that the other data frames data are good.

4.1.1 Missing Values

From the data separation and pivot from the original dataset, we need to remember that although all entries have good data integrity, it does not mean that there is an entry for each date of every ticker. We first check for missing values in one of the data frames and found that there are 1128562 missing values which is about 46% of the dataset. There is also at least one missing value from every row in the dataset.

There are mainly two reasons for the missing values. Firstly, tickers that are just newly listed or delisted to ASX on the time frame that we are working on. Secondly, some of the data are actually missing; which means, in some tickers, data is not entered every working day.

For the newly listed or delisted, we will not do anything to the missing values as the missing values are missing for a good reason. However, for the other reason, we will need to either drop the ticker or fill in the missing values.

We first need to determine a condition to either drop the ticker or fill in the missing values. In this attempt, we will assume that if a ticker data is missing for two consecutive days, it is no longer workable for our model. Otherwise, we fill the data with the previous price assuming that there will be no significant price difference in the span of two days.

From 883 tickers, we are now left with 540 tickers that have workable data.

4.2 Data Quality

The goal of this section is to find suspect values.

4.2.1 Negative Values

We are not expecting negative stock prices in this data. Thus, we check if any of the value is negative and found no entries. This means that the data passed the negative value condition.

4.2.2 Large Values

In order to check the condition for large values, we will utilise the high low ratio data frame. We first plot the data in a scatter plot which can be seen at Figure 2.

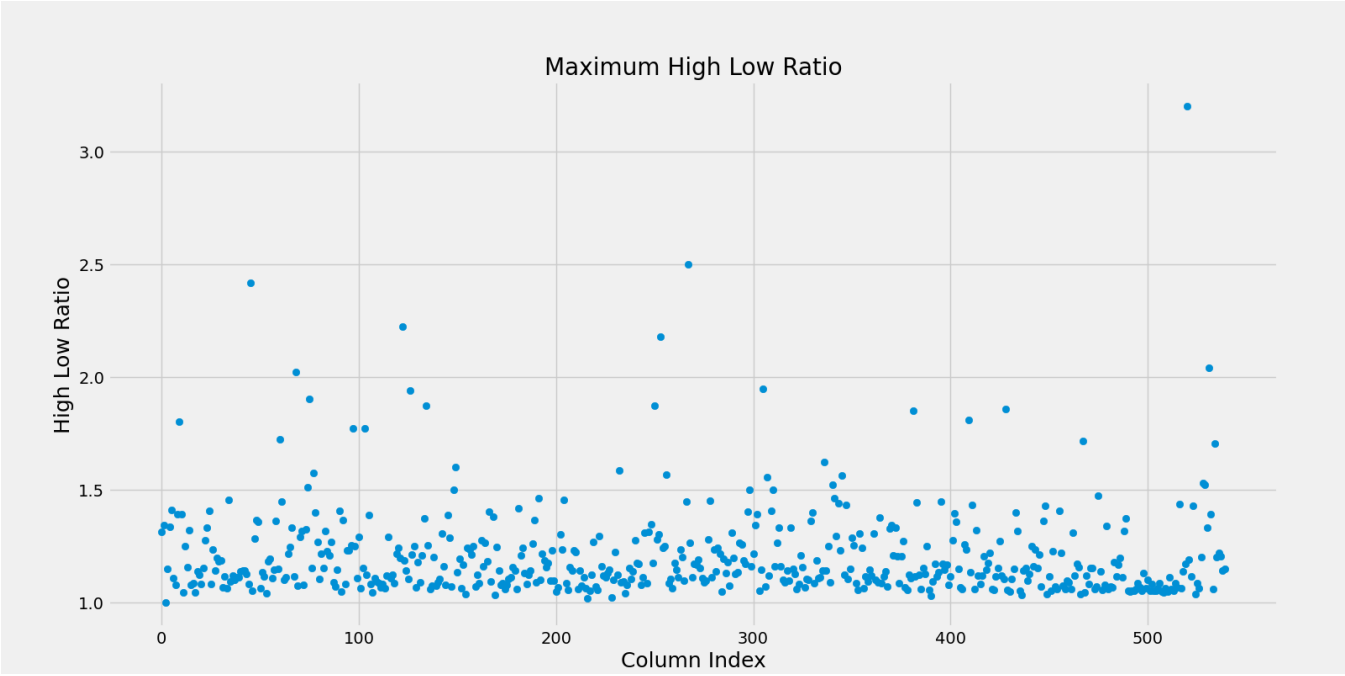

Figure 1: Maximum High Low Ratio for every ticker.

