BT4222 Mining Web Data for Business Insights
Group Project

Group 19

Ang Guo Xiong	A0199646H
Ching Zheng Ing	A0202003L
Darius Seah Kuan Wei	A0199491L
Lim Huai Xing	A0202965A
Leung Hoi Kit Alvin	A0201468J

1. Background

1.1 Introduction

Stock market prices and trends are extremely volatile in nature in the finance industry. Numerous models have been devised in order to capture the volatile characteristics of the stock market, with the main goal to forecast or predict market trends. However, the development of a consistently accurate stock forecasting model remains difficult due to the vast number of factors affecting stock prices[1].

In general, there are two methods for forecasting market trends - technical analysis and fundamental analysis[2]. Technical analysis consists of past prices and volumes to predict future trends whereas fundamental analysis involves analyzing a company's financial data and financial news to get some insights.

Financial news articles on a particular company typically explain its performance and possibly signal which direction its stock price moves towards. With the proliferation of the internet, the propagation of news has been facilitated. As such, not only do the news carry more weight in affecting stock prices (and movement) since the information is more accessible and available, the advancement of technology also eases the process of data mining for such information[3].

1.2 Project Objectives

This project augments fundamental analysis techniques to technical analysis by incorporating information from news articles related to our selected companies into more traditional technical analysis such as exponential moving average price (EMA). It integrates both types of analysis to achieve a better prediction while understanding the impact of financial news to market trade directions.

In the fundamental aspect, this project focuses on non-quantifiable data such as financial news and articles about a company to predict future stock trends using different techniques. Sentiment analysis is performed on these news content for selected companies to classify news as positive or negative. Market sentiment is a qualitative measure of investors behaviour and mood, which drives price action and investment opportunities[4]. The sentiment scores serve as features to traditional models as additional variable(s) in an attempt to improve the model's predictive powers. As such, textual data gleaned from multiple sources are feature engineered to complement the existing technical indicators in the predictions using regression, classification, and time series.

We hypothesized that news data are likely to have varying impacts on different industries. In particular, we believe that the technology industry should be more sensitive to news articles as it is a booming industry as compared to the more mature consumer industry.

1.3 Further Refinements

In addition to incorporating sentiment analysis as an indicator of how the market is reacting to current events, other techniques such as topic modelling and using Google Trends data are explored in aim to further refine our models. We conjecture that the use of these methods will definitely be helpful in improving forecasting accuracy.

For topic modelling, identifying topics with high correlation to stock price changes such as 'Economic Recession', could serve as a relatively strong indicator in predicting day to day changes in stock price, which goes beyond the results of analyzing just basic trends. On the other hand, Google Trends data indicates how popular a search term is intrinsically tied to the public excitement and curiosity towards a certain company, as searches for their company name should increase upon the release of positive news.

2. Data Preparation

2.1 Data Source

The data to be used in this project is obtained from three sources - Markets Insider, CNBC news, and a Kaggle[5] dataset (collated from investing.com). News content from Markets Insider was scraped from its website using BeautifulSoup, while news content from CNBC was scraped using the back-end API of CNBC in python. Complementing the existing data from Kaggle is imperative here as a comprehensive set of news or textual data is required to improve the reliability and accuracy of our model training results. Furthermore, news headlines from the Kaggle dataset were not used as they alone may not be reflective of the actual content, which might cause the sentiment analysis to predict incorrectly. For example, the news headline "Apple needs a bit of love" is likely to be interpreted as a positive sentiment due to the word "love". However, the actual news sentiment is likely negative. Furthermore, more data should improve the reliability of the resulting models used since there is more news in the same day/period to reflect the overall sentiments more accurately (by averaging for each day).

2.2 Data Cleaning & Adjustment

In addition, since there can be duplicate news when combining the three sources, duplicates in the text column are dropped to ensure each row's text is unique. To better capture sentiments on trading days after weekends and public holidays, we adjusted the respective dates to the next trading day before combining the three sources (Market Insider, CNBC news, Kaggle) to form our news dataset for each stock to reflect the effect of sentiments on the next trading day more accurately. Lastly, the sentiments are combined with market prices and data.

2.3 Stock Price Data

The historical stock market prices (High, Low, Open, Close) and volume were taken from Yahoo Finance. Movement of stock was determined by comparing today's closing price to the closing price of the previous day, in which 1 shows an increase in stock price and 0 otherwise. We merged both the sentiment scores from the news data and historical stock market prices to obtain our compiled input data set to be used.

3. Stock Market and Models used

The project studies Procter & Gamble, Apple and Facebook stocks. These companies represent the different industries we are exploring; Procter & Gamble (ticker: PG) is largely a consumer based company, while Facebook (ticker: FB) is an information technology firm. Apple (ticker: AAPL) represents the middle ground between the two industries as it has a mixture of the type products and processes from both industries.

The models were generally either predicting for the Movement of Closing Price (categorical), or the Closing Price (regression). For categorical models, Extreme Gradient Boost was chosen for its fast execution speed with generally strong model performances (being part of many machine learning challenges' winning solutions[6]), and Logistic Regression for its ease of implementation and interpretation[7]. For regression models, Neural Network was selected for its strong learning of data during the training process with gradient descent and error backpropagation algorithms. Multivariate LSTM was included to see if capturing long-term dependencies within variables in the market can help improve price predictions. Lastly, Linear Regression model is included to better capture linear relationships between market variables and prices.

Beyond these 5 models, we also included SARIMAX in an attempt to better capture time-trends and seasonality that may potentially exist within the markets. Also, a CNN model was included to analyze textual data and predict movement of stock prices.

Prediction	Models Used
Movement (categorical)	Extreme Gradient Boost (XGB) Logistic Regression
Closing Price (regression)	Neural Network Multivariate Long Short Term Memory (LSTM) Linear Regression
Time Trends & Seasonality	Seasonal Auto-Regressive Integrated Moving Average with eXogenous factors (SARIMAX)
Text Classification	Convolutional Neural Network (CNN)

4. Methodology

The methodology used is categorized into 4 main phases. In phases 1 to 3, only 1 stock will be used for training and evaluating the model, as the focus is on features and model selection. AAPL stock's data will be used as the data for models in phases 1 to 3 as it represents the middle ground between the two industries as highlighted in section 3.

4.1 Phase 1

This phase consists of the training of base models. The above mentioned models are first trained with technical indicators to predict the movement of stock. The technical indicators selected are indicators that are being used widely currently. They can be categorized into trends, signals and others.

4.1a Trend:

Trend indicators indicate which direction the market is moving and are commonly used to gauge trends within the market. Moving Average is a trend indicator that smooths out price data constantly by making average prices that eliminates variations due to random price fluctuations.

- Simple Moving Average (SMA) calculates the average of a selected range of prices, by the number of periods in that range. The popular SMA window size of 20, 50, and 100 will be used to reflect the common trend signals seen by investors.
- Exponential Moving Average (EMA) is a type of moving average that places a greater weight and significance on the most recent data points. Similarly, commonly used EMA window size of 10, 20 and 40 will be used to reflect trend signals observed by investors.

4.1b Signal:

Signal indicators are often used to provide buy-sell signals for investors as they reflect changes in momentum of the market as well as warning signs for dangerous price movements.

- Moving average convergence divergence (MACD) is a trend-following momentum indicator that shows the relationship between two moving averages of a security's price. The MACD is calculated by subtracting the 26-period exponential moving average (EMA) from the 12-period EMA.
- The relative strength index (RSI) is a momentum indicator used in technical analysis that measures the magnitude of recent price changes to evaluate overbought or oversold conditions in the price of a stock or other asset.

4.1c Others:

Besides popular Trend & Signal indicators used in technical analysis, fundamental market conditions are also included such as the 10-year Treasury Yield which is closely watched as an indicator of broader investor confidence[8].

- 10-year Treasury Yield (TNX) - as an indicator of interest rate & inflation rate expectation
- Volume of Trades

We hypothesise that these widely used technical indicators [SMA (20,50,100), EMA (10,20,40), MACD, RSI, TNX, Volume] can form a basis on which investors look to buy-sell their holdings. Hence, in phase 1, the best-performing selection of indicators will be used for model training and the technical indicators for the models are chosen based on the following combinations:

1. SMA (20, 50, 100)
2. EMA (10, 20, 40)
3. Better performing of either SMA or EMA + MACD + RSI
4. Better performing of either SMA or EMA + MACD + RSI + TNX + Volume

After determining a suitable combination of technical indicators from Phase 1, the next phase will investigate the relationship between sentiment in news articles and stock prices.

4.2 Phase 2

Sentiment analysis is performed on our news data as well as data supplemented from the Kaggle dataset to determine the sentiment of news for each day. A sentiment score is produced for each article. This was done using one of the most popular Natural Language Processing (NLP) libraries[9] - Natural Language Toolkit (NLTK) Valence Aware Dictionary and Sentiment Reasoner (VADER) sentiment analysis tool. The score is then added as a variable to the model with the selected technical indicators from Phase 1 to check if the model predictions improves.

4.3 Phase 3

After incorporating sentiment analysis into the model (if suitable), Phase 3 focuses on improving model prediction through capturing more information on the market using additional methods such as Topic Modelling and Google trends data to the base model. Topic modelling was selected as it provides us with methods to organize, understand and summarize large collections of textual information[10]. This can complement the sentiment scores that are used in Phase 2 through discovering hidden topical patterns that are present across the collection. It was also hypothesized that stock prices may be related with general market interest of the stock which can be captured through the historical search volume on the largest web search engine - Google. Therefore, this will be integrated into the model in Phase 3.

4.4 Phase 4

As Phase 1 to 3 focuses on AAPL stock (being the middle ground between Consumer-based and a Information Technology company), Phase 4 attempts to find out if the models perform differently across P&G and Facebook to compare differences in model performances (if any) across industries and test the adaptability of the models to various company market data.

