DATA ACQUISITION AND PROCESSING SYSTEMS ELEC0136 22/23 REPORT

SN: 12345678

ABSTRACT

This project simulates a real-life data-science situation that can be approached using the process including data acquisition, storing, preprocessing, exploration and inferring. The data is acquired by relevant application programming interface and python functions for sending request. Then the acquired data is stored both locally and in the cloud. The data could be retrieved from local disk and cloud database for the downstream preprocessing including data cleaning, visualization and transformation. Next, the exploratory data analysis is conducted to find potential pattern of the data and hypothesis testing is carried out to better understand the composition of your dataset and its representativeness. Finally, the preprocessed data is feed as input to train and test the LSTM model which could be later used to predict the trend of stock values in the future.

Index Terms— Data acquisition, storage, data preprocessing, exploratory data analysis, machine learning

1. INTRODUCTION

This report illustrates the process of how the data is acquired, stored, preprocessed, analyzed and finally used to fit a machine learning model. In the start, we first describe the details of data which is used for this project. Stocks dataset is the main dataset required because the task is to select one company and study their market trends to ultimately be able to advice on when and whether an investor should buy, hold, or sell this stock. Specifically, stocks data for AAL is chosen and weather data and covid data for New York are collected to be used as auxiliary data. Both weather condition and covid could have an impact on airline cause, thereby may influence the trend of stocks. For each of these data set, content, size and format of the data is clearly described in data description section. The data is acquired by relevant application programming interface including Yahoo! Finance, NOAA, and CDC API. Python functions are called to send HTTP request and receive the corresponding response containing the objective data. Then the acquired data is stored both locally and in the cloud. For local storage, the data is stored in JSON files on disk. For cloud storage, we use MongoDB cloud database. The data could be retrieved from local disk and cloud database for the downstream preprocessing including data cleaning, visualization and transformation. From data visualization, it could be noticed that there are some missing values and outliers in the dataset so data cleaning techniques including binning and filtering are applied to the original data. Next, the exploratory data analysis is conducted to find potential pattern of the data and hypothesis testing is carried out to better understand the composition of your dataset and its representativeness. Finally, we build two machine learning models by using stocks data only and using all collected data respectively. The model using LSTM architecture is developed by training and testing and it could be later used to predict the stock values in the future.

2. DATA DESCRIPTION

The stocks dataset collected in this project is the stocks data of American Airlines Group Inc (AAL). The stocks dataset contains four features which are open, high, low and close value for a given time slot. Besides, this project uses weather data in New York (NY) which is the most populated city in the United States. The reason of using weather data in NY is because airline services are easily influenced by local weather. If there is heavy rain or snow, less people will buy air tickets sold by airline companies like AAL considering the inconvenience caused by possible flight delay or even cancellation. The weather dataset contains 5 features including precipitation, snowfall and other statistical features in meteorology. Another auxiliary dataset used is covid-19 data in NY. As it has been proved in the late three years, covid-19 pandemic could influence airline services to a very large extent. With the explosion of covid-19, there could be a salient drop of the number of travelers and working stuff in the airport. Meanwhile, the number of flights in NY might decline sharply if there are many infect case and national or international flights are canceled due to the safety concerns. The features of covid data includes total cases and total death. The data format for the features of each dataset is float numbers.

3. DATA ACQUISITION & STORAGE

3.1. Data Acquisition

The stocks data could be acquired by using Yahoo! Finance API. It is relatively convenient to call Finance API functions in python to obtain historical stocks data. By specifying parameters like date and interval in the API

function, we could easily acquire stocks data in a given period with specific time interval.

To acquire the external data, HTTP request is used to send the relevant request to the application programming interface (API). In particular, the weather data could be acquired through National Oceanic and Atmospheric Administration (NOAA) API. Concretely, it could be done by sending a HTTP request using python requests module. The HTTP request should follow the instructions of NOAA API which defines the standards of constructing a request. The covid-19 data could be acquired by using API provided by Centers for Disease Control and Prevention (CDC). The calling methods for the CDC API is very similar to NOAA API, except that the parameters in the HTTP request should follow the standards defined by CDC API. It is very convenient to call the NOAA and CDC API to obtain the weather data and covid data for a specific period of time and city respectively.