We can Identify one point in which the high low ratio is larger than 3. We then can identify the point to be the ticker "BMB" on the date "2015-02-13". Thus to verify if the data is correct, we will compare the high low ratio to the volume. From figure 2 we can see that there is a spike in the volume on the date "2015-02-13". Furthermore we can compare that these prices match the prices from external an source [2].

	BMB
Date	
2015-02-12	529440.0
2015-02-13	13058877.0
2015-02-16	1745012.0
2015-02-17	65000.0
2015-02-18	412846.0

Figure 2: Maximum High Low Ratio for every ticker.

4.3 Time Series

4.3.1 Time Series in General

Time series or a collection of well-defined data items through repeated measurements over time [1] is commonly used to analyse or forecast weather, inventory stocks, stock prices and etc. An observed time series can be decomposed into three components:

the trend (long term direction), the seasonal (systematic, calendar related movements) and the irregular (unsystematic, short term fluctuations)[1].

4.3.2 Time Series in Stock Markets

In the stock markets, trends and seasonal effects are the ones we want to focus on as irregular events such as random CEO tweets, commodity price spike due to a external events and natural disasters are not predictable. We cannot foresee such events and it is unlikely that our model will be able to do so. Thus, we would like to focus on stocks that are predictable, having good seasonality and stable trends.

5 Section 5 - Forecasting

5.1 Choosing a Ticker by Random

For this attempt we will be choosing a random ticker. The ticker chosen was AAD. We joined the 5 data frames together and rename the columns according to the relevant data frame data.

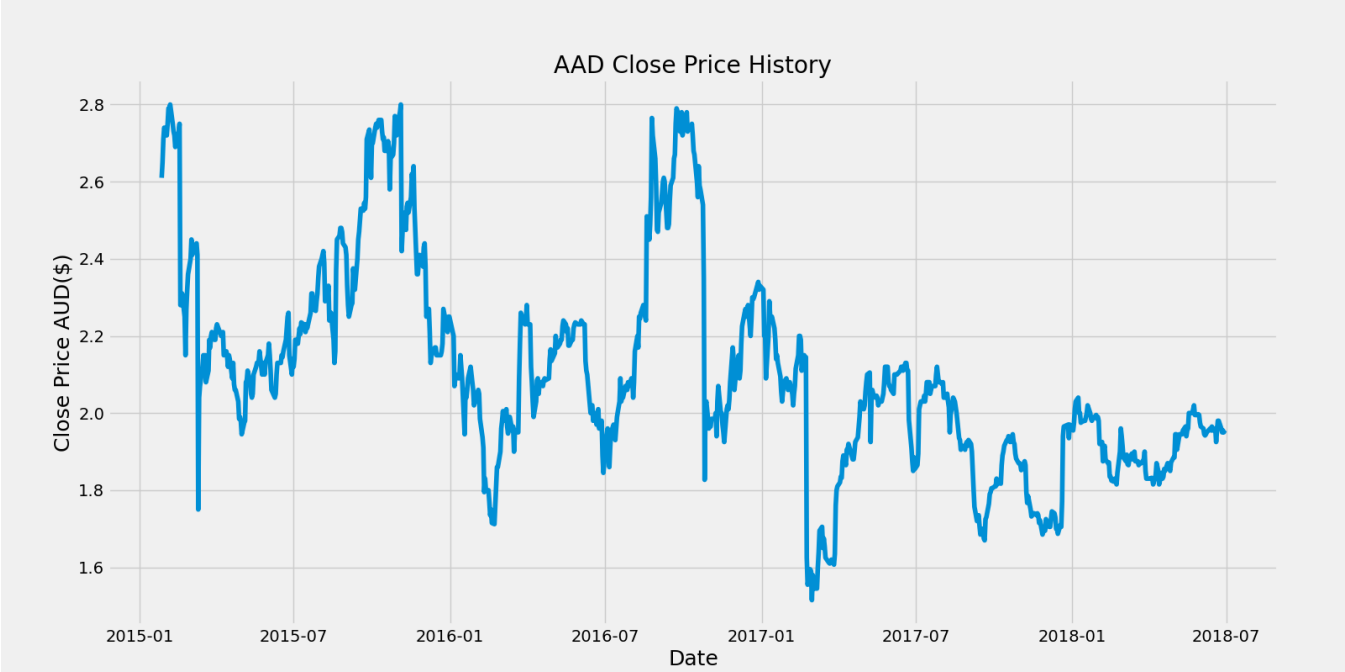

Figure 3: Historical Prices of AAD from Jan 2015 to Jun 2018.

5.2 LSTM Model With Closing Prices

In this approach we decided to use the closing prices as we assume that it prices the events that have happened in the current day. Events can be dividends, news, government announcements and all other factors that may affect the price of the particular stock ticker.