5 - 8 Training of Models

5. Phase 1 - Base Models with Technical Indicators

Section	Prediction	Models Used
5.1	Movement (categorical)	Extreme Gradient Boost (XGB) Logistic Regression
5.2	Closing Price (regression)	Neural Network Multivariate Long Short Term Memory (LSTM) Linear Regression

5.1 Base [Categorical] Model for Predicting Movement of Closing Price

Classifiers were used to build prediction models to predict the movement of closing price . In this section, 2 classifiers were used, a simple Logistic Regression (LR) model and the more popular Extreme Gradient Boosting (XGB) model. For closing price movement, 2 class labels were used here - '1' representing upwards movement of closing price, and '0' representing downwards movement.

All models are evaluated based on F1 score and AUC score, with the latter being prioritised. These 2 metrics were chosen because the class labels are imbalanced, thus F1 score would be better than Accuracy. Since we are exploring the effects of each set of features, we do not want to assume a classification threshold. In addition, we are interested in accurate predictions of both positive and negative classes[11]. Thus, AUC score will be the main evaluation metric.

For the first part of feature selection, a classifier with only the SMA (20, 50, 100 Days) features and separately with only the EMA (10, 20, 40 Days) features were tested. Between these 2 sets of features, SMA produces the best AUC score for the XGB model, while for the LR model, EMA produces the best AUC score. Hence, SMA will be used for XGB while EMA will be used for LR in the next set of feature additions. For the second part of feature selection, adding MACD and RSI to MA shows an improvement in AUC score for the XGB model, but not for the LR model. For the third part of feature selection, adding 10 year Treasury Yield and Trading Volume to the previous set of features, did not show an improvement in AUC score for both XGB and LR models. Ergo, these features will not be used.

Overall, the best features found in phase 1 are SMA together with MACD and RSI for the XGB model, and EMA alone for the Logistic Regression model. Between the 2 models, XGB provided better results (XGB: 0.534 vs LR: 0.526). As such, further tuning and optimisation will be done on XGB only and the model will be used in subsequent phases.

Extreme Gradient Boosting	F1 Score	AUC Score
SMA	0.247	**0.511**
EMA	0.267	0.495
With SMA, MACD & RSI	0.0788	**0.534**
With SMA, MACD, RSI, TNX & Volume	0.14	0.53

Logistic Regression	F1 Score	AUC Score
SMA	0.676	0.515
EMA	0.642	**0.526**
With EMA, MACD & RSI	0.608	0.519
With EMA, MACD, RSI, TNX & Volume	0.649	0.51

Due to the large number of tunable parameters, Bayes Optimisation will be used for XGB model tuning. Figure below shows the parameter range for tuning.

```
'n_estimators': scope.int(hp.quniform('n_estimators', 50, 550, 20)),
'max_depth': scope.int(hp.quniform('max_depth', 1, 25, 1)),
'learning_rate': hp.uniform('learning_rate', 0.01, 0.5),
'booster': hp.choice('booster', ['dart','gbtree']), #gblinear
'gamma': hp.uniform('gamma', 0, 20),
'min_child_weight': hp.uniform('min_child_weight', 1, 5),
'subsample': hp.uniform('subsample', 0.1, 0.9),
'colsample_bytree': hp.uniform('colsample_bytree', 0.1, 0.9),
'colsample_bynode': hp.uniform('colsample_bynode', 0.1, 0.9),
'colsample_bylevel': hp.uniform('colsample_bylevel', 0.1, 0.9),
'reg_lambda': hp.uniform('reg_lambda', 1, 5),
'reg_alpha':hp.uniform('reg_alpha', 0.01, 0.1),
'scale_pos_weight':hp.uniform('scale_pos_weight', 1, 10),
'use_label_encoder':False,
'random_state': 1
```

We will be maximising AUC score, and 500 evaluations will be used. Based on this optimization, both the F1 score and AUC score improved - F1 score: 0.476, AUC score: 0.570.

5.2 Base [Regression] Model for Predicting Closing Price

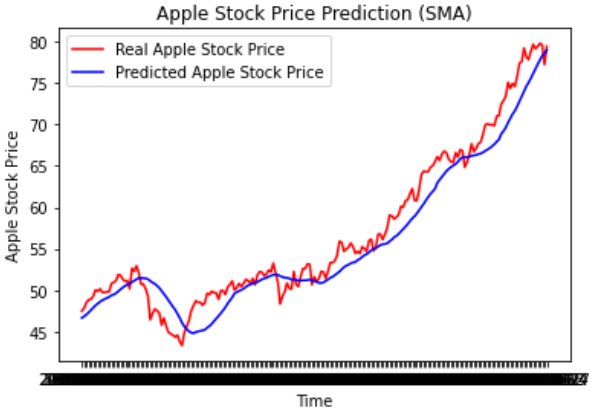
To predict Closing Price, regression models were used as the prediction models. In this section, 3 regression models were used, namely a simple Linear Regression model, a Neural Network model and a Multivariate Long Short-Term Memory (LSTM) model.

As the Closing Prices were heavily affected by fear and uncertainty brought by Covid-19 during the March 2020 crash, the regression models are trained and predicted on data until Jan 2020 in view of achieving the best model performance.

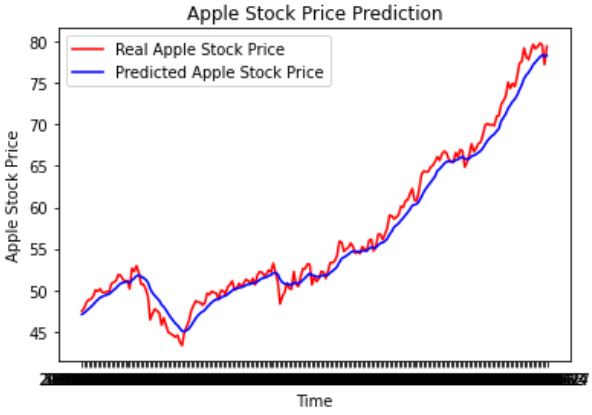
All models are evaluated based on Mean Squared error, R-squared and Adjusted R-squared.

5.2a Neural Network

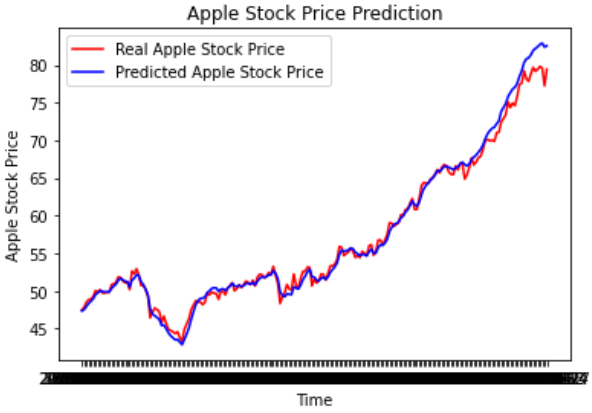
The neural network was built with 2 dense layers with 10 neurons each and fitted at 50 epochs. Scaling was applied to the training and test data as the neural network utilises gradient descent based algorithms. Hence scaling ensures that the gradient descent moves smoothly towards the minima and that the steps for gradient descent are updated at the same rate for all the features. The predicted stock price is then plotted as shown below.



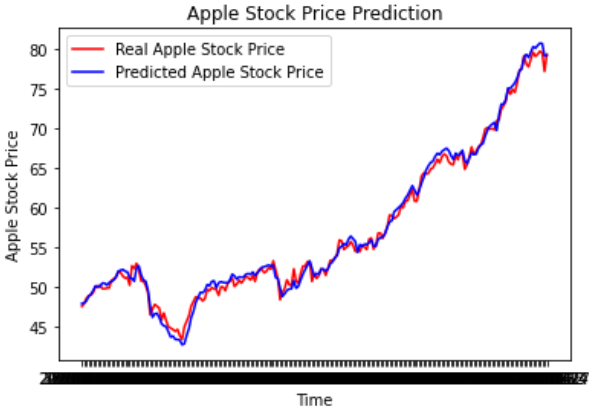
SMA (20,50,100)



EMA (10,20,40)



EMA (10,20,40) + MACD + RSI



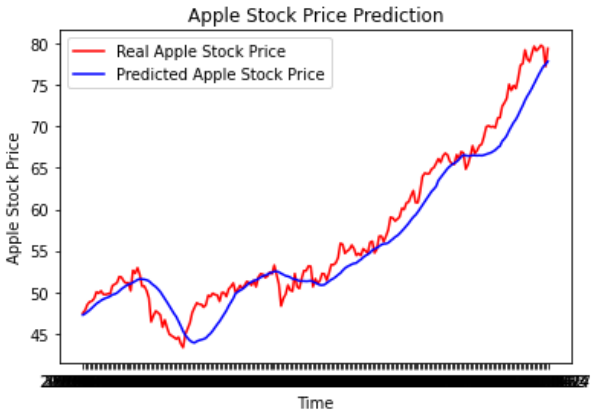
EMA (10,20,40) + MACD + RSI + TNX + Volume

Neural Network	SMA	EMA	With MACD & RSI	With TNX & Volume
Mean Squared Error	5.21	1.83	1.02	**0.60**
R-Squared	0.943	0.980	0.989	**0.993**
Adjusted R-Squared	0.942	0.980	0.988	**0.993**

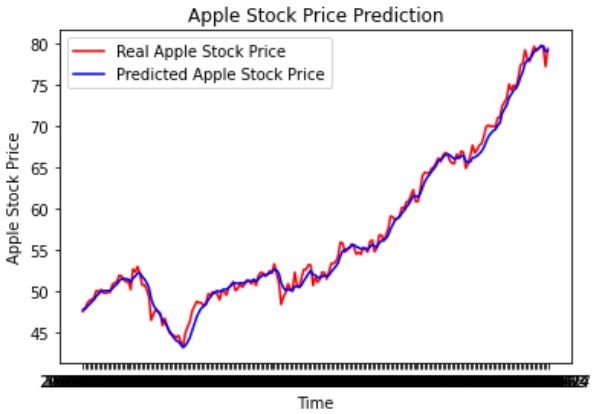
The Neural Network model with EMA data significantly decreases the MSE and increases both the R-Squared and adjusted R-Squared values as compared to using SMA data. This suggests that exponentially weighted prices give relatively higher predictive power. By including MACD & RSI as indicators of buy-sell signals in the market as well as 10-year Treasury Yield as an indicator of interest and inflation rate, the model performance further improves to MSE of 0.60.

