

FINANCIAL DATA FORECASTER

Supervisor: Dr. Beta Yip

Shubhankar Agrawal (3035345306)

Karan Mahajan (3035346025)

INTERIM REPORT

02 FEBRUARY 2020

# **ABSTRACT**

Forecasting financial instruments has been a topic of interest in the world of research for a considerable number of years. The uncertainties in risk along with the potential for significant returns have lured people into financial data prediction. Over time, a wide range of methodologies and approaches have been implemented to analyse and forecast market movement. The ever-changing world of technology and finance constantly demands for newer strategies to obtain successful results for forecasting the market. This project aims to create a solution that will incorporate time series data of financial markets on a cross domain and international scale, along with news and public sentiment to obtain a better comprehension of the factors driving market movement and to predict them. Data extracted from sources online along with analysed sentiment of social media posts will be used to obtain forecasts for market indices in a future time period. A wide range of machine learning models will be employed, ranging from naïve regression to deep learning networks over an iterative process to improve the learning approach. A project website has been set up to describe the different aspects of the project. Financial data over a historical time period has been collected and stored in a database for further analysis and feature engineering. Immediate steps including raw data pre-processing and data analysis to gain an initial understanding of the nature of the data before it is used in machine learning models for prediction. Following a proposed schedule, this project aims to deliver a forecasting tool that can predict movement of cross-domain market indices using historical data with a high accuracy. This can additionally provide insight into the underlying factors influencing economies and help professionals in several domains.

# **ACKNOWLEDGEMENT**

We would like to extend my gratitude to my project supervisor Dr. C L Yip for giving us the opportunity to work on this topic as our final year project. His assistance in navigating the complex field of machine learning while dealing with financial instruments was invaluable in enhancing the quality of our research and guiding us throughout the workflow of the project.

**TABLE OF CONTENTS**

[ABSTRACT I](#_Toc31559436)

[ACKNOWLEDGEMENT II](#_Toc31559437)

[LIST OF FIGURES V](#_Toc31559438)

[LIST OF TABLES V](#_Toc31559439)

[LIST OF ABBREVIATIONS VI](#_Toc31559440)

[1. INTRODUCTION 1](#_Toc31559441)

[**1.1 Background** 1](#_Toc31559442)

[**1.2 Focus** 1](#_Toc31559443)

[**1.3 Deliverables** 2](#_Toc31559444)

[**1.4 Report Outline** 2](#_Toc31559445)

[2. LITERATURE REVIEW 2](#_Toc31559446)

[**2.1 Previous Research** 2](#_Toc31559447)

[**2.1.1 Naïve Methods** 2](#_Toc31559448)

[**2.1.2 Sophisticated Methods** 3](#_Toc31559449)

[3. METHODOLOGY 5](#_Toc31559450)

[**3.1 Software Development Practices** 5](#_Toc31559451)

[**3.2 Project Framework** 5](#_Toc31559452)

[**3.3 Data Collection** 6](#_Toc31559453)

[**3.3.1 Numeric Data** 6](#_Toc31559454)

[**3.3.2 Textual Data** 7](#_Toc31559455)

[**3.4 Data Cleaning and Pre-Processing** 7](#_Toc31559456)

[**3.4.1 Numeric Data** 7](#_Toc31559457)

[**3.4.2 Textual Data** 7](#_Toc31559458)

[**3.5 Exploratory Data Analysis and Feature Engineering** 8](#_Toc31559459)

[**3.5.1 Exploratory Data Analysis** 8](#_Toc31559460)

[**3.5.2 Feature Engineering** 9](#_Toc31559461)

[**3.6 Algorithm Implementation** 9](#_Toc31559462)

[**3.6.1 Train Test Split** 10](#_Toc31559463)

[**3.6.2 Parameter Selection** 10](#_Toc31559464)

[**3.6.3 Train and test model** 11](#_Toc31559465)

[**3.6.4 Evaluate Metrics** 11](#_Toc31559466)

[4. PROGRESS 12](#_Toc31559467)

[**4.1 Phase 1 Deliverables** 12](#_Toc31559468)

[**4.2 Data Gathering** 13](#_Toc31559469)

[**4.2.1 Numeric Data** 13](#_Toc31559470)

[**4.2.2 Textual Data** 13](#_Toc31559471)

[**4.3 Sentiment Analysis** 15](#_Toc31559472)

[**4.4 Data Processing and Feature Engineering** 16](#_Toc31559473)

[**4.5.1 Numeric Data Collection** 18](#_Toc31559474)

[**4.5.2 Non-Numeric Data Collection** 19](#_Toc31559475)

[**4.5.3 Data Processing** 19](#_Toc31559476)

[5. FUTURE PLANS 19](#_Toc31559477)

[**5.1 Project Schedule** 20](#_Toc31559478)

[6. CONCLUSION 21](#_Toc31559479)

[REFERENCES 22](#_Toc31559480)

# **LIST OF FIGURES**

[Figure 1: Term Frequency - Inverse Document Frequency Algorithm. Extracted from [14] 3](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559265)

[Figure 2: Cross-Domain aspect of Forecasting indicating correlation within a domain, between domains and across a time series. Extracted from [15] 4](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559266)

[Figure 3: Seasonal Decomposition of data indicating observed data, trend, seasonal and residual components. 8](#_Toc31559267)

[Figure 4: Iterative Machine Learning Process 9](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559268)

[Figure 5: Train test split and Cross validation 10](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559269)

[Figure 6: Evaluation Metrics - MAE, MSE, RMSE, R^2 11](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559270)

[Figure 7: Evaluation metrics - F1 Score 12](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559271)

[Figure 8: How Scrapy works, from sending a request, getting a response to parsing data and saving it in a database. Extracted from [11] 14](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559272)

[Figure 9: BERT Accuracy; increase in accuracy owing to the Masked LM feature of the model 15](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559273)

[Figure 10: Seasonal Decomposition of Hang Seng Index Returns Over 10 Years 16](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559274)

[Figure 11: Results for Augmented Dickey-Fuller Test Highlighting the Stationarity of Calculated Daily Returns 17](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559275)

[Figure 12: Correlation Matrix for Currencies Highlighting Correlations Discovered Over the 10 Year Period 18](https://d.docs.live.net/1b841d3d0b6ede16/FYP/3035345306_PR2.docx#_Toc31559276)

# **LIST OF TABLES**

Table 1: Project Schedule [20](#_Toc31557992)

# **LIST OF ABBREVIATIONS**

**APAC** Asia Pacific

**OHLCV** Open High Low Close Volume

**ARIMA** Auto-Regressive Integrated Moving Average

**MVP** Minimum Viable Product

**BERT** Bidirectional Encoder Representations from Transformers

**API** Application Program Interface

**NoSQL** No Structured Query Language

**NLP** Natural Language Processing

**TF-IDF** Term Frequency – Inverse Document Frequency

**TDD** Test Driven Development

**MAE** Mean Absolute Error

**MSE** Mean Squared Error

**RMSE** Root Mean Squared Error

**JSON** JavaScript Object Notation

**CSV** Comma Separated Values

# **1. INTRODUCTION**

The following chapter entails the overview for the project. To begin with, the background and motivation of the project are presented, followed by a section about the primary focus of the project. Further described in this outline are the key deliverables for this project along with an outline for the remaining content of this report.