5.2.1 Choosing Appropriate Columns

From Section 3, we have obtained nine features. We then plot these features to a correlation heat map. From figure **??** we can see that open, high and low are the most correlated to close prices. However, these are expected as daily prices tend to not change significantly. So we will take the second-best group, which are VolumeChangeLag1 and FutureReturn1 with an absolute correlation of around 0.4. We assume that these two features will help our model forecast future returns.

5.2.2 Model Architecture

As we are using LSTM to model our data, which requires a 3D input, we need to first transform our data into batches of a certain length. We will choose the length to be 21 days as it is the average number of trading days in a month. In which the target value will be the prices on the 22nd day.

5.2.3 Model Result

For our model, we have chosen two LSTM layers with 50 nodes which then is passed through a dense layer with 25 nodes and a final single dense layer. Which was then run through 10 epochs with a batch size of 1 and validation split of 0.1. The hyperparameters were chosen arbitrarily to give us a baseline for our hyperparameter tuning. We achieved a root mean squared (rmse) of 1.5869.

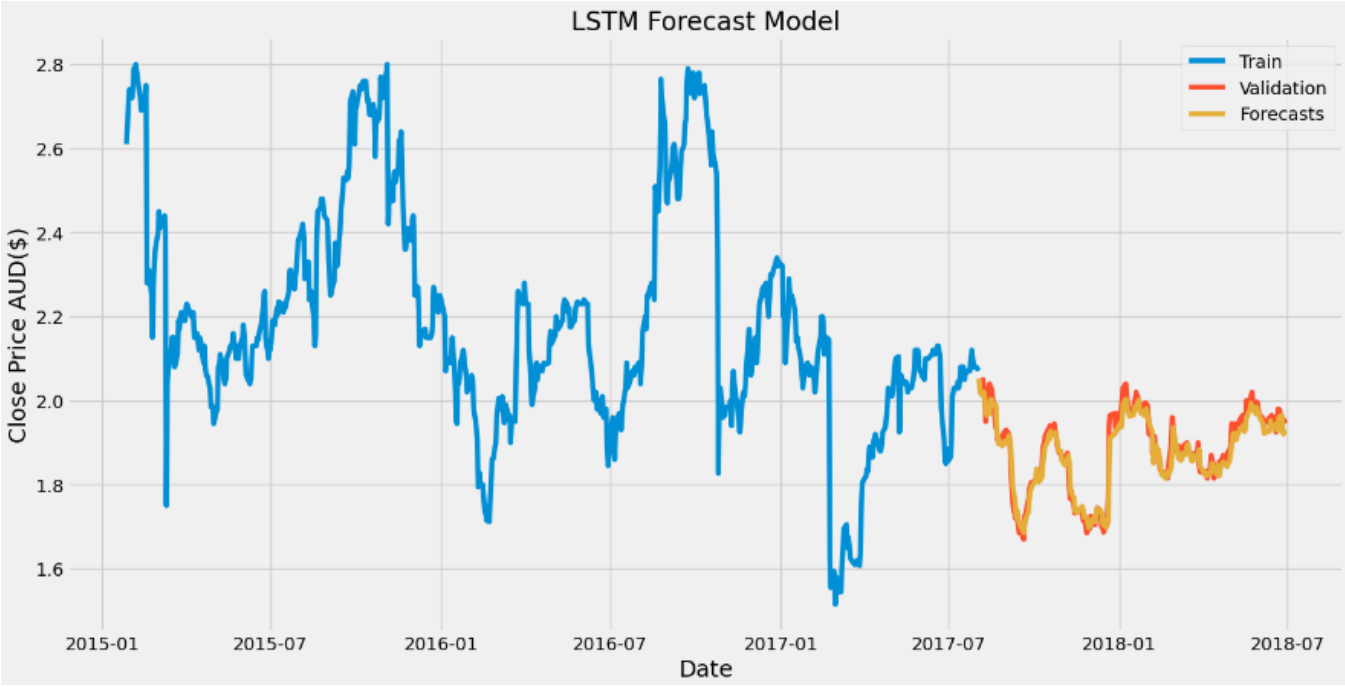

Figure 4: Arbitrary Hyperparameter model with close price as a target value.

5.2.4 Hyperparameter Tuning

We then tune the hyperparameters with a bayesian optimization algorithm. For the LSTM layers, we chose a min value of 32, max value of 128 with a step of 32. The dense layer's min value is 32, max value 128, and step of 32. Due to time limitations, we then run the models with the same epochs, batch size and validation split. The best final model generated a rmse of 1.6307 which was actually worse than our benchmark.