5.2.b Multiple Linear Regression

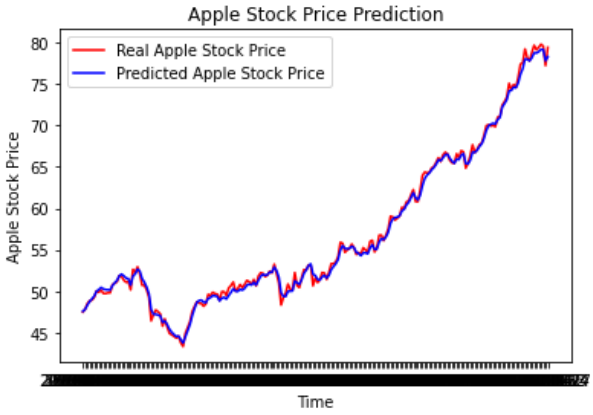
The Linear Regression model was also chosen to investigate the relationship between Closing Price and trend indicators (SMA & EMA), signal indicators (MACD & RSI), Treasury yield as well as volume data. The predicted price is plotted with the actual price as shown below.



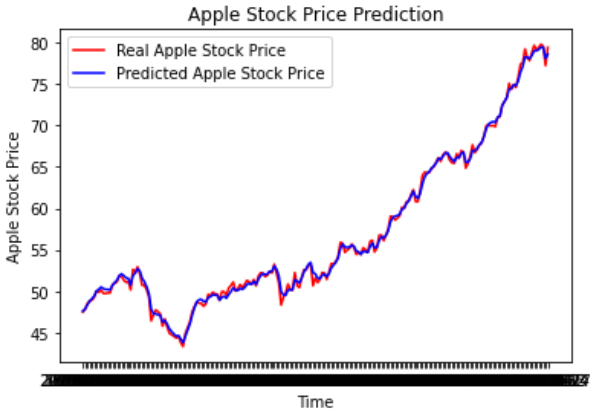
SMA (20,50,100)



EMA (10,20,40)



EMA (10,20,40) + MACD + RSI



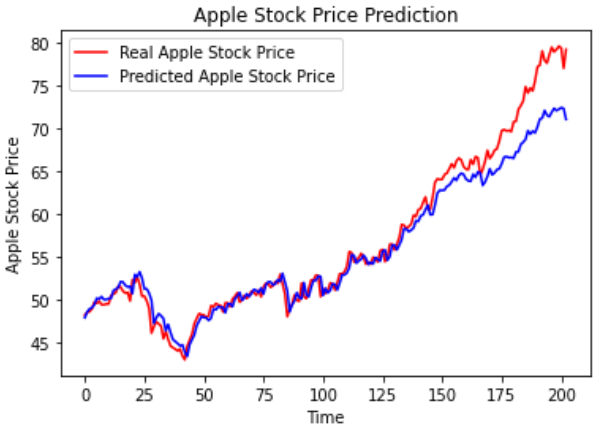
EMA (10,20,40) + MACD + RSI + TNX + Volume

Linear Regression	SMA	EMA	With MACD & RSI	With TNX & Volume
Mean Squared Error	5.670	0.612	0.218	**0.181**
R-Squared	0.938	0.993	0.998	**0.998**
Adjusted R-Squared	0.937	0.993	0.998	**0.998**

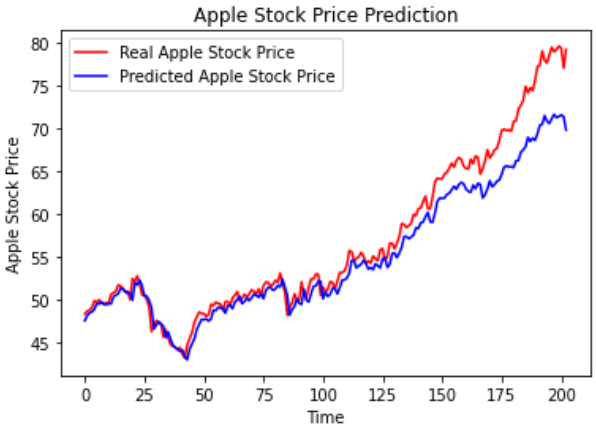
Similarly for Linear Regression, EMA price data gives better model performance over SMA price data. Adding MACD, RSI, Treasury yield and volume also further improved the model to give a MSE of 0.181. Overall, the lower MSE of the Linear Regression model suggests better model performance over the Neural Network model.

5.2.c Multivariate LSTM

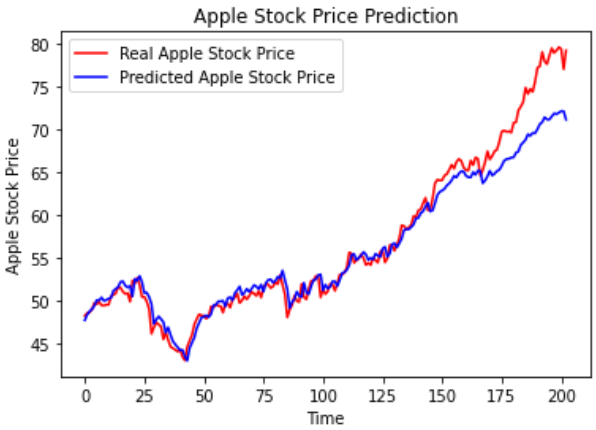
The LSTM model was included as the market prices are sequential Time-series data and to observe if the memory capacity of LSTM models are useful in helping to predict Closing Price. The model was built with 100 LSTM units, with 0.2 dropout to reduce overfitting, which was then passed through a neural network with the same number of neurons as LSTM units for better learning through error backpropagation. The predicted stock prices are plotted below.



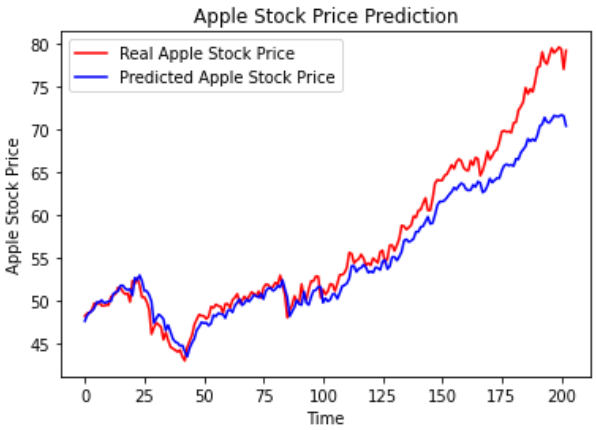
SMA (20,50,100)



EMA (10,20,40)



SMA (20,50,100) + MACD + RSI



SMA (20,50,100) + TNX + Volume

Multivariate LSTM	SMA	EMA	With MACD & RSI	With TNX & Volume
Mean Squared Error	**5.72**	8.35	5.89	8.10
R-Squared	**0.937**	0.908	0.936	0.911
Adjusted R-Squared	**0.936**	0.906	0.933	0.909

In contrast to both the Neural Network and Linear Regression models, the SMA price data works better for the multivariate LSTM model. This could be due to the exponential weighted price values being difficult for the LSTM to detect long-term dependencies. Additionally, adding MACD, RSI, Treasury yield and volume data does not seem to improve the MSE and instead reduces the R-Squared values. This seems to suggest weak long-term dependencies between closing price and the price moving averages and the other trading indicators used.

Overall, Linear Regression performs better than Neural Network and Multivariate LSTM with the latter having the highest MSE. Next, sentiment scores are included in the set of predictor variables to investigate if there exists a strong relationship between investors' buy-sell actions and market sentiment.

6. Phase 2 - Incorporating Sentiment Analysis

In this phase, results of sentiment analysis of news will be added to models selected in phase 1. Sentiment analysis is generated with the help of NLTK, specifically the VADER sentiment analysis tools. This Sentiment Analysis package produces 4 outputs, a probability of ‘Negative’, ‘Netural’, and ‘Positive’ sentiment together with a compound score which is calculated based on the probabilities of the 3 sentiments. As mentioned in Data Preparation, the sentiment scores are generated based on the brief description of the article.

Section	Prediction	Models Used
6.1	Movement (categorical)	Extreme Gradient Boost (XGB)
6.2	Closing Price (regression)	Neural Network Multivariate Long Short Term Memory (LSTM) Linear Regression
6.3	Time Trends & Seasonality	Seasonal Auto-Regressive Integrated Moving Average with eXogenous factors (SARIMAX)
6.4	Text Classification	Convolutional Neural Network (CNN)

6.1 Optimised [Categorical] Model for Predicting Movement of Closing Price

Based on the optimised model from phase 1, we will now add in Sentiment Analysis result to the model as an additional feature, to find out if news sentiment has an additional effect on closing price.