3.2. Data Storage

After data acquisition, both stocks data and external data should be stored for the downstream data preprocessing task. In this project, the data is stored both in local file and in cloud database.

For local storage, the data is stored in JSON format by simply calling storing function provided by python module called json. The three datasets are stored in "stocks_data.json", "weather_data.json", and "covid_data.json" respectively.

For cloud storage, MongoDB cloud database is used to store the data. MongoDB is a source-available cross-platform document-oriented database program and classified as a NoSQL database. There are three collections containing data for stocks, weather and covid in the database called "daps_data" created for this project particularly.

4. DATA PREPROCESSING

This section includes procedure that first cleans the data from missing values and outliers then provides useful visualisation of the data and finally transforms the data using normalization to improve the forecasting performance.

4.1. Data Cleaning

We first conduct data cleaning by interpolating to deal with missing values.

For stocks dataset, the stocks information is missing for weekends. Since the dataset is in a form of a time-series data, there should be a unified interval in terms of time for the original stocks data, before feeding it into machine learning model to train and predict stocks close value. The date column in the original dataset for stocks is set as index in order to conveniently interpolate, and more importantly, make data tidy and easy to understand. In particular, no clear seasonality for close value is detected, thereby linear interpolation is implemented for missing values of weekends. After interpolation, the interval of time for each datapoint is exactly one day.

For weather data, the feature like the station where the meteorology data is collected is deleted because it is not beneficial for characterizing the data point itself. Then we inspect missing value distribution of weather data. It is noticed that a large amount of data in the year 2017 and 2018 is missing. The imputing method is applied to deal with missing data problem. For the specific date when the data is missing, the average of the data on the same date in other years is calculated and imputed to the original weather dataframe. Then to smooth the dataset, binning method is conducted to the dataframe after imputation.

There is also data missing problem for covid-19 dataset. The original covid dataset only contains the data after 2019 when the covid-19 pandemic began, which means the data between 2017 and 2019 is missing. It is assumed that there is no covid case before there is a statistical investigation about covid. Therefore, all features between 2017 and 2019 of covid dataset are set to zero. It is also noticed during data visualization process that there exists some obvious outliers in the feature including "new_case" and "new_death". In order to benefit further procedure, so it is necessary to eliminate the outliers. In particular, we use the gaussian filter to smooth the original data.

4.2. Data Visualization

After data cleaning, data visualization is provided to better understand the dataset itself.

The close value for stocks data is plotted in Fig. 1.

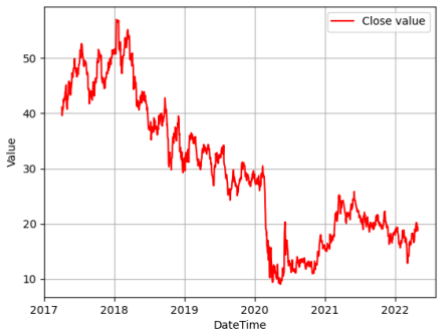

Fig. 1 Close value of stock over time

To better examine the seasonality of stocks close value, the close value for each year is plotted separately in Fig. 2. It suggests that there could be no salient seasonality for the close value of stock.

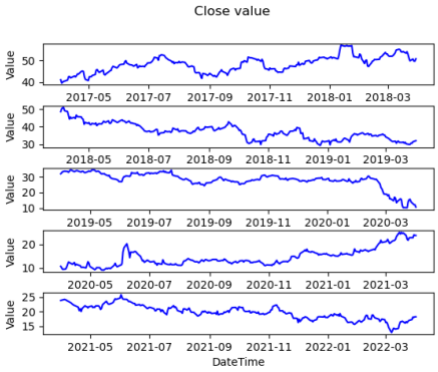

Fig. 2 Close value curve shown in separate years

Then we provide boxplot of the close value in Fig. 3. It is suggested that there is no outliers for close value feature and fifty percent of data falls in approximate range of twenty to forty.