## **1.1 Background**

The analysis of financial data is an important tool in the business world. The use of computational power as a catalyst in such analyses has created a new space for technology in finance. As emerging technologies are constantly bringing in disruptions to the financial industry, there is a constant need for the development of new and innovative ways of exploiting technology in the domain of finance. Machine learning, a powerful tool is being increasingly deployed to understand financial market movements and to forecast future performances of financial instruments, according to Zhang [1].

In addition, the global economy and the access to information has possibly led to an increased relevance in the interdependence of international economies. The open nature of world markets can prompt discussions as to how and to what extent they affect each other. Such economic possibilities lead to interesting studies in the field of machine learning in order to investigate if such cross-market relationships can be determined using algorithms and if they can be exploited to help forecast future financial data.

Market movement is also affected by other non-financial factors. Ludhiyani et Al say that some of these may include political agendas, environmental aspects and/or public sentiment about these issues [2]. The ability of human psychology to drive investor decisions can be a crucial factor in certain cases where a technological analysis of historic data may not suffice. Capturing these records may also provide valuable insight into some of the non-numeric data that causes market indices to rise or fall.

## **1.2 Focus**

The primary focus of this project is to obtain successful forecasts for some of the prominent market indices and use analysis to discover the important factors which influence these markets. To narrow down the focus, the Asia Pacific (APAC) sector has been chosen along with the US markets. Data for Open High Low Close Volume (OHLCV) will be collected for the market indices in these countries over a historical time period daily dating to the existence of these markets. In addition, the forex rates for these countries will also be collected and investigated. The final element will be social media sentiment from Twitter and Reddit using keywords pertaining to the countries such as their capital cities and their presidents. These factors together will be used to predict the market movement by splitting the time series obtained into training and validation data to backtest our algorithms.

## **1.3 Deliverables**

Encompassing the deliverables of the project are a project website, a script to obtain forecasts for the next time period and research documentation. Creation of a script to generate forecasts will additionally require scrapers that can obtain both numeric and non-numeric data that are periodically stored in a maintained database. The submission of these above-mentioned deliverables will result in the completion of work on our final year project.

## **1.4 Report Outline**

The report has been divided into 6 chapters. Chapter 1 has given a background to the topic chosen and a brief introduction. Chapter 2 provides a brief description of the literature reviewed by the team in order to narrow down to the topic. Chapter 3 explains the methodology of the approach that will be followed to complete this project as per the requirement of the deliverables. Chapter 4 contains details on our current progress, challenges faced by us and the next immediate steps. Chapter 5 discusses future plans while Chapter 6 acts as a conclusion summarizing the information discussed in this report.

# **2. LITERATURE REVIEW**

This section entails a brief summary of the literature review conducted by the team for this project. Past research has provided us with information on the different machine learning approaches that have been employed in the financial industry. This acted as a guiding tool to formulate our methodology.

## **2.1 Previous Research**

There have been several studies with the aims of predicting future prices of financial instruments such as derivates, securities and cash instruments. Through a rigorous process of literature review, our team moved from research discussing simplistic methods to ones using more complicated statistical models for analysis.

### **2.1.1 Naïve Methods**

Many of these studies have primarily focused with one particular entity to attempt a forecast for future periods [3]. For example, the studies performed by Boldt, and Chong and Pu focus on only the sales of Nike and the paying rate index respectively [4][5]. By limiting research to one individual instrument, these research attempts may lack several potentially impactful insights in the data. Historical data taken over a long period of time is sometimes a sufficiently good indicator of the future values in a time series owing to the seasonal and repetitive nature of a financial instrument over a quarter, a year or other financial periods. However, quite often external events often disrupt this periodic nature of the industry thereby creating unexpected movements in the prices of financial instruments. Thus, it is useful to take into consideration external variables to capture such external influences which lead to unprecedented volatilities in data.

Some alternative studies have focused on a range of products albeit having only tested them on a few specific machine learning methods [6][7]. There are two different kinds of studies found, of this type. The first consisted of research that used learning models which could be implemented across a range of financial products. This can be seen in the paper by Wang and Nie where the methodology implemented by them can be reproduced across several different market indices thus enabling greater flexibility in the learning model [8]. This added configurability also permits the audience to review and compare performance across different markets and evaluate the research based on the results obtained. The second and other type of study uses a range of products in a different way. In this case, several different financial products are used as predictive variables (features) to forecast the value of one entity. Research by McCluskey and Liu uses one such approach to forecast the NASDAQ index based on financial data from 12 other technical indicators which comprise of global stock market indices [9]. The usage of external variables can potentially improve the learning process but there is often data in other formats that can also act as an important feature in prediction.

### **2.1.2 Sophisticated Methods**

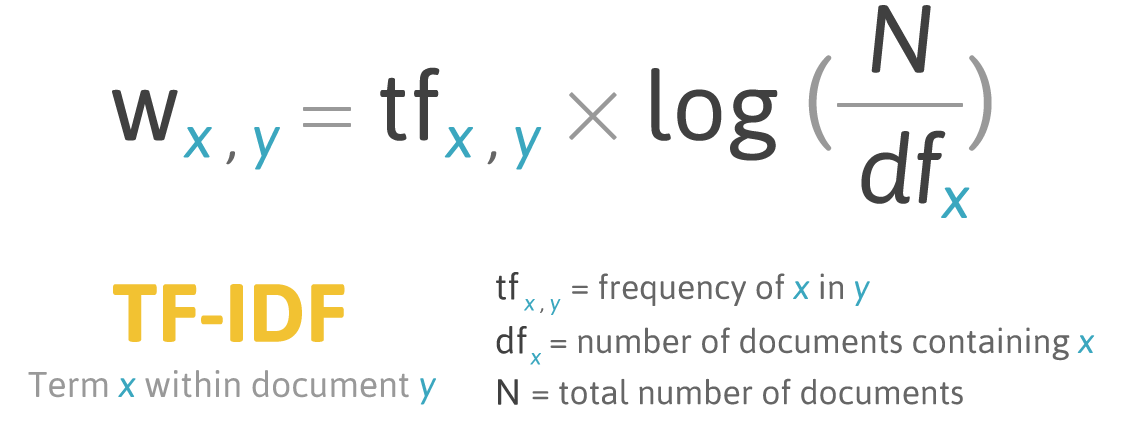
Research has also been conducted on using sentiment obtained from social media websites, but they are confined to the domain of forecasting the direction of the market movement (although with a relatively higher accuracy compared to the methods formerly mentioned) and not the exact values themselves [10][11][12]. These pieces of literature describe using methods of Natural Language Processing (NLP) to extract keywords from the data. As seen in research by Yıldırım et. al. one common method used in the text-processing approaches is using the Bag-of-ngrams approach, which helps in text representation to count the frequency of words in a document [13].