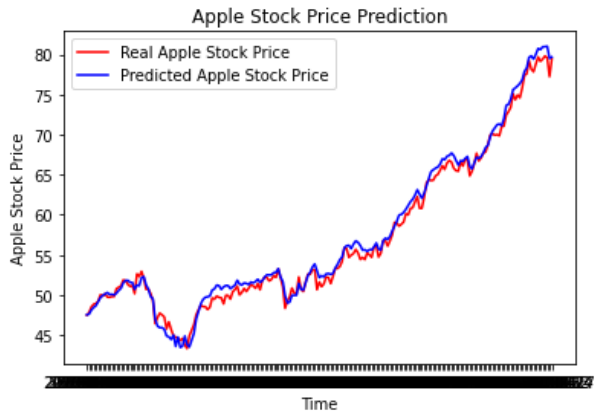
The model after adding sentiment score will be optimised with the same process as in phase 1, using Bayes optimisation with 500 evaluations, maximising the AUC score. Based on the results below, it shows that adding a Sentiment score did improve the AUC score and F1 score. This suggests that news sentiment does have an impact on closing price movement.

Optimised XGB	F1 score	AUC score
Best features from phase 1	0.476	0.570
With Sentiment	0.605	**0.584**

6.2 Optimised [Regression] Model for Predicting Closing Price

Similarly, the sentiment scores are also incorporated into the various regression models to see if it improves the model performance. The 'Negative', 'Netural', and 'Positive' sentiment as well as the compound score was added as variables to each of the models below.

6.2.a Neural Network

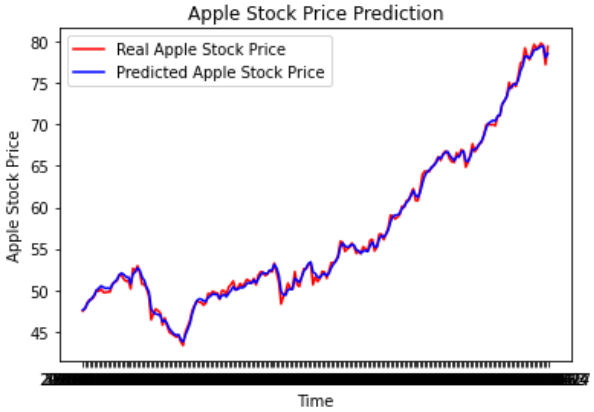


EMA (10,20,40) + MACD + RSI + TNX + Volume with Sentiment (NN)

Optimised Neural Network	MSE	R^2	Adjusted R^2
Best features from Phase 1	**0.60**	0.993	0.993
With Sentiment	0.83	0.991	0.990

Adding the sentiment scores did not seem to improve the Neural Network model as seen by the higher MSE and lower R-Squared values. This can be somewhat counterintuitive as we would expect sentiment scores to improve model predictions. Nonetheless, this could also suggest that the market is less affected by sentiment of news articles and that other trading indicators are more relevant. This is possible as news can be reactionary and therefore moves after market prices, thus using the former to predict the latter may show little improvements.

6.2.b Multiple Linear Regression

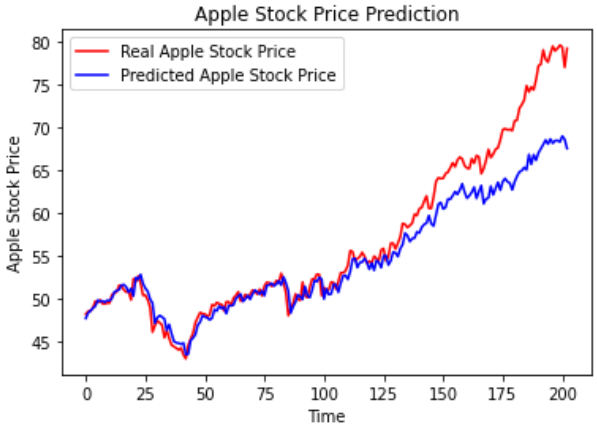


EMA (10,20,40) + MACD + RSI + TNX + Volume with Sentiment (Linear Regression)

Optimised Linear Regression	MSE	R^2	Adjusted R^2
Best features from Phase 1	0.181	0.998	0.998
With Sentiment	**0.177**	0.998	0.998

As compared to the neural network model, sentiment scores improved the model predictions for Linear Regression. However, the improvement is only by a very minute amount as indicated by the 0.004 improvement in MSE, which is almost negligible. Hence, sentiment scores may not have as big an impact as we may have previously assumed.

6.2.c Multivariate LSTM



SMA (20,50,100) with Sentiment (LSTM)

Multivariate LSTM	MSE	R^2	Adjusted R^2
Best features from Phase 1	**5.72**	0.937	0.936
With Sentiment	14.05	0.846	0.840

Including sentiment scores in the LSTM model greatly decreases the model prediction performance. This could be due to the sentiment scores not accurately capturing the magnitude of the price increase as seen by the gap between the Actual (red) and Predicted (blue) price in the above figure. Another reason could be a weak dependency between older and current sentiments that may not have been well-captured by the LSTM model.

Overall, the LSTM model has significantly higher MSE as compared to Linear Regression and Neural Network models for regression predictions. Thus, it will not be further built on in Phase 3.

6.3 Optimised [Time Series] Model for Predicting Closing Price

A time series is a series of data points indexed based on linear time order. Alternatively, a time series is a sequence taken at successive spaced points in time. Given this definition, stock price data can be considered a time series and time series forecasting can be used to try and forecast future values.

Seasonal Autoregressive Integrated Moving Average, or SARIMA, is one of the most widely used forecasting methods for univariate time series data forecasting. It belongs to a family of models that aim to predict future values based on previously observed values while taking into account both overall trends in the data and seasonalities present.

In order to configure a SARIMA model, we have to find out what hyperparameters are suitable to model both the trend and seasonal elements of the underlying time series, which in this case is the daily closing price of AAPL.

We choose 3 hyperparameters for the trend elements. The first of them being the trend autoregressive order, which is the number of immediately preceding values in the time series that are used to predict the value at the present time. The second hyperparameter is the trend difference order. This is the number of differencing procedures required to get a stationary series from the original time series data, where differencing is the transformation of the series to a new time series where the values are the differences between consecutive values of the series. Finally, the moving average order is the series of averages of different subsets of the full data set in order to smooth out the influence of outliers.

Next, for the seasonal elements, similar to the 3 hyperparameters for the trend elements, there is also a seasonal autoregressive order, seasonal difference order and seasonal moving average order. Additionally, we also took into account the number of time steps for a single seasonal period.

In summary, we can specify the notation for an SARIMA model as SARIMA(p,d,q)(P,D,Q)m, where:

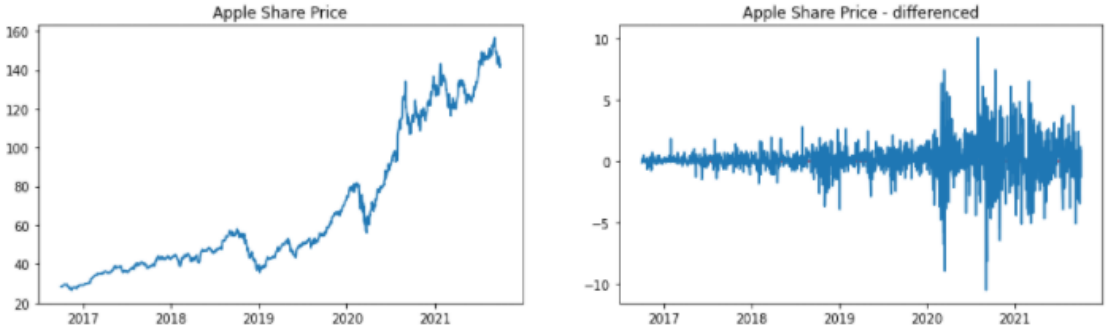
- p: Trend autoregression order
- d: Trend difference order
- q: Trend moving average order
- P: Seasonal autoregressive order
- D: Seasonal difference order
- Q: Seasonal moving average order
- m: The number of time steps for a single seasonal period

6.3.a Finding Model Parameters

After importing the stock ticker data for Apple between 2016 to 2021, we split the data into our training and testing sets, with the training set consisting of the closing prices between 1st October 2016 to 1st October 2020 (1008 days) and the testing set being the subsequent year up to 1st October 2021 (252 days). The number of rows in our stock price data is a multiple of 252 because that is the number of days in each trading year.

Graph of Apple stock's closing price from 2016 - 2021

After visualising Apple's stock prices, we proceed to select the hyperparameters for our SARIMA(p,d,q)(P,D,Q)m model. From the graph, it is clear that Apple's share price contains an upwards trend. Differencing is used to see if a stationary process with mean 0 is obtained.