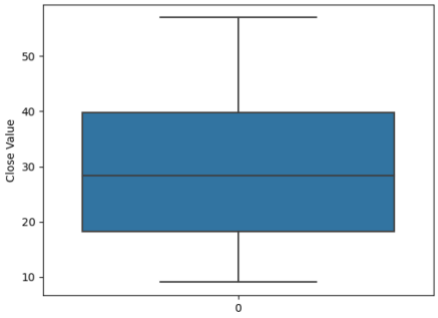

Fig. 3 Boxplot of Close value of stock

Also, z-score for close value is plotted below in Fig. 4. The value for each datapoint is below 3.0, which suggests that the values in the whole dataset are relatively smooth.

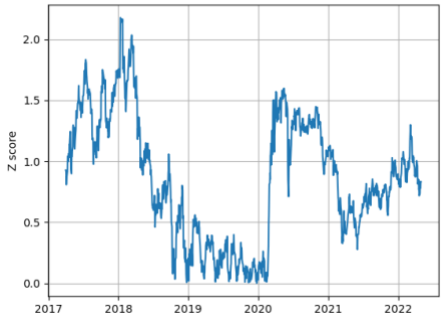

Fig. 4 Z-score of close value of stock

For weather dataset, we first plot the distribution of missing values for each feature. As it is shown in Fig. 5, that there is a large number of data points which lacks values for some features like SNOW and WESD.

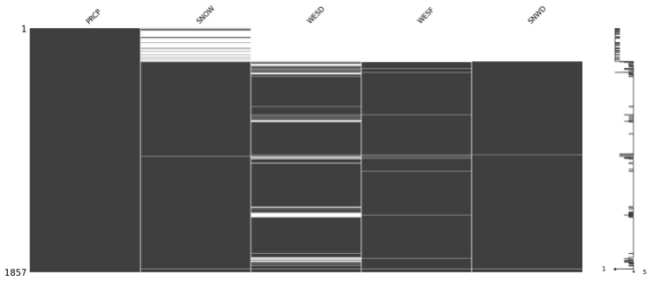

Fig. 5 Missing value distribution of weather data

Here the snow value of weather data is plotted in Fig. 6. The measured value for snow feature is missing to a large extent before 2018.

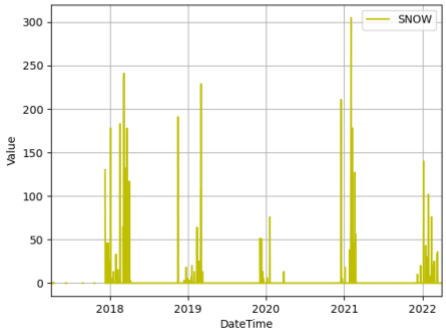

Fig. 6 Original snow value of weather data

After imputation, the feature value for snow is available every day from April 1st 2017 to March 31st 2022 as it is shown in Fig. 7.

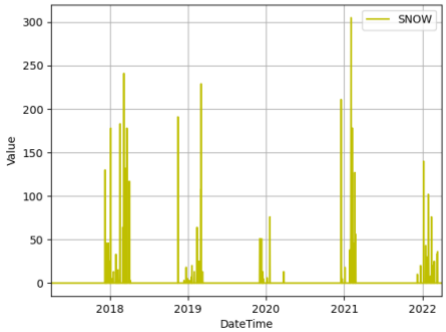

Fig. 7 Imputed snow value of weather data

To further reduce the noise in this time series data, binning is applied, and the generated data is shown in Fig. 8.

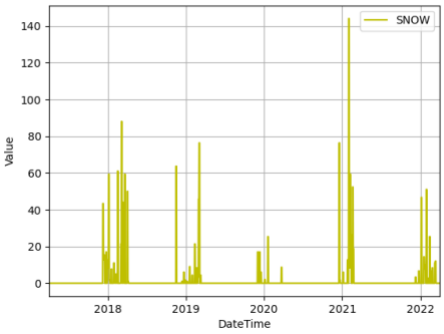

Fig. 8 Imputed and binned snow value of weather data

For covid data, new contaminated case for each day is shown in. As it is shown in Fig. 9, there exists some outliers with the most obvious one around the date at the end of 2021.