Figure 1: Term Frequency - Inverse Document Frequency Algorithm. Extracted from [14]

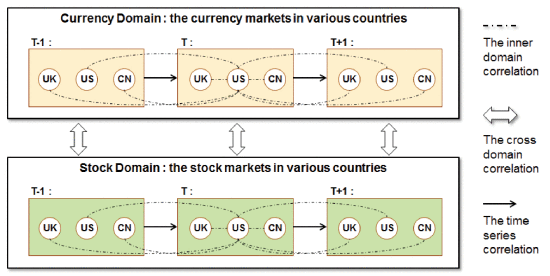
A more popular approach considered is that of Term Frequency – Inverse Document Frequency (TF-IDF) which is used to understand the importance of words in a document by calculating their weights with respect to the remainder of the document. As seen in Figure 1 above, the TF-IDF algorithm calculates the weight of each word within a document to get a measure of the overall importance of phrases, sentences and/or paragraphs. This helps convert textual data of various forms into numeric probabilities which are then used in machine learning algorithms. Similar to the approach mentioned before with the range of financial products, there has been some research in the cross-domain aspect which utilized statistical analysis models such as the Auto-Regressive Integrated Moving Average (ARIMA) to perform an analysis for correlation between time series data using automatic regression and moving average. In addition, more sophisticated methods such as the neural networks were also implemented to compare performance with statistical models to check for improvement in performance thereby producing better results. One interesting approach was carried out by Pan et. Al. where a cross-domain study was carried out considering a few major worldwide markets along with their currency pairs over a time series, as illustrated in Figure 2 [15].

Figure 2: Cross-Domain aspect of Forecasting indicating correlation within a domain, between domains and across a time series. Extracted from [15]

This approach involved using the correlations between countries along with the correlations amongst different markets within the same country as important variables of prediction. In addition, the time series aspect ensured historical values were not ignored and used as a predictor of future values as well. This served as our primary inspiration for carrying out the project. By combining certain aspects of these previous works and adding in some new ideas, our solution will leverage existing research and build on it to forecast financial data.

# **3. METHODOLOGY**

A plan of action has been set up to ensure continuous progress for the project through the academic year. The project has been split up into phases for easier development and to pursue a systematic approach of work.

## **3.1 Software Development Practices**

The adoption of a software development lifecycle enables us to pursue an organized method of work. The agile system of development, an iterative and incremental model is chosen to allow extended flexibility while working on the project. As per the agile development cycle, the project is split into its requirements as Minimum Viable Products (MVP), each of which are developed iteratively within their own respective phases of work. Bi-weekly sprints act as these individual phases of development where deadlines to complete every 15 days are set within the team. A complete schedule of the proposed plan of work can be found in the Future Plans section along with the status on the individual parts.

The team aims to follow several agile programming practices while moving through the phases of the implementation. A system of simple design is ensured while programming for the project by avoiding complicated programming style. For example, modularity is ensured by separating tasks into different functions. At the same time, redundancy is avoided by not creating impractical functions for menial tasks. The standard of coding is maintained across the members of the team in terms of readability and functionality. For example, the structure of naming entities, indenting code and file formats are kept constant throughout the project. A system of regular refactoring is done to avoid any errors in variable and function naming across the programming environment. The codebase is shared through the team using an effective version control system that ensures there are no leakages in code and that commits can be traced back in a chronological fashion. The process of continuous integration is followed to ensure changes made are added to the pipeline as when they are completed. Test Driven Development (TDD) helps to keep code in check by writing tests to suit our needs which initially fail and then following it up by programs that work to pass those tests. In addition, a pair programming approach is used often in the team to help debug important parts of the code and to discuss efficient ways of writing parts of a program.

## **3.2 Project Framework**

The project work is carried out primarily using Python as the development language. Python is chosen by the team for multiple reasons. The presence of large-scale standard libraries enables to use the language in a variety of applications. In addition, the simplicity of coding in Python along with the ability to integrate lambda functions and other aspects of functional programming in the codebase. The extensive usage of Python also enables the use of open source libraries (such as those written by Google and Facebook), some of which are used regularly in this project. Libraries such as **pandas** and **scikit-learn** facilitate working with big data by providing tools to use configurable data structures (like data frames) and provide functionality for machine learning algorithms usable with a wide range of parameters.

In addition, we also aim to utilize several frameworks available on Python for the different requirements of our project. The **Scrapy** framework on Python is used to construct crawlers for websites that enable quick and efficient access to web data in the required format. The alternative to Scrapy, being Selenium, is not considered for this project owing to the ineffective nature of Selenium where it requires the opening of the webpage on a browser window in order to scrape data from it. The use the **Django** framework in Python, permitted us to also deployed a web server to **Heroku** to handle data queries. The **Anaconda** platform provides the functionalities of **Jupyter**, a service useful for experimentation while working on the data with the machine learning models.

For the purpose of the development of the Final Year Project website, HTML, CSS and JavaScript are used to create a static webpage to demonstrate the progress of the project.

**GitHub** is used as the primary version control system to ensure code is maintained in a secure and organised allowing the team members to trace back work at any point of time. The services of **Trello** are also used to create a virtual scrum board for the team to keep a track of progress through the different iterations of the project.

## **3.3 Data Collection**

The initial steps required for our project of financial data analysis required data to be collected from the various sources. The process of data collection first required the identification of the different types of data required for the project. The data needed was divided into numeric financial data and non-numeric textual data.

### **3.3.1 Numeric Data**

The numeric financial data comprised of 2 different types of data. The former is the values of the market indices for the prominent stock markets in the APAC zone. For this purpose, 18 different countries were selected, and the primary indices were picked from these limiting to 2 indices from each country. In addition, 3 indices were selected from the United States to provide a common ground while cross analysing these indices. The values of the indices were obtained daily containing the OHLCV data for each day the individual markets were open and operating in their respective countries.