Graphs of Apple stock's price and differenced share price

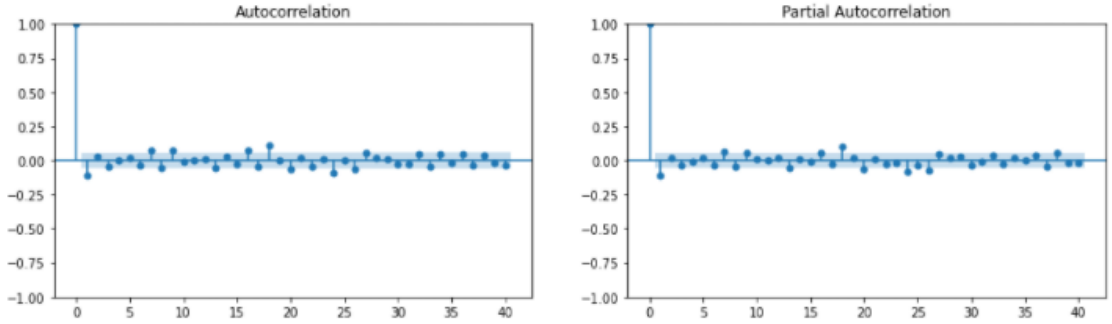
The share price appears to cluster around 0 after differencing the time series once, giving us d = 1 in our model. This was confirmed by running an Augmented Dickey-Fuller test on both the Apple closing stock prices and the differenced Apple closing stock prices, where the null hypothesis is that the data has a unit root and is non-stationary.

```
Results for Apple closing stock prices
ADF Statistic: 0.122152
p-value: 0.967541
Critical Values:
        1%: -3.436
        5%: -2.864
        10%: -2.568
Results for Differenced Apple closing stock prices
ADF Statistic: -7.296031
p-value: 0.000000
Critical Values:
        1%: -3.436
        5%: -2.864
        10%: -2.568
```

Results of Augmented Dickey-Fuller test

From the Augmented Dickey-Fuller test, the p-value for the test run on the differenced Apple closing stock prices is 0.00000 < 0.05, which means that we can reject the null hypothesis at 5% significance level that the data does not have a unit root and conclude that it is stationary. However we cannot do so for the original time series where the p-value is 0.967541 >= 0.05. Thus we can set d = 1.

Next, to figure out the Trend autoregression order (p) and Trend moving average order (q) we plotted the autocorrelation function (ACF) and the partial autocorrelation function (PACF) of the differenced stock price and see which time steps are significant.



ACF and PACF plots of the differenced Apple closing stock prices

From these plots, we observe that the significant time steps, besides 1 and 18, all appear to be very close to being insignificant. As such, to expedite the process of choosing the parameters, we utilized the auto_arima function to automatically discover the optimal order for an ARIMA model.

6.3.b Training SARIMA Model and Forecasting

Auto-ARIMA works by conducting differencing tests to determine the order of differencing, d, and then fitting models within ranges of defined start_p, max_p, start_q, max_q ranges. We

enabled the seasonal optional so auto-ARIMA will also seek to identify the optimal P and Q hyper-parameters after conducting the Canova-Hansen, to determine the optimal order of seasonal differencing, D.

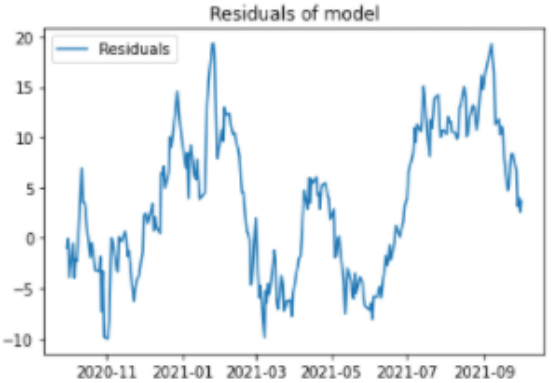
```
Performing stepwise search to minimize aic
 ARIMA(2,1,2)(0,0,0)[0] intercept   : AIC=3372.215, Time=0.20 sec
 ARIMA(0,1,0)(0,0,0)[0] intercept   : AIC=3382.421, Time=0.02 sec
 ARIMA(1,1,0)(0,0,0)[0] intercept   : AIC=3367.800, Time=0.04 sec
 ARIMA(0,1,1)(0,0,0)[0] intercept   : AIC=3369.130, Time=0.05 sec
 ARIMA(0,1,0)(0,0,0)[0]             : AIC=3385.051, Time=0.01 sec
 ARIMA(2,1,0)(0,0,0)[0] intercept   : AIC=3368.366, Time=0.07 sec
 ARIMA(1,1,1)(0,0,0)[0] intercept   : AIC=3368.243, Time=0.12 sec
 ARIMA(2,1,1)(0,0,0)[0] intercept   : AIC=3370.199, Time=0.25 sec
 ARIMA(1,1,0)(0,0,0)[0]             : AIC=3371.746, Time=0.02 sec

Best model:  ARIMA(1,1,0)(0,0,0)[0] intercept
Total fit time: 0.788 seconds
```

Auto_arima doing stepwise search to find optimal model parameters

After finding the optimal parameters - SARIMA(1,1,0)(0,0,0)[0] model, we forecasted the predicted closing stock prices for the next year for comparison with our test set of data.

Apple stock's closing price from 2016 - 2021 (with predicted stock prices in green)

```
MSE: 62.851237
```



MSE and residuals from the predictions

It is clear that the model has not done exceptionally well in predicting the future stock prices. Although it has captured the trend, there are many remaining residual errors present at each time step, resulting in a mean squared error of 62.851. Such an error is quite high considering Apple's stock closing price during the period of about $140.

6.4 [Text Classification] Model for Predicting Movement of Closing Price

Apart from using XGB as a classifier, we attempted Convolution Neural Network (CNN) as well, as it is appropriate with textual data. In this case, instead of using Sentiment score together with features finalised in phase 1, we built a multivariate CNN model with the textual data as one

input, with the other inputs being the finalised features from phase 1. The rationale for leaving out the Sentiment score is to allow CNN to extract out information from the textual data that may already capture elements of the news sentiment. As a result, we want to avoid overweighting the effect of news sentiments.

Different CNN architectures were tested and adjusted to reduce overfitting. However, the best CNN model only resulted in a F1 score of 0.682 and AUC score of 0.516, which suggests that the model is only slightly better than a simple random guess model. This shows that CNN model does not work well for predicting closing price movement.

7. Phase 3 - Incorporating Topic Modelling & Google Trends

Section	Content / Prediction	Models Used
7.1	Topic Modelling	-
7.2	Google Trends Data	-
7.3	Movement (categorical)	Extreme Gradient Boost (XGB)
7.4	Closing Price (regression)	Neural Network Linear Regression
7.5	Time Trends & Seasonality	Seasonal Auto-Regressive Integrated Moving Average with eXogenous factors (SARIMAX)
7.6	Text Classification	Convolutional Neural Network (CNN)

7.1 Topic Modelling

In this phase, Topic Modelling and Google Trends will be added to the models from phase 2.

Topics generated from topic modelling are used as features to complement the existing features in the prior phases to determine whether the type of topics improve the predictions. To reduce the amount of unnecessary and distracting texts before fitting into the Latent Dirichlet allocation(LDA) model, text preprocessing is done.

A series of preprocessing steps are done to glean the topics. Firstly, we utilised *gensim* package's *simple_preprocess* to tokenize documents, convert to lowercase and remove punctuations in them. Stop words from the NLTK package are removed from this set of documents for each stock. Additional words are added to this list of words to reduce the likelihood of common but not meaningful words affecting the resulting topics generated. Examples are namely the stock tickers, company names as well as website prefixes. Moreover,

bigrams are created to increase the reliability in our methodology. Finally, lemmatization is employed to convert each word into its base form.

We experimented with a different number of topics and decided to set the number of topics to be 10. The rationale behind is to reduce chances of words and/or content overlapping in some topics generated; the likelihood of overlapping increases as the number of topics increases, which makes interpreting and distinguishing the topics difficult. We trained the LDA model with the specified number of topics and interpreted them. We used the top 10 more prominent words in each topic to label it as follows:

Interpreted Topic	**Top 10 words**
Investors' Sentiment	0.080*"stock" + 0.034*"market" + 0.024*"investor" + 0.016*"buy" + 0.016*"look" + 0.013*"time" + 0.013*"good" + 0.013*"term" + 0.013*"investment" + 0.012*"see"
iTunes	0.038*"service" + 0.024*"business" + 0.016*"music" + 0.015*"also" + 0.013*"grow" + 0.013*"position" + 0.012*"retail" + 0.012*"pay" + 0.012*"store" + 0.011*"acquisition
US Government Tax	0.009*"government" + 0.009*"take" + 0.009*"case" + 0.009*"car" + 0.009*"work" + 0.008*"tech" + 0.008*"income" + 0.008*"business" + 0.007*"country" + 0.007*"tax"
Apple Technology(iCloud)	0.018*"user" + 0.015*"datum" + 0.014*"technology" + 0.014*"even" + 0.014*"call" + 0.011*"cloud" + 0.010*"platform" + 0.010*"internet" + 0.009*"information" + 0.009*"provide
Apple's earnings and Stock's Price	0.051*"earning" + 0.042*"quarter" + 0.031*"revenue" + 0.028*"report" + 0.028*"share" + 0.027*"growth" + 0.025*"expect" + 0.021*"estimate" + 0.017*"sale" + 0.016*"stock"
Sales and Trading	0.029*"high" + 0.024*"week" + 0.020*"market" + 0.020*"stock" + 0.019*"day" + 0.016*"low" + 0.016*"trade" + 0.015*"index" + 0.014*"close" + 0.013*"point"
iPhone	0.032*"new" + 0.023*"iphone" + 0.018*"device" + 0.017*"product" + 0.015*"phone" + 0.012*"watch" + 0.012*"also" + 0.012*"launch" + 0.011*"app" + 0.011*"feature"
Relative Stock Performance	0.058*"price" + 0.021*"percent" + 0.018*"rate" + 0.016*"value" + 0.016*"target" + 0.014*"stock" + 0.014*"share" + 0.014*"low" + 0.013*"current" + 0.013*"month"
Stock's Risk	0.012*"profit" + 0.010*"issue" + 0.009*"actually" + 0.008*"comment" + 0.007*"question" + 0.007*"take" + 0.007*"decision" + 0.007*"holding" + 0.007*"pattern" + 0.007*"never"
Apple's TV Subscription	0.028*"report" + 0.013*"tv" + 0.013*"watch" + 0.013*"game" + 0.012*"accord" + 0.012*"video" + 0.012*"deal" + 0.011*"show" + 0.010*"screen" + 0.010*"work"

Each document is fitted in the LDA model to obtain the respective probabilities of it falling into these ten topics. Thus, the documents are feature engineered to probability values to be used as input variables to the stock predicting models.