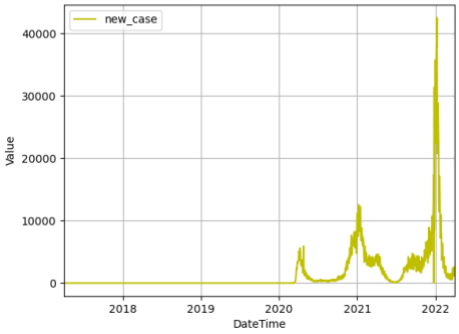

Fig. 9 Original new case value of covid data

Then gaussian filter is applied to original data (new case) to exterminate outliers. As we could observe in Fig. 10, the curve is smoothed.

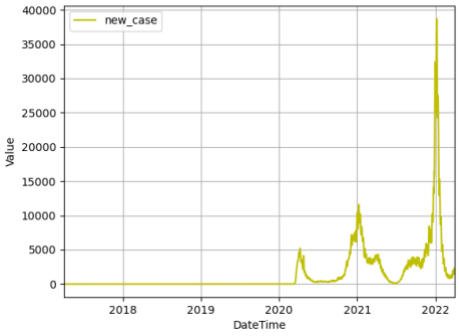

Fig. 10 Smoothed new case value of covid data

4.3. Data Transformation

Normalization is a effective method to use when the distribution of the data is not known or not Gaussian.

It is necessary to conduct normalization to the data because the feature values in the acquired three datasets has varying scales and the algorithm in the later data inference process does not make assumptions about the distribution of the data.

Therefore, to improve the forecasting performance, data normalization is used, and its concrete mathematical form is shown below (suppose x is one of the features in a dataset)

$$x_{norm} = \frac{x_{ori} - x_{min}}{x_{max} - x_{min} + 1}$$

5. DATA EXPLORATION

After ensuring that the data is well preprocessed, data exploration is carried out including EDA (exploratory data analysis) and hypotheses testing. Hypotheses and intuition about possible patterns could be inferred in this process. Depending on the data, different EDA techniques can be applied, and a large amount of information could be extracted.

5.1. EDA

This section aims at providing exploratory data analysis to well preprocessed data.

First, seasonality of stocks data is explored. We first explore seasonal cycles using a 30-day rolling average. As it could be seen in Fig. 11, there is a spike at the start of April. Also, the close value of stocks during April and September is generally higher than the other month in a year.

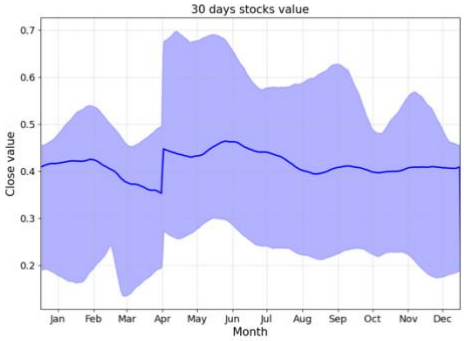

Fig. 11 Seasonality trend of close value of stocks data

Then, explore dependencies among year, month, week and day to find if there is some noise along with fixed or predictable events. The dependency on year and month and that on week and month in terms of stock close value is plotted via carpet plot/heatmap in Fig. 12 and Fig. 13, repsectively. It is suggested from these two heatmap that the stock value during 2017 to 2019 is generally higher than that in other years. Besides, there is no clear dependency on month and week in terms of stock value. The trend of stock close value is also explored on the weekday and weekend base. As it is shown in Fig. 14, the close value curve for weekday interweaves closely with the curve for weekend.