The latter part of the numeric data incorporated the currency exchange (i.e. Forex) rates amongst these countries. Forex rates weren’t calculated and obtained for each of the pairs permutable among these 18 countries. Instead, the US Dollar (USD) was kept as a baseline and the Forex rates were obtained for each country’s currency in the form XYZ/USD (where XYZ represented the currency). The currency data is also obtained daily in the OHLC format.

### **3.3.2 Textual Data**

The non-numeric data to be obtained consisted of social media posts from some popular social media such as Reddit and Twitter. Keywords relating to the countries selected, their primary capital cities and their heads of state were used to identify relevant tweets for further analysis. A similar methodology is adopted with Reddit to gather posts from certain subreddits. These subreddits were identified by picking the countries and their capital cities.

## **3.4 Data Cleaning and Pre-Processing**

Similar to the data collection steps, different pre-processing methods must be employed to the numeric and textual data owing to their individual natures. The numeric data can be directly cleaned as it exists on a day to day basis whereas the text data must be processed first before other computations can be performed on it.

### **3.4.1 Numeric Data**

The numeric data is cleaned by getting rid of missing values and ensuring that time periods are consistent between the different countries selected, taking into consideration the different time zones. The records containing missing values will be removed if they are in a minority. A better workaround is to replace the missing values with an average or median of the prices for the days immediately preceding and following it to remove any potential discrepancies. Owing to the different countries for each market index picked, the dates for the market being open are to be taken into consideration while cleaning. The common dates across which the different markets are open are taken as they are keeping the US market as the standard, whereas the unique dates for certain countries must be dealt with differently. One potential solution for this is to aggregate the market movement over the extra open days and reflect it in the first common date of both the markets.

### **3.4.2 Textual Data**

In order to utilize the obtained text data in machine learning models, they will have to be transformed to some numeric format to be used for the purpose of optimization. One method we aim to employ for this is the use of sentiment analysis on the stored text. Sentiment analysis analyses a piece of text and provides a mathematical value on a scale for the general sentiment of the phrases. Each post from social media is evaluated individually to get a score that is then to be aggregated on a daily basis. If the post is in a different language, the evaluation is done in the language of the post using a multi-lingual encoding program. This aggregation of sentiment will be done by calculating a geometric mean rather than an arithmetic mean to ensure that the compounded effect of sentiment is not ignored. In addition to the post itself, the social media data also consists of values such as number of replies and favourites which will be aggregated in a similar manner.

## **3.5 Exploratory Data Analysis and Feature Engineering**

The Exploratory Data Analysis (EDA) and the Feature Engineering step of the process are extremely crucial to the project. This phase holds greater significance to the project than the implementation of the machine learning algorithms. It is only when the data has been understood and transformed to the relevant form can the machine learning algorithms make the substantial predictions for the future time periods with greater accuracy.

### **3.5.1 Exploratory Data Analysis**

Exploratory Data Analysis is the part of the project pipeline which investigates the data to identify existing patterns or trends that can ease the process of machine learning. The exploration phase of the project can help us obtain initial analyses of the raw data to draw our own relevant conclusions as to what some of the potential parameters can be in the learning models. Common techniques of exploration include charting out the data using graphs such as histograms, scatterplots and candlesticks.

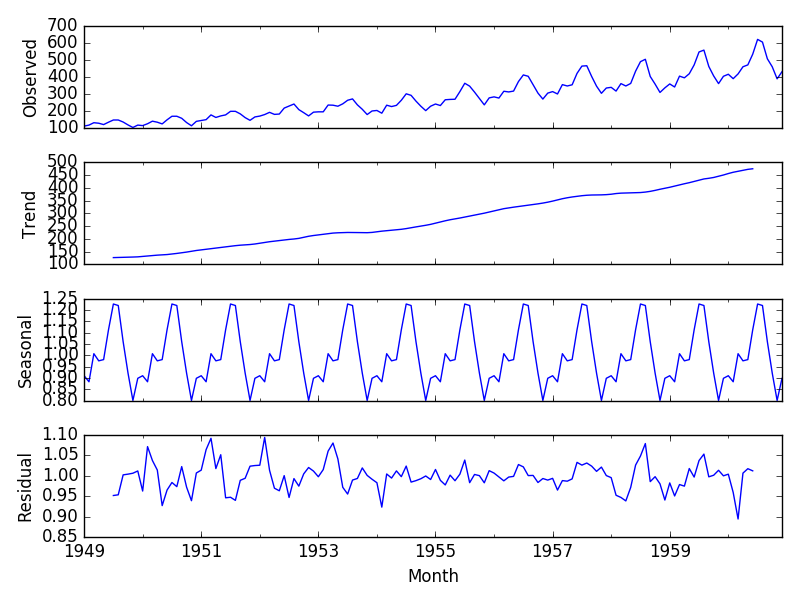


Figure 3: Seasonal Decomposition of data indicating observed data, trend, seasonal and residual components.

As seen in Figure 3 above, applying a method like seasonal decomposition can identify trend and seasonality and allow us to compare it with the actual data to obtain relevant insights. A yearly seasonality can be clearly noticed in the data. This can be exploited in our machine learning models where setting a seasonal component of 12 months can achieve better results than naïve computations.

### **3.5.2 Feature Engineering**

Based on the results obtained from the data analysis, we can perform feature engineering to obtain the relevant features and manipulate them. Common methods involve resampling the data to obtain a non-biased sample. However, dealing with several time series, resampling of data points in our dataset cannot be performed for this purpose without systematic interpolation. The data once plotted can give an indication of outliers in the data. Such values of acute skewness can be removed by transforming the data by performing mathematical functions. Some of the common ways to address this are to use normalization, standardization and even simple mathematical functions such as a logarithm to remove negative values and scale all values to a smaller and easier computable range. Feature engineering will increase the relevancy of the data to be incorporated in the machine learning algorithms to achieve better results.

In addition to the above-mentioned ways of transforming the data, there are also steps in feature engineering based on the machine learning models chosen. For naïve machine learning models which do not implicitly evaluate the time series nature of the data, it is important to provide it with a few values from the past to allow previous records to be used as features too. This will be done by either directly including the raw values of time period **(t-1)** in row **(t)** or by calculating the rates of return over the periods thereby further simplifying the data. For such machine learning methods, inbuilt functions will also help us generate moving averages over a weekly/monthly time period to be used as features.

## **3.6 Algorithm Implementation**

Our methodology incorporates the use different algorithms iteratively over our data through the phases of implementation. This serves the purpose of comparing the performance over different algorithms as well as identify the algorithm with the best results.

Figure 4: Iterative Machine Learning Process

As mentioned in Figure 4, the iterative machine learning process starts with the train test split of the data. Next, we pick the best parameters for the models that are experimented with. Following this, the model is trained on the train data and then test it using the test data. The model’s performance is evaluated using the metrics selected.