7.2 Integrating Google Trends data

In order to attempt to account for these residuals and further improve our model, our group incorporated Google Trends data in order to see if such information can help explain the variance observed between our predictions and what is actually observed better.

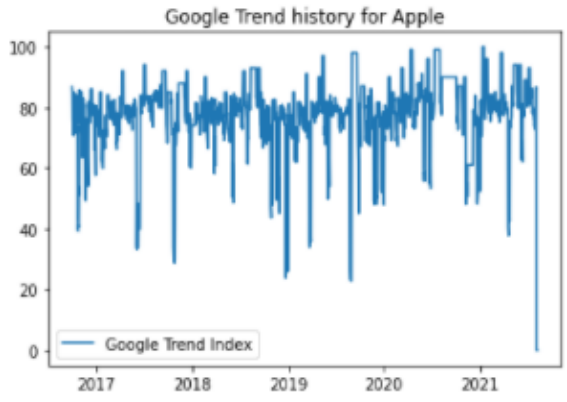
Conceptually we would expect that this variance could be caused by market factors such as investor sentiment or excitement that is occurring in real-time and cannot be explained using a time-series model. However, there is a possibility that such variance might be reflected in Google search keywords and patterns given that the Internet is widely used to look up and retrieve information - which is the basis for incorporating the Google Trends data.

Information pertaining to search queries are available in Trends. Trends is a website by Google that analyzes the popularity of top search queries in Google Search across various regions and languages. Graphs are shown to compare the search volume of different queries over time. Such data enables us to better understand the factors causing stock market price fluctuations, as the search volume of a query is an indicator of how interested the population of the world is in that topic at any given point in time.

Consequently, we expected some form of relationship between the Google Trends search volume of a company and its stock price at any point in time since there should be a surge in search queries during times of large price movement. We queried Google Trends API for the search volume of Apple over the past 5 years, from 1st October 2016 to 1st October 2021 to obtain this data. The data consists of the search volume for Apple indexed, with the maximum search volume being 100 and the rest are scaled accordingly.

	date	Apple
0	10/1/2016 0:00	94
1	10/1/2016 1:00	93
2	10/1/2016 2:00	90
3	10/1/2016 3:00	85
4	10/1/2016 4:00	78

Apple search volume between the period of 2016-2021 on an hourly basis



Historical Search Volume for Apple from 2016-2021

Since we are predicting prices on a daily basis, the data was processed such that each day has aggregated all the hourly readings by taking the mean values for the entire day instead.

7.3 Optimised [Categorical] Model for Predicting Movement of Closing Price

In Phase 2, adding Sentiment score did improve some of the models, especially categorical models. In this phase, we will separately add Topic Modelling and Google Trends to the phase 1 model to observe the effects of these new features.

Similar to Phase 2, each model after adding the new features, will be optimised with the same process as in Phase 1, using Bayes optimisation with 500 evaluations, maximising AUC score. Based on the results below, adding Topic Modelling and Google Trends separately to the model in Phase 1 had a positive impact on AUC score. In fact, all the models performed better. This suggests that Topic Modelling and Google Trend has an impact on stock price movement.

Optimised XGB	F1 score	AUC score
Best features from Phase 1	0.476	0.570
With Topic Modelling	0.693	**0.607**
With Google Trends	0.515	**0.578**

Additionally, we tested the effects of having both Topic Modelling and Google Trends together with the model in Phase 1 as well as adding Sentiment score on top of Topic Modelling and Google Trends. The results below show that a model with both Topic Modelling and Google Trends achieved a similar AUC score as a model that has only added Topic Modelling. This suggests that the Topic Modelling has captured some of the effects of Google Trends, hence adding both these features together has no difference from just Topic Modelling alone.

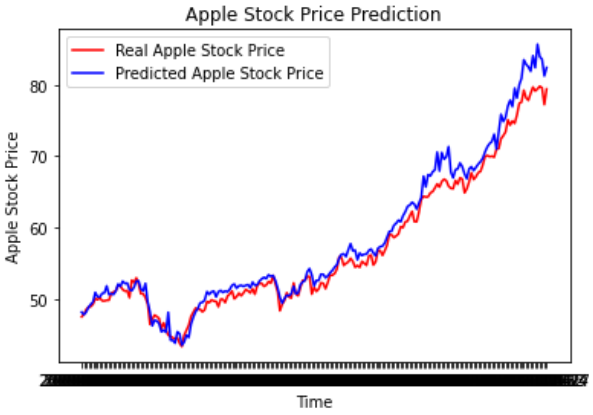
The effect of adding Sentiment score together with Topic Modelling and Google Trends was slightly worse than the model with just Topic Modelling and Google Trends (0.607 vs 0.603). This may be due to Sentiments and Topic Modelling having some correlation. For instance, certain topics by themselves could possibly possess a particular sentiment. As a result, this impacts the model's learning ability, resulting in a lower AUC score.

Optimised XGB	F1 score	AUC score
Best features from Phase 1	0.476	0.570
With Topic Modelling & Google Trends	0.694	**0.607**
With Sentiment, Topic Modelling & Google Trends	0.696	**0.603**

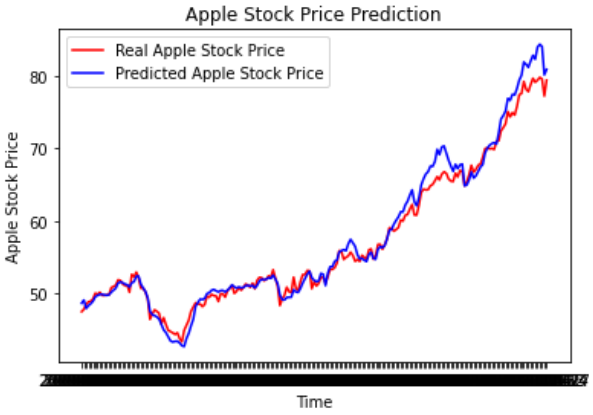
7.4 Optimised [Regression] Model for Predicting Closing Price

Likewise, Topic Modelling and Google Trends data were added as variables for training the regression models in an attempt to improve predictions. The results are shown below.

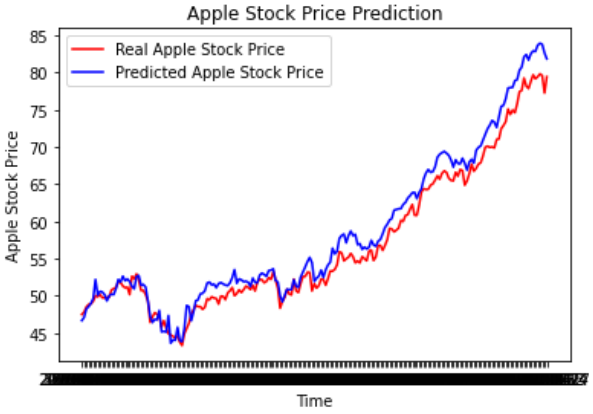
7.4.a Neural Network



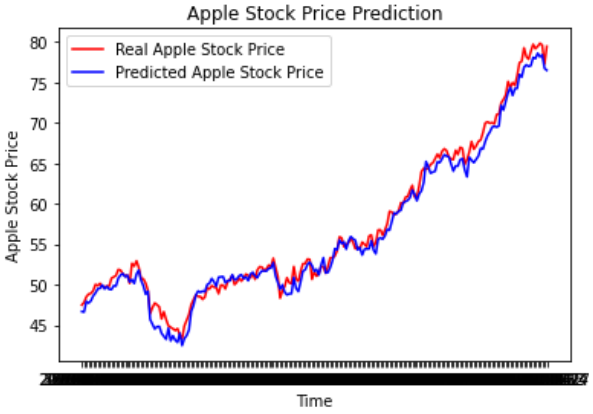
With Topic Modelling



With Google Trends



With Topic Modelling & Google Trends

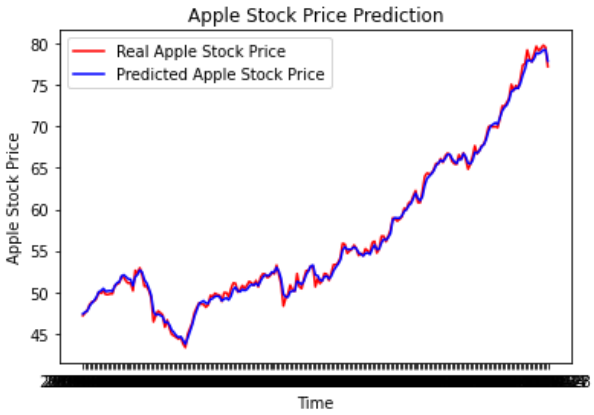


With Sentiment & Google Trends

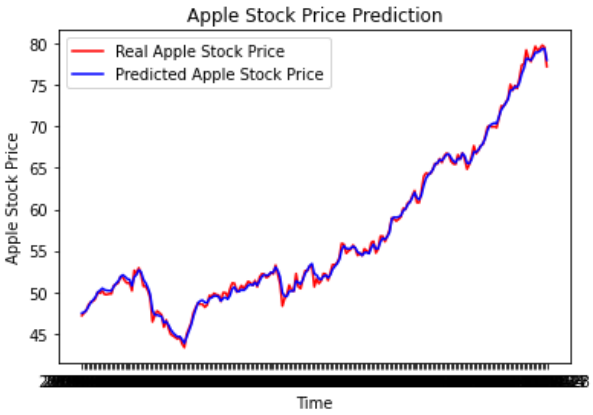
Optimised Neural Network	MSE	R^2	Adjusted R^2
Best features from Phase 1	0.60	0.993	0.993
With Topic Modelling	3.27	0.964	0.961
With Google Trends	2.02	0.978	0.977
With Topic Modelling & Google Trends	3.98	0.956	0.952
With Sentiment & Google Trends	1.34	0.985	0.984

By comparing the MSE, the original Neural Network model from Phase 1 has relatively the best model performance. However, it is insightful to observe that coupling the sentiment scores with Google Trends data improves the MSE in contrast to just using Google Trends alone. Previously from Phase 2, sentiment scores alone did not contribute to improving predictions. On the other hand, Topic Modelling generally gave poorer model performance in contrast to the XGB model with slight improvement in AUC.