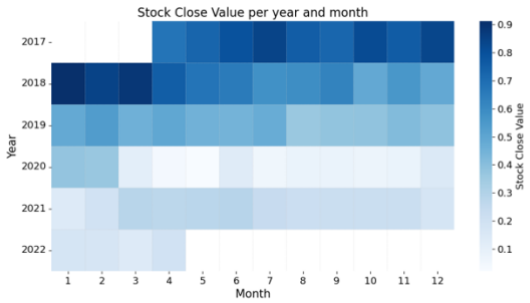

Fig. 12 Dependencies between year and month for stock close value

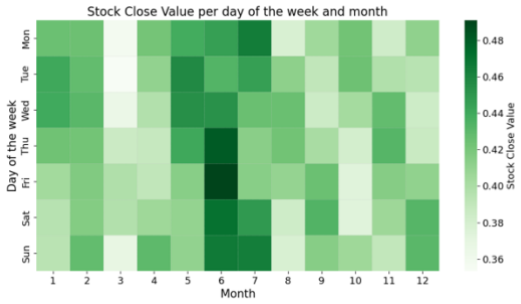

Fig. 13 Dependencies between day and month for stock close value

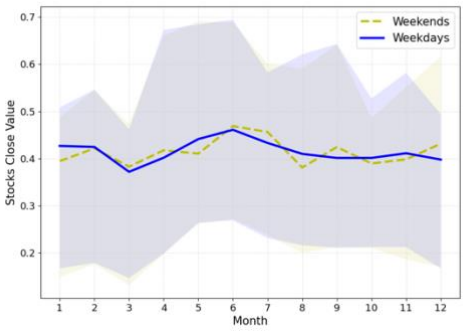

Fig. 14 Dependencies between weekends and weekdays for stock close value

After this, we explore the correlation between features of stocks data and the correlation between stocks data and auxiliary data including weather and covid data.

Features correlation provides additional insight into the data structure as it is indicated in Fig. 15. The relationship among Close, Open, High, Low feature is almost linear. Also, the volume of stock become

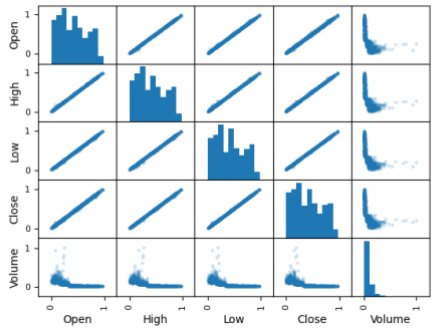

Fig. 15 Correlations between feature pairs of stock data

The correlation between "close" feature in stock dataset and features in weather dataset is shown in Fig. 16, and that between "close" feature and features in covid dataset is shown in Fig. 17. There is not very clear relationship between weather features and close value, while the close value of stock could be low when the number of covid case and death is large. It could be speculated that the stocks could suffer a decrease when the severity of pandemic is high.

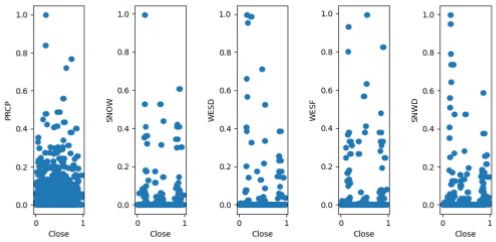

Fig. 16 Correlation between stock close value and weather data

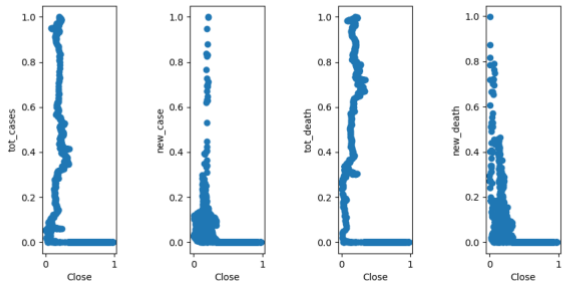

Fig. 17 Correlation between stock close value and covid data

In the next section, hypothesis testing is conducted to better understand the composition of the dataset and its representativeness.

5.2. Hypothesis testing

Hypothesis testing is a formal procedure for investigating specific predictions, called hypotheses, by calculating how likely it is that a pattern or relationship between variables could have arisen by chance.

For this task in particular, we first define the hypothesis as sampling two groups of data of which one is 20 sampled close value when there is no PRCP, the other is 20 sampled close value when there is. The hypotheses testing is to judge whether stock close value is related to weather condition like precipitation.

There are a variety of statistical tests available, but they are all based on the comparison of within-group variance (how spread out the data is within a category) versus between-group variance (how different the categories are from one another). If the between-group variance is large enough that there is little or no overlap between groups, then your statistical test will reflect that by showing a low p-value. This means it is unlikely that the differences between these groups came about by chance. Alternatively, if there is high within-group variance and low between-group variance, then your statistical test will reflect that with a high p-value. This means it is likely that any difference you measure between groups is due to chance.

The p value generates 0.218611 and one tailed p value:0.109306. The result of hypotheses testing is “Fail to reject null hypothesis”.