### **3.6.1 Train Test Split**

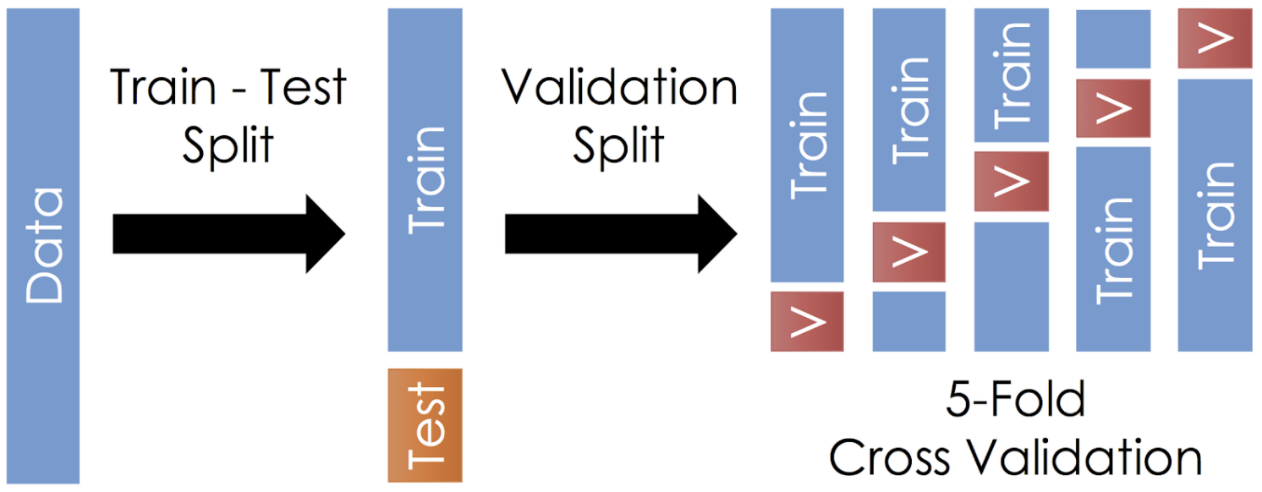
Once the data has been collected and cleaned, the next step is to split it into a training and testing data to pass into the algorithms. For the naïve methods, we can randomly sample the data as the past values have already been taken into consideration for each individual record. However, for methods that take the time series factor into consideration (implicitly such as ARIMA) the train test split if performed by considering the older values as part of the training set and the newer values as part of the testing set. Potential values for the train test split

Figure 5: Train test split and Cross validation

range from anywhere between 50% to 80% which will be maintained throughout the models. To improve the performance of the training model, cross-validation is deployed on the training set as shown in Figure 5 below. By splitting the training data into several validation sets, the model learns during the process of training by penalising itself for its errors and improving its parameters.

### **3.6.2 Parameter Selection**

Once the data has been split into the necessary train, test and validation sets, the best parameters for each of the individual models are selected. By using the cross-validation approach as mentioned before, the most optimal parameters for each of the models are picked by simultaneously building models with the different parameters and picking the ones with the ideal learning curve. For example, in the Random Forest approach, the number of trees per cluster is decided after running the model training over the cross validated subsets of data and choosing the parameter with the most accurate fitting. For more sophisticated models like neural networks, the parameters such as the number of nodes per layer and activation function will be manually varied and tested to arrive at the conclusion for the best set of parameters.

### **3.6.3 Train and test model**

The machine learning algorithms will take the features as inputs and aim to predict the target variables as outputs. The features used in the algorithms will be the time series data of the indices of the other markets, the currency exchange rates between their countries, the social media and news posts. Concerning the target variables, the machine learning models will not only try and forecast raw values of the indices but also consider the accuracy of prediction in terms of only forecasting the direction and attempting to calculate the bins of the price range into which the values fall into. The modelling of algorithms will start with classical regression and classification approaches, along with ensemble methods. The supervised learning methods the team aims to implement here include Ordinary Least Squares, Ridge, Lasso, ElasticNet, KNN, Naïve Bayes, Support Vector Machine (SVM) and RandomForest and AdaBoost. The next phase consists of using time series modelling using statistical models. Models such as ARIMA, SARIMAX, and Facebook Prophet are examples of some of the models as part of this. Finally, we will implement neural networks to aim to forecast market behaviour as has been performed before [16].

### **3.6.4 Evaluate Metrics**

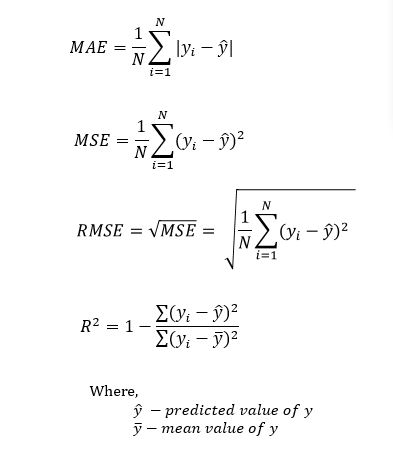
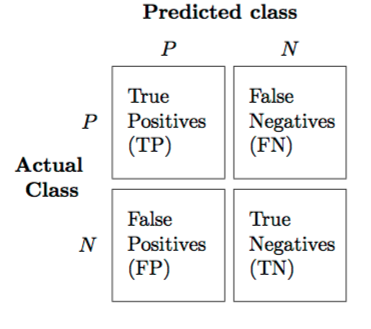
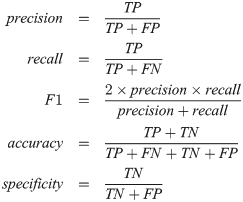
The results across models will be consolidated in an organised way to help clearly identify the best model that is able to learn from the data provided. Owing to the different nature of the target variables (raw values-continuous, bins-discrete), there are different metrics to be considered for each of those. Metrics such as the Mean Absolute Error (MAE), Mean Squared Error (MSE), the Root Mean Squared Error (RMSE) can be maintained across the different types to judge the performance of the models. The regression methods will also have an additional metric of the R-Squared Score (R^2) which is an indicator of the extent to which the feature values are correlated with the target values. The formulae for these metrics can be seen in Figure 6.

Figure 6: Evaluation Metrics - MAE, MSE, RMSE, R^2

The classification methods will have the additional metric of the F1 Score (as in Figure 7) which is a good measure of the accuracy. The F1 score is calculated by first creating a confusion matrix that represents the various values predicted for each category across the actual values. This also calculates the Precision and Recall scores which indicate the ratio of true and false positives with the data. The metrics mentioned are to be compared only across similar target variables to ensure there is no bias in judgement.