5.2.5 Limitations of Using Close Prices

From figure 4 we can see that the forecasted prices are actually lagging behind the validation prices. This means that our model is not good as we want our model to be predictive instead of forecasting previous historical prices.

5.3 LSTM Model With Future Returns

5.3.1 Model Architecture

Due to limited time resources, we build this model with the same architecture with the close prices model.

5.3.2 Model Result

The new model generated a lower rmse of 0.6072 when compared to the previous models. This shows that our model is able to forecast the daily returns.

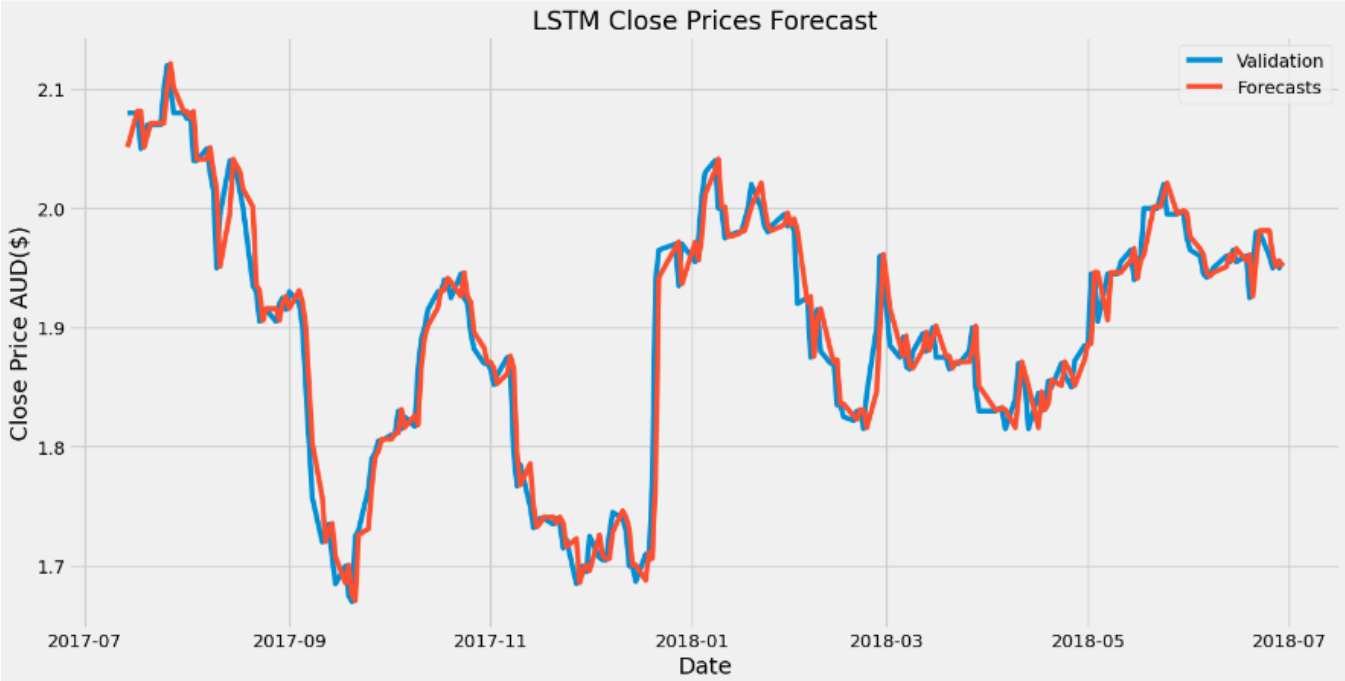

Figure 5: Validation of LSTM with future return

However, after further analysis of the model's forecasts, we see that the model is biased. The model only generated positive forecasts. Although it performed better based on rmse, it does not give us any information on the direction of where the market is heading to.

5.3.3 Strategy for the model

Although our model here is much better than the previous ones, we have not generated a model in which it can forecast prices in a leading manner. Unfortunately, creating a strategy with this model is yet to be feasible.

5.4 Future Improvements

There are many ways to improve this model such as introducing more LSTM layers to add complexity, using other target variables, and hyperparameter tuning using different optimization algorithms. We would like to focus on the target variables as from this attempt, we have found that changing the target variable improves the model drastically.

5.5 Conclusion

This attempt in creating a model to forecast future returns have not yielded a good enough result to create a profitable trading strategy. However, more future improvements can help the model learn and yield a better leading model.

References

[1] https://www.abs.gov.au/websitedbs/d3310114.nsf/home/time+series+analysis:+the+basics

[2] https://au.investing.com/equities/balamara-resources-ltd-historical-data