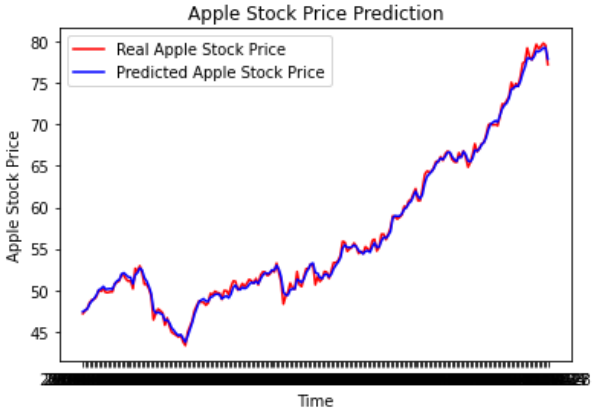
7.4.b Multiple Linear Regression



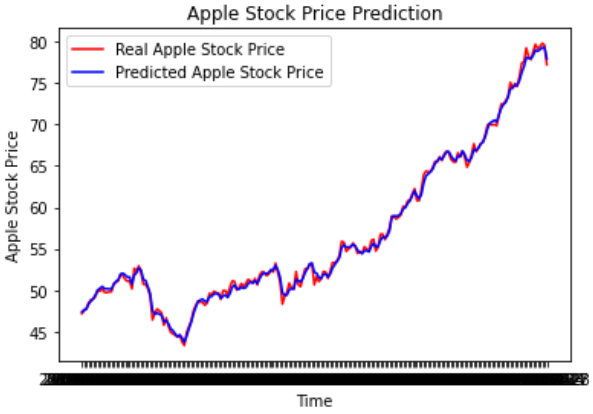
With Topic Modelling



With Google Trends



With Topic Modelling & Google Trends



With Sentiment, Topic Modelling & Google Trends

Optimised Linear Regression	MSE	R^2	Adjusted R^2
Best features from Phase 2	0.177	0.998	0.998
With Topic Modelling	0.186	0.998	0.998
With Google Trends	0.181	0.998	0.998
With Topic Modelling & Google Trends	0.187	0.998	0.988
With Sentiment & Google Trends	0.178	0.998	0.998

For the Linear Regression model, using EMA, MACD, RSI, Treasury yield, Volume achieves the lowest MSE while including Google Trends data produces very comparable results with MSE just off by 0.001. Similar to Neural Network, Topic Modelling does not seem to work well for regression models.

7.5 Optimised [Time Series] Model for Predicting Closing Price

After training the SARIMAX model, by incorporating the Google Trends data for Apple inside, we find that the p-value for the coefficient for the search volume 'Apple' is both small (0.0043) and insignificant (p = 0.406 > 0.05).

SARIMAX Results

Dep. Variable:	AAPL	No. Observations:	1007
Model:	SARIMAX(1, 1, 1)x(1, 1, 1, 14)	Log Likelihood	-1682.501
Date:	Mon, 15 Nov 2021	AIC	3377.003
Time:	01:29:33	BIC	3406.401
Sample:	0	HQIC	3388.180
	- 1007		
Covariance Type:	opg		

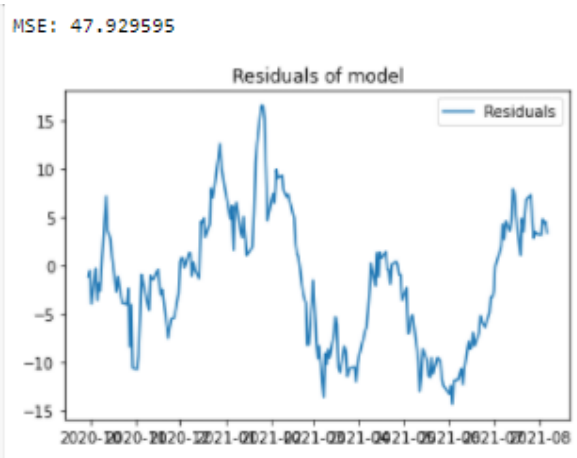
	coef	std err	z	P>|z|	[0.025	0.975]
Apple	0.0043	0.005	0.832	0.406	-0.006	0.014
ar.L1	-0.3743	0.105	-3.576	0.000	-0.579	-0.169
ma.L1	0.2645	0.110	2.410	0.016	0.049	0.480
ar.S.L14	0.0746	0.023	3.222	0.001	0.029	0.120
ma.S.L14	-0.9743	0.013	-76.874	0.000	-0.999	-0.949
sigma2	1.6729	0.028	58.725	0.000	1.617	1.729

Ljung-Box (L1) (Q):	0.00	Jarque-Bera (JB):	11501.16
Prob(Q):	0.98	Prob(JB):	0.00
Heteroskedasticity (H):	20.17	Skew:	-0.50
Prob(H) (two-sided):	0.00	Kurtosis:	19.65

Model Summary of SARIMAX model incorporating search volume as an exogenous variable

Unfortunately that implies that despite this additional source of information, that Google trends data does not aid us to predict stock price changes.

Model predictions compared to test set



MSE and residuals of new model

Overall, there is a slight increase in model performance on the test set. However, the MSE dropped to 47.929595. But due to the lack of significance of the exogenous variable, it cannot be concluded that the poorer prediction is due to the inclusion of Google trends data as a predictor.

7.6 [Text Classification] Model for Predicting Movement of Closing Price

Although a CNN model was barely able to predict movement of closing price at all in Phase 2, we continued examining the effects of adding Google Trends as a feature, since the XGB model showed that Google Trends as an added feature helps in improving AUC score. Similar to Phase 2, Sentiment score and Topic Modelling are excluded from the CNN model to extract textual information naturally and not overweight sentiment analysis or topic modelling effect.

Even with Google Trends added as an additional feature, the best CNN model still resulted in a F1 score of 0.550 and AUC score of 0.486, which is worse than a simple random guess model. Since this result is worse than Phase 2's CNN result, we conclude that CNN as a classifier model does not work well for predicting closing price movement.

8. Phase 4 - Comparing across Industries

After consolidating technical indicators to be used for each model in Phase 1, investigating the relationship of sentiment scores on stock prices on Phase 2, and incorporating Topic Modelling and Google Trends in Phase 3, the best performing and most promising models will be used to further compare predictions across industries. As mentioned, Apple, P&G and Facebook are selected as our companies as they are situated along the spectrum when moving across from Consumer-based to Information Technology.

For categorical models, XGB performs the best with the highest AUC score. For regression models, Linear Regression achieves the lowest MSE. SARIMAX and CNN models were not used in Phase 4 due to their comparatively poor model performance.

Overall, Google Trends had little to no improvements in most of the model predictions. This could be because Google Trends data could have correlated with sentiment and other trading indicators such as Volume of trades and as such it provided little value add. Therefore, only sentiment scores and Topic Modelling will be used in testing across industries in Phase 4.

8.1 Optimised [Categorical] Model for Predicting Movement of Closing Price

From the previous phases, Sentiment Analysis and Topic Modelling, both of which comes from analysis of news, had a positive impact on the AUC score for Apple. Hence, in this section, we will see if similar impact can be observed on different industries.

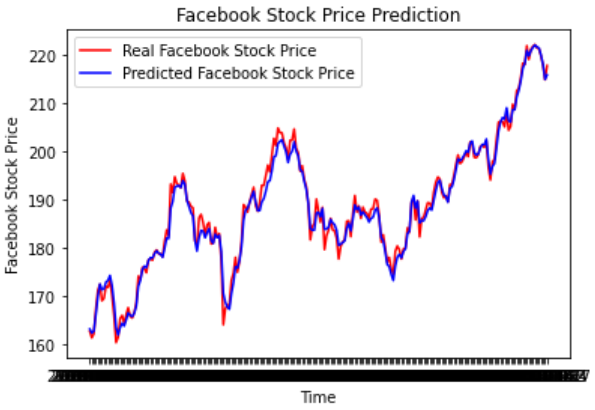
For both FB and PG, representing Information Technology (IT) and Consumer Product industry respectively, adding on Sentiment score with the best features selected from Phase 1 did not actually worsen the prediction performance instead of improving it. In contrast, AAPL, which consists of a mix of both IT and Consumer Product, experienced a higher AUC score after adding Sentiment score. This suggests that the effect of news sentiments might be more company specific, rather than being industry related.

In our study of the effect of Topic Modelling, PG experienced an increase in AUC score, while FB's AUC score worsened. PG's behavior on the addition of Topic Modelling is more in line with AAPL's behavior. This may suggest that news Topic Modelling has a positive impact on firms price movement in the Consumer Product industry. On the contrary, in the IT industry, news Topic Modelling is not a good feature in predicting closing price movement. AAPL, being a mix of both industries, will likely have conflicting impacts as well. However, since AAPL is more aligned as first a Consumer Product firm among the eyes of the public, it is likely to experience the same effect as PG. As such, this explains why news Topic Modelling has a positive impact on AAPL, despite it being partially an IT firm as well.