Figure 7: Evaluation metrics - F1 Score



# **4. PROGRESS**

Continuous work is being carried on to meet the deadlines of our proposed schedule. Having crossed the milestone of the first deliverable, the initial arrangement has been confirmed and work is being carried on based on those decisions.

## **4.1 Phase 1 Deliverables**

Due at the end of September, the phase 1 deliverables of the project primarily aided us in deciding the areas of work for the project and how to work about them. The two submissions for phase 1 included a project plan for the project and an initial website for the same. The project plan was a detailed introduction to our ideas for the project and how it was intended to be completed. This included the literature review, the proposed methodology to be carried out and the schedule for the project. The latter of the deliverables, the project website was created using an existing template found on the Creative Commons. This website provided a comprehensive outline to our Final Year Project containing an overview of the project, links to all the documentation, the schedule and proposed approach for the solution. This static webpage created is deployed on the HKU CS server for public access at the URL <https://i.cs.hku.hk/fyp/2019/fyp19020/>

## **4.2 Data Gathering**

The requirement for collecting data of two different natures needed two different processes for their procurement.

### **4.2.1 Numeric Data**

The two types of numeric data are obtained by using Application Program Interfaces (APIs) provided by open source projects online which contain historical data from websites that offer free real time quotes of financial data. The **InvestPy** package was identified to procure the numeric finance data. This data collection was achieved by installing the relevant packages in Python and utilizing these APIs to scrape data between the required time periods. The flexible configurability of the APIs allowed us to select between different time periods from the inception of the preselected markets until the current date to obtain the records. By storing the countries, their currencies and indices in a text file which can be easily parsed over the Python, the Pandas package was used to iterate over the pairs of currencies stored in the text file and save relevant data frames with the information that was required. These data frames were stored in a locally managed spreadsheet for quick and easy access. In addition, owing to the small size of the data, the need to maintain a server to store it was made redundant as it would lead to overhead costs.

### **4.2.2 Textual Data**

A different process had to be employed for the non-numeric data required for this project. For this part of the information retrieval, APIs could not be used as they limited access to historic data and hence a web crawler was constructed to collect the desired data. Tools such as Scrapy and Beautiful Soup enabled the construction a social media crawler for Twitter which will provide tweets between any two historical dates. The working of Scrapy can be seen in Figure 8 below which indicates the process our scraping mechanism goes through to obtain the tweets. A request is sent to the web server from which information is to be extracted. Once a positive response is received, Scrapy extracts the relevant parts of it and saves it in the database of choice.

The query can further be narrowed down to searching for a keyword or limiting it to tweets of a user. Posts from users were scraped from these specific subreddits providing public information about the happenings in these countries. In addition, news articles were obtained using APIs online as well.

All the textual data collected is stored in JavaScript Object Notation (JSON) files. As there is a large amount of data from social media. This will also provide decentralized access to the data allowing to use external computing power to perform analysis on the data.

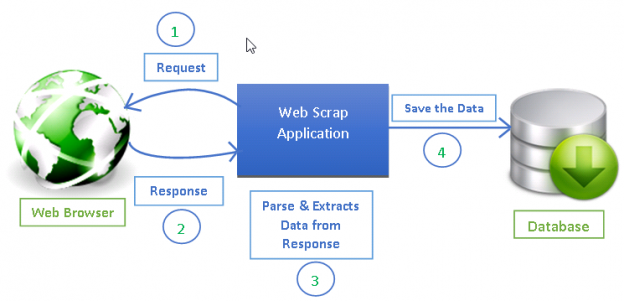


Figure 8: How Scrapy works, from sending a request, getting a response to parsing data and saving it in a database. Extracted from [11]

The Python scripts involving the use of the APIs and the scrapers are set up to run on a periodic basis. This will ensure that data is constantly collected and stored regularly. Moreover, this will quicken the process of financial data forecasting when needed as new data will only have to be scraped from the last checkpoint of the scraping process.

To periodically run these scrapers and for storage online, a server with virtual machine access was obtained from the HKU Computer Science department. This enabled us to set up scripts that would run on a frequent basis to query data, transform it, update relevant queries and move them to a storage directory.

Several scripts were written to accomplish this goal. A file containing the queries required is maintained. The first of our scripts starts a background process using one of the incomplete queries from the file. Another script helps verify the completion of a query and if so, updates the file with the status of the query. A third script is used to check if there are missing dates in the data collected and helps create duplicate queries for the missing dates. Following this, a script converts the data obtained to the required JSON file removing the unnecessary fields, and creates a Comma Separated Values (CSV) file copy with only the post for use in sentiment analysis. The results of these scripts are stored in the appropriate location along with log files containing notifications of whether the script crashed due to an error or caused any issues.

Automating this part of the process enables a more seamless transition to the server-based prediction model we aim to deliver. In addition, this would also eliminate the redundancy of the team having to manually run scripts to collect the data for the project.

## **4.3 Sentiment Analysis**

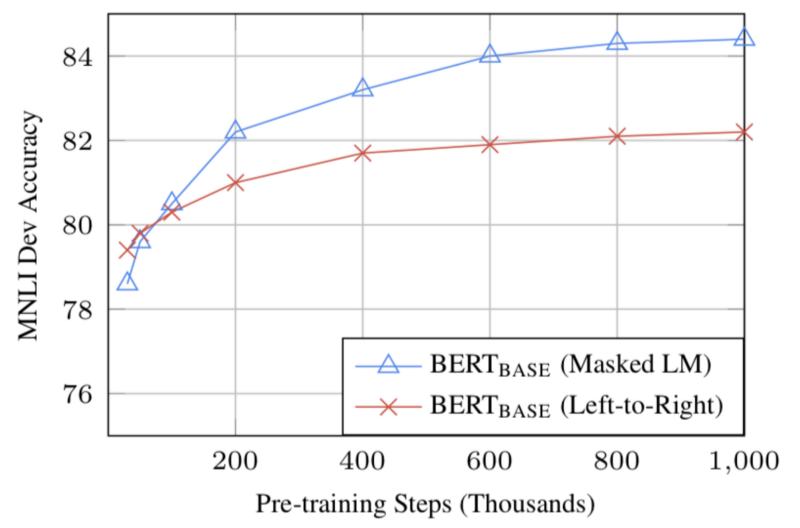
The sentiment model used in this project is built upon a publicly available model titled BERT which stands for a Bidirectional Encoder Representations from Transformers.

Figure 9: BERT Accuracy; increase in accuracy owing to the Masked LM feature of the model

BERT works by using a mechanism to initially encode the words in a sentence. Contrary to other methods which parse a sentence in one specific direction (either left to right or right to left), BERT reconstructs the sentence from both directions thus building contextual relations for each of the words present. As shown in Figure 9 above, by proceeding in a bidirectional format the BERT model is able to perform better than models which read through a sentence in only one direction. The data used in the statistical comparison is of the MNLI data set, which is a corpus of nearly 400,000 sentences spread across several genres.