6. DATA INFERENCE

The inference section aims at training models to predict the closing stock price on each day for the data acquired, stored, preprocessed and explored from previous steps. The data spans from April 2017 to April 2022. One single LSTM-based model architecture is used for developing two separate models of which one is a model for predicting the closing stock price on each day for a 1-month time window (until end of May 2022) using only time series of stock prices and the other is a model for predicting the closing stock price on each day for a 1-month time window (until end of May 2022) using the time series of stock prices and the auxiliary data you collected. Furthermore, it evaluates the performance of the model using mean absolute error and create visualizations to provide useful insight of the prediction result.

6.1: Development of model using stocks

The model for this subsection is one for predicting the closing stock price on each day for a 1-month time window (until end of May 2022), using only time series of stock prices.

LSTM based model is used and only stocks data were used to train this model. The original dataset is split into training, validation and testing dataset with a size ratio of 7:2:1.

The number of training epochs is set to 20 and early stopping mechanism is applied to shorten the training time and prevent overfitting. After training, testing result is shown in Fig. 18.

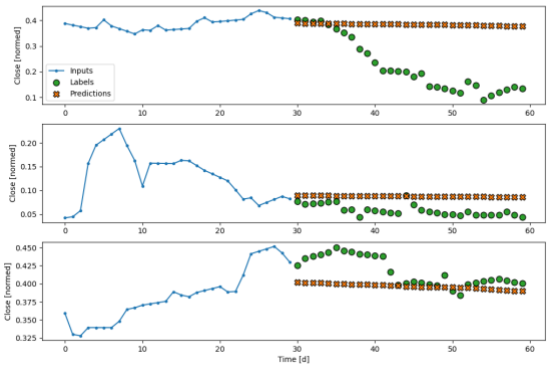

Fig. 18 Performance of model using stocks

6.2: Development of model using stocks and other data sources

In contrast to the dataset used in subsection 6.1, here we use both stocks data and auxiliary data including weather and covid data. The three dataset could be combined into one integrated dataset because they share the same dataframe index which is the time.

LSTM based model is used and integrated data were used to train this model. The original dataset is split into training, validation and testing dataset with a size ratio of 7:2:1.

The number of training epochs is set to 20 and early stopping mechanism is applied to shorten the training time and prevent overfitting. After training, testing result is shown in Fig. 19.

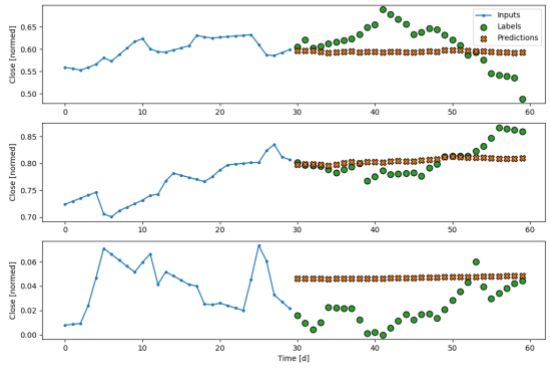

Fig. 19 Performance of model using stocks and auxiliary data

6.3: Evaluation metrics implementation

Evaluate the prediction result through metric like mean absolute error. In particular, the loss and mean absolute error between the labels and predictions is computed during training and validation process. The learning curves for the stocks model and integrated model using both stocks and auxiliary data are shown in Fig. 20 and Fig. 21, respectively.

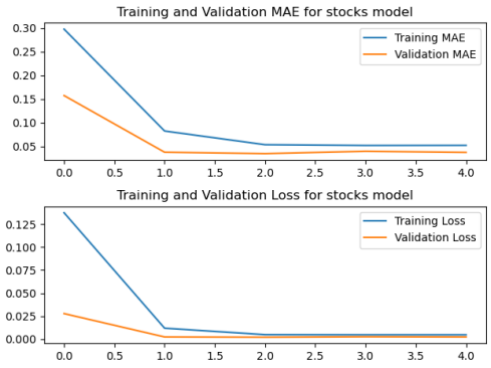

Fig. 20 Training and validation MAE for stocks model

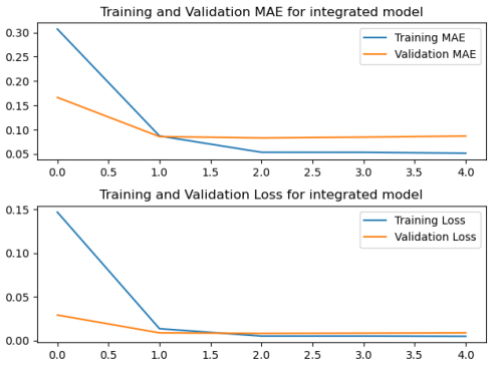

Fig. 21 Training and validation MAE for integrated model

Joint plot is created in Fig. 22 and Fig. 23 showing marginal distributions to understand the correlation between actual and predicted values.