Next, two strategies are employed in the learning process of our BERT model. The first is one called Masked Learning Method (Masked LM). As part of this, certain words from the sentence are masked out and the model aims to predict the scores of these words while analysing the remaining words. The second is called Next Sentence Prediction (NSP) where using one sentence, the next sentence is predicted. Utilizing these two to minimize the losses caused due to errors, BERT can arrive at the general sentiment for a sentence.

The BERT model is initially trained using a Multilingual Cased model provided by Google which encompasses the training corpus for the project. The results of BERT give a matrix of analysed sentiment. This is further fine-tuned by adding a layer of classification to get results on a binary scale. The fine- tuned version of BERT provides us with two probabilities indicating whether the post is positive or negative. These probabilities are converted to absolute values on a 0/1 scale to aid in our machine learning approach.

## **4.4 Data Processing and Feature Engineering**

Owing to being from several different countries, there were inconsistencies on when the time series period started for the different markets. In addition, the dates on which missing values were observed to be different across the countries due to dissimilar sets of public holidays across Asia pacific. Therefore, it was important to standardize the available data. For this, the data was then interpolated using the time series component and only the weekdays were kept for each market from their date of inception.

A screenshot of a cell phone

Description automatically generatedOn plotting out graphs of the raw values and examining them, our team noticed that it would not be feasible to use the raw values as the predictor or feature variables themselves. This was because the market indices were on different scales across countries and could not be treated similarly in any manner. The data was to be made stationary to ensure comparable results across different machine learning approaches as well as varying kernel sizes. We thus resorted to calculating their daily returns and using those as the value we aimed to predict.

Figure 10: Seasonal Decomposition of Hang Seng Index Returns Over 10 Years

In order to ensure our time series was stationary, the Augmented Dickey-Fuller test was performed across all our data. Inspection of individual markets showed us that the row values weren’t stationary while the returns calculated were consistently stationary across all markets. We thus decided on using the calculated daily returns as our target variables.

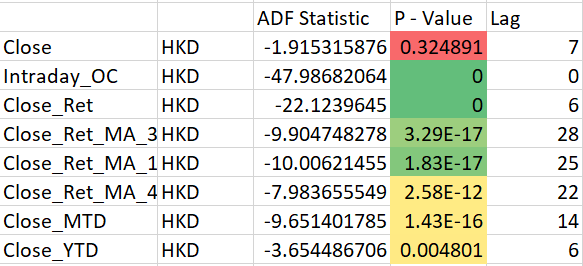
In order to generate our predictor variables, some additional features were generated. The intraday return between the Opening and Closing price, along with the Intraday return between the High and the Low were calculated. Moving averages over 3, 15 and 45 days were obtained as well. The lagged values of our time series along with these features generated will serve as our prepared data set for the machine learning.

Figure 11: Results for Augmented Dickey-Fuller Test Highlighting the Stationarity of Calculated Daily Returns

The results of sentiment analysis are aggregated on a daily basis using a variety of methods. Firstly, the sentiment score of each tweet/ reddit post is stored in a dictionary of arrays with dates as keys. We use these scores to get various statistical parameters for the data on a daily basis. From this data, we compute, the mean, max, min, standard deviation and variance along with number of posts per day. These aggregated sentiment scores are used as features in addition to our time series values after transformation. The scores being collected for all the days of the week were especially aggregated for the weekend. As per our hypothesis, the social media sentiment over the entire weekend would affect the scores for the following Monday. Thus, the scores from Saturday and Sunday were aggregated along with the scores from Friday.

On further investigating our prepared time series, we were able to identify clear correlations between certain pairs of currencies which are to be exploited as part of our machine learning approach where we will use these statistics to enable us to utilize cross market correlations as additional features. Moreover, seasonal decomposition was performed to identify is there was any underlying seasonality that could be utilized in our time series models. This allowed us to identify a yearly seasonality in the results, although a very marginal one, which our proposed time series models should be able to identify and use to their advantage.

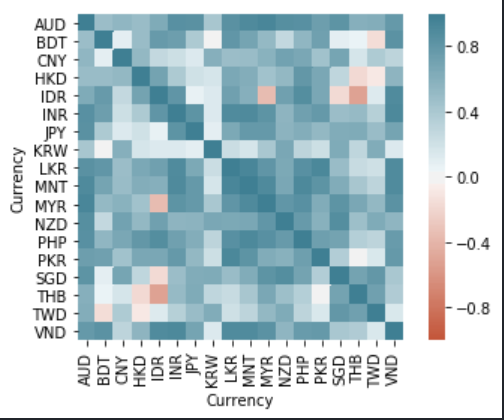
**4.5 Challenges Faced**

Figure 12: Correlation Matrix for Currencies Highlighting Correlations Discovered Over the 10 Year Period

The problem of successfully predicting financial data is tremendously difficult. It is plagued with assumptions and missing knowledge. It was the aim of the project to train models to understand the relationships between data that a human has failed to notice so far. Every step of the solution has posed problems. This section will highlight some main challenges faced by us and some solutions we developed to overcome them

### **4.5.1 Numeric Data Collection**

Another challenge presented in front of us was getting useful market data. Despite being the most insightful, tick by tick data for market data is the hardest to get access to but there are multiple sources available for market open-close rates and high-low rates. Investpy was identified as a useful program to retrieve relevant market data with ease. Since we cannot access tick by tick data and correlate it with market sentiment, we can generate an aggregate of market sentiment for the day and correlate it with the various values of market data available for the day. However, in order to tackle the loss of information due to unavailability of tick by tick data, we need to apply our algorithms over a large period to gather relevant insights and make accurate predictions.

### **4.5.2 Non-Numeric Data Collection**

Data collection comes with an inherent challenge of extracting relevant data. Twitter’s API has restrictions on number of tweets that can be retrieved and how old the tweets can be. In order to overcome this challenge, we developed our scrapper using scrapy. Designing a custom scrapper allowed us to scrape data as ancient as desired and we were not restricted by Twitter’s API call restrictions. This is attributed to the fact that a web scrapper simply reads data presented in the webpage and any information that can be retrieved using an API, can be retrieved using a simple search. However, the scraper struggled to deal with ten years’ worth of data at times. To deal with this issue, we developed scripts that would run as scheduled tasks every morning to check whether the data collected by scrapy matches the duration of our query. If it does, then we simply mark the query complete, and if it doesn’t, then we update the query and run scrapy again. This makes the process of data collection automatic and removes the need of manual checking for each query.