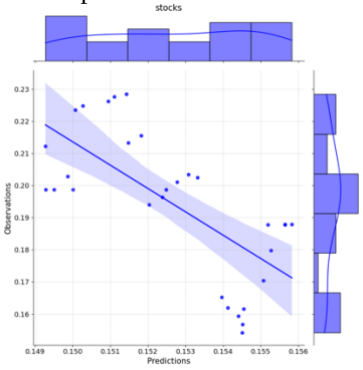

Fig. 22 Joint plot of predictions and true values for the model developed by stocks data

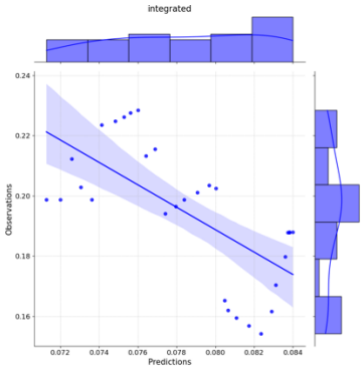

Fig. 23 Joint plot of predictions and true values for the model developed by stocks and auxiliary data

In addition, we create residual distribution plot for labels and predictions during April 1st and 30th 2022 and then find the mean, median and skewness of it. The plot is shown in Fig. 24 and Fig. 25.

It indicates that sometimes the prediction value is close to the true value but it is not at other time. The predictive capability of the model still requires improvement.

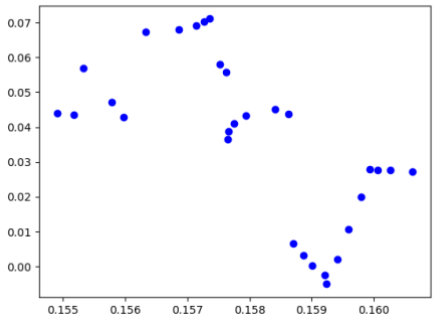

Fig. 24 Residual distribution plot for stocks model

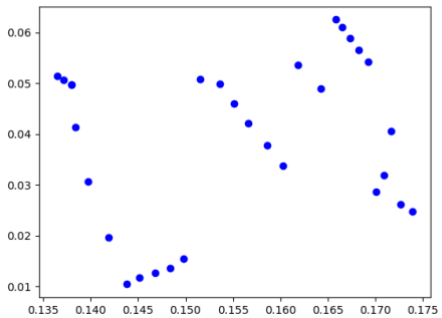

Fig. 25 Residual distribution plot for integrated model

Finally, the close stocks value in May 2022 is inferred by our developed models. The graphs in Fig. 26 and Fig. 27 demonstrate the trend of stocks value through April 2017 to May 2022

during which the value in May 2022 is predicted by proposed model.

Fig. 26 Stocks trend including predicted value by stocks model

Fig. 27 Stocks trend including predicted value by integrated model

7. CONCLUSION

This project simulates a real-life data-science situation that can be approached using the process including data acquisition, storing, preprocessing, exploration and inferring. The data is acquired by relevant application programming interface and python functions for sending HTTP request. Then the acquired data is stored both locally and in the cloud. The data could be retrieved from local disk and cloud database for the downstream preprocessing including data cleaning, visualization and transformation. Next, the exploratory data analysis is conducted to find potential pattern of the data and hypothesis testing is carried out to better understand the composition of your dataset and its representativeness. Finally, the preprocessed data is feed as input to train and test the LSTM model which could be later used to predict the trend of stock values in the future. The testing result of our model indicates that there might exists overfitting problems. Therefore, in the next step, model hyperparameters should be tuned or more advanced model could be used to fit our data in order to enhance the predictive performance.