The files created from the crawlers are large (~1 GB) each and were unable to parse through normal Python scripts. To move past this issue, the team resorted to different languages for writing the script. First a Shell script was attempted as a lower level language would bypass memory restrictions. Although it showed better performance than Python, some of the issues were still prevalent. Some research about programming languages pointed us to Perl, which differs from other languages in the way that it uses a buffer to process files in small parts contrary to reading the whole file at once, as Python does. Recreating the script in Perl helped mitigate this challenge and run the scripts successfully.

### **4.5.3 Data Processing**

Despite having overcome the initial challenge of retrieving a large amount of data generally required for such a project, we are faced with a new challenge of cleaning the output provided by our web scrapper. Out of all the data the scraper collects for each tweet, only a small amount is relevant to data analysis and the rest can be discarded to save memory. We are currently writing scripts to parse our data into JSON objects with only the relevant data. This part is posing a big challenge as the amount of RAM available on the university’s servers is limiting our processing capabilities.

# **5. FUTURE PLANS**

Following the schedule mentioned below, our immediate sprints consist of deploying machine learning models on features that we have gathered and engineered. We will begin by analysing naïve machine learning models on a small data set first to establish the importance of various features and to narrow down our final target variable. Later, we will use the results of these naïve models as benchmarks for ensemble models that we will develop to potentially exploit the hidden relationships between features and domain.

Our team is in constant contact with our supervisor ensuring that we are on track in terms of progress through regular emails and meetings.

## **5.1 Project Schedule**

Following is the updated project schedule highlighting the remaining steps for the project.

Table 1: Project Schedule

|  |  |  |
| --- | --- | --- |
| **Date** | **Goals** | **Status** |
| January 30, 2020 | * Prepare interim progress report **(Deliverable 2)** * Preliminary implementation **(Deliverable 2)** | Done |
| February 15, 2020 | * Deploy initial machine learning approach * Deploy models based on neural networks * Analyse and evaluate performance of models * Start improving on top models | Pending |
| February 28, 2020 | * Deploy ensemble models and evaluate performance | Pending |
| March 15, 2020 | * Testing and improvements on models | Pending |
| March 30, 2020 | * Final review of software * Make changes if needed | Pending |
| April 19, 2020 | * Prepare final report **(Deliverable 3)** * Prepare final presentation **(Deliverable 3)** | Pending |
| April 30, 2020 | * Prepare project poster | Pending |

# **6. CONCLUSION**

This project aims to create a tool that provides the service of forecasting financial data in the domain of cross-market analysis. Forecasts of market indices on an international level are to be obtained using historical time series data along with currency exchange rates, public news and social media sentiment. Previous works have been reviewed to gain a deeper understanding of the methodologies used and decide on the approach to be pursued.

We identified the importance of discovering the hidden relationships between various parameters and input data. Correctly detecting these relationships are imperative to the success of this project. Moreover, effective feature extraction can aid in the process of developing machine learning models with higher accuracy. Data analysis, feature extraction and building models is an iterative process that will constitute most of the project.

At the current stage, scrapers have been set up to obtain the different types of data required and a sentiment analysis model has been identified for the processing of text data. Usage of Python modules has enabled the reuse of existing codebase to facilitate the data collection process.

An architecture for the project has been set up to enable the implementation of the following phases of the project involving the pre-processing and cleaning of data along with some exploratory data analysis. Despite the limitations in some aspects of the project, work is being continued to develop a tool successful in generating market forecasts using financial data. In later stages of this project, we aim to work towards building a new model that can be used to make financial predictions.

# **REFERENCES**

[1] G. Zhang, *Neural Networks in business forecasting*. USA: IGI Publishing, 2004.

[2] A. Ludhiyani, S. Joshi, R. Pathak, P. Parandkar, and S. Katiyal, "Quant techniques and regime analysis for increase in trade & business between India and Israel," in *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2015: IEEE, pp. 442-447.

[3] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[4] L. C. Boldt et al., "Forecasting Nike's sales using Facebook data," 2016 IEEE International Conference on Big Data (Big Data), Washington, DC, 2016, pp. 2447-2456.

[5] W. Chong and C. Pu, "Application of Support Vector Machines in Paying Rate Forecasting," 2006 International Conference on Management Science and Engineering, Lille, 2006, pp. 1494-1497.

[6] E. K. Laitinen and T. Laitinen, "Bankruptcy prediction: Application of the Taylor's expansion in logistic regression," *International review of financial analysis,* vol. 9, no. 4, pp. 327-349, 2000.

[7] M. R. Hassan and B. Nath, "Stock market forecasting using hidden Markov model: a new approach," in *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*, 2005: IEEE, pp. 192-196.

[8] W. Wang and S. Nie, "The Performance of Several Combining Forecasts for Stock Index," 2008 International Seminar on Future Information Technology and Management Engineering, Leicestershire, United Kingdom, 2008, pp. 450-455.

[9] J. McCluskey and Jianguo Liu, "US financial market forecasting using data classification with features from global markets," 2017 2nd International Conference on Image, Vision and Computing (ICIVC), Chengdu, 2017, pp. 965-969.

[10] A. Mittal and A. Goel, "Stock prediction using twitter sentiment analysis," *Stanford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis. pdf),* vol. 15, 2012.

[11] K. Zhang, L. Li, P. Li, and W. Teng, "Stock trend forecasting method based on sentiment analysis and system similarity model," in *Proceedings of 2011 6th International Forum on Strategic Technology*, 2011, vol. 2: IEEE, pp. 890-894.

[12] J. Si, A. Mukherjee, B. Liu, Q. Li, H. Li, and X. Deng, "Exploiting topic based twitter sentiment for stock prediction," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2013, pp. 24-29.

[13] S. Yıldırım, D. Jothimani, C. Kavaklıoğlu and A. Başar, "Classification of "Hot News" for Financial Forecast Using NLP Techniques," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 4719-4722.

[14] “中文文本探勘初探：TF-IDF in R Language,” Mr. Opengate, 14-Apr-2016. [Online]. Available: https://mropengate.blogspot.com/2016/04/tf-idf-in-r-language.html. [Accessed: 05-Dec-2019].

[15] S. Pan, X. Jiang, J. Jiang and G. Long, "Cross-Domain Deep Learning Approach For Multiple Financial Market Prediction," presented at the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, 2018.

[16] X. Ding, Y. Zhang, T. Liu, and J. Duan, "Deep learning for event-driven stock prediction," in *Twenty-fourth international joint conference on artificial intelligence*, 2015.

[17] “Making Web Crawlers Using Scrapy for Python,” *DataCamp Community*. [Online]. Available: https://www.datacamp.com/community/tutorials/making-web-crawlers-scrapy-python.