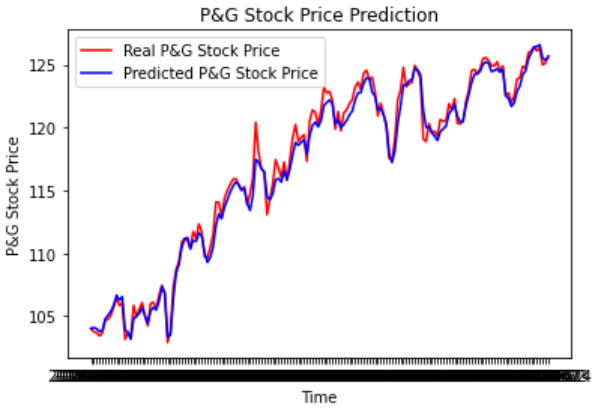
Industry Type	***Information Technology (IT)***		***Mix of IT & Consumer Product***		***Consumer Product***	
Optimised XGB	**FB**		**AAPL**		**PG**	
	F1 score	AUC score	F1 score	AUC score	F1 score	AUC score
Best features from Phase 1	0.685	**0.617**	0.476	0.570	0.675	0.593
With Sentiments	0.673	0.588	0.605	0.584	0.675	0.571
With Topic Modelling	0.639	0.567	0.693	**0.607**	0.675	**0.623**

8.2 Optimised [Regression] Model for Predicting Movement of Closing Price

Phases 1 to 3 were also repeated for the regression models used to predict FB and PG. The predictions of the best-performing model with their parameters are shown below.



EMA (10,20,40) + MACD + RSI



EMA (10,20,40) + MACD + RSI with Topic Modelling

Industry Type	***Information Technology (IT)***			***Mix of IT & Consumer Product***			***Consumer Product***		
Multiple Linear Regression	**FB**			**AAPL**			**PG**		
	MSE	R^2	(Adj) R^2	MSE	R^2	(Adj) R^2	MSE	R^2	(Adj) R^2
Best features from Phase 1	**2.256**	0.988	0.987	0.181	0.998	0.998	0.512	0.990	0.989
With Sentiments	2.276	0.988	0.987	**0.177**	0.998	0.998	0.512	0.990	0.989
With Topic Modelling	2.301	0.988	0.987	0.186	0.998	0.998	**0.490**	0.990	0.989

From the table above, it can be seen that there are varying impacts across the 3 companies. Similar to the categorical model, predictions for FB achieved the lowest MSE when sentiment scores and Topic Modelling are not included. Thus, there is a possibility that the stock prices of IT firms fluctuate easily, and already reflect sentiment scores and topic modelling. As a result, the inclusion of the latter 2 variables does not help improve predictions.

On the other hand, including Topic Modelling as a feature improved the MSE for PG. This might suggest that certains topics are of interest to investors and are useful in increasing prediction accuracy for the Consumer Product Industry. The use of sentiment scores in improving model predictions is also limited as evident from increase in MSE for FB, marginal decrease in MSE for AAPL, and constant MSE for PG. It is also interesting to note that TNX and Volume data did not help to improve predictions for both FB and PG but helped with AAPL.

9. **Model Insights & Analysis**

9.1 Categorical vs Regression Models

The 2 aforementioned models - XGB (categorical) measuring movement of closing price and Neural Network & Linear Regression (regression) measuring the Closing Price seemed to have vastly different levels of model performance, with the former only having a high AUC of around 0.6 while the latter easily having R-Squared of over 0.9. This seems to be contradictory as the categorical models only attempt to predict direction whereas the regression models have to predict both direction and magnitude, making the latter significantly more challenging.

Nonetheless, the MSE of the regression models explains why their performance may not be as good as the high R-Squared values suggest. At each of their best performing levels, Neural Network and Linear Regression still achieve a MSE of 0.60 and 0.177 respectively. Consequently, this results in a root mean squared error of around 0.77 and 0.42 respectively. Hence, on average, the predictions of both regression models are still off by over $0.40 in magnitude without taking into account the direction. Considering this with the magnitude of average stock price movement within a day, which usually ranges uptill $0.10 (in fact, the average price movement of AAPL for each trading day within the last 5 years was $0.0594), the MSE of the regression models indicates that it cannot accurately predict price movements. This aligns with the 0.5-0.6 AUC that is shown by the categorical models.

9.2 COVID as a Black Swan event

One particular insight we gleaned from predicting stock prices was that for the period of 2020 to present day, the models performed exceptionally poorly compared to what we initially expected. This was confirmed through backdating our same models and rerunning them over a similar time horizon to predict prices beforehand. Our group hypothesised that the cause of this drop in model performance could be intrinsically tied to the underlying situation of COVID-19.

A black swan event is an event that cannot be predicted due to it being beyond what is normally expected of a situation and has potentially severe consequences. Since they are characterised as being extremely rare that one cannot quantify the probability of such an event, ordinary statistical methods such as time series modelling or other forecasting methods that depend on historical data, cannot be used to account for such events.

For example, in the late 1990s, Long-Term Capital Management (LTCM) was a large hedge fund that collapsed due to the Russian Government's debt default. Such a black swan event could not be predicted despite the numerous models that were built by the Nobel Prize-winning economists and industry knowledge of renowned Wall Street traders heading the hedge fund. In a similar vein, COVID-19 is an event that causes our model predictions for the period of 2020 to 2021 to be out of calibration with the reality of the situation, as its effect is so large that it cannot be fully accounted for by our models.

The performance of the regression models could not have been achieved had it been trained on pre-Covid market numbers and then be used to predict for the market affected by Covid. In fact, the MSE of most models predicting in the Covid market would sit at above 5, effectively indicating the predictions are too inaccurate and practically unusable. This highlights the robustness of our models that performs better in a relatively stable climate (pre-Covid as marker was generally upward trending) but is unable to adapt its predictions to such black swan events. It is also debatable whether including past market crashes such as the financial crisis of 2008 or the 2000 Dot Com Bubble could have helped the regression models predict better. As these events happened more than 10 or 20 years ago, the market may also have not reacted similarly and may otherwise skew predictions in a more normal market.

9.3 Predicting Tomorrow's Closing Price

The models were initially trained on predicting today's "Close" using today's market data. However, the idea was fundamentally flawed as some of the market data may already directly or indirectly contain the Closing Price. The moving average for example, will have already taken into account today's Closing Price, and including moving averages, especially EMA which gives more weight to the most recent data, can significantly increase the performance of such models. The above reasons could have contributed to the classifiers generating a good AUC of 0.7 and higher before adjusting to predict tomorrow's "Close".

Predicting tomorrow's "Close" based on today's data is a more viable and practical approach that has better real-world use applications such as being applied in actual stock market price predictions. Next, we will summarise the technical indicators to use, effectiveness of incorporating sentiment analysis, topic modelling, Google Trends data and how the models performed across industries.

9.4 Model Evaluations

Prediction	Models Used
Movement (categorical)	Extreme Gradient Boost (XGB) Logistic Regression
Closing Price (regression)	Neural Network Multivariate Long Short Term Memory (LSTM) Linear Regression
Time Trends & Seasonality	Seasonal Auto-Regressive Integrated Moving Average with eXogenous factors (SARIMAX)
Text Classification	Convolutional Neural Network (CNN)

From the previous sections, we have only evaluated each model based on their performances, with the assumption that these models are suitable for the prediction problem. However, in reality, a model's suitability depends on the context and purpose for its usage. For instance, focusing on Closing Price prediction, among Neural Network, Multivariate LSTM, and Linear Regression, although they have varying performances, one important key factor in selecting a model is the interpretability of factors and its effect on performance. In this aspect, a Linear Regression model may be preferred over Neural Network and Multivariate LSTM, due to its ease of interpretability. Both Neural Network and Multivariate LSTM are known as black box models, in which interpretability is almost impossible. Interpretability is important as users would want to know what causes a certain effect and the impact of it. This is especially crucial in finance as traders and investors are not only interested in accuracy of prediction but also factors contributing to the prediction. For instance, if the model can inform the trader that a certain combination of events will lead to a certain movement of an asset, then the trader can maximise return by placing the trades earlier when he/she expects the combination of events to occur soon. This exemplifies the importance of model interpretability. This is similar for the closing price movement prediction models, where Logistic Regression is preferred over XGB for interpretability.

Another perspective in evaluating models would be to use Occam's razor principle, where a simple model is preferred over a more complex one. For instance, in comparing between 2 models of comparable performance, the simpler model would be preferred in this case. This principle was applicable for the SARIMAX model. Despite the expectation that it would fit our stock ticker well due to being designed to work on time series, that model forecasted close to a straight line with little variation, and only a small seasonal effect, which was due to the stepwise algorithm choosing a fairly simple model instead of more complicated ones. It is entirely possible that there exists a highly complex SARIMA model that would forecast Apple's stock price well, but it would be difficult to find and the model might be overfitting which the loss function in our evaluation penalized, resulting in the simpler, but unfortunately a poorer performing model being chosen in the end.

Thirdly, due to the fast moving and changing nature of the financial markets, a good model should be one that is robust to changes or retrains quickly in reacting to changes in the environment. Earlier in section 9.2, we mentioned how COVID as a Black Swan event affects the robustness of the models. In such cases, it is indeed difficult to design and train a model that is robust enough. One possible method would be to create synthetic data to model these events to 'shock' the model. However, even so, such measures may still be insufficient. As such, an alternative would be to choose a model which can be retrained quickly, while maintaining prediction capabilities. In this scenario, a model with a simpler optimization problem is preferred such as regression models as compared to Neural Network or XGB.

Ergo, while this report mainly evaluates each model based on its performance in relation to the added features, an important evaluation to consider before deploying for practical use would be the context and purpose that this model is intended to fulfill.

9.5 Technical Indicators to Use

To reiterate, the technical indicators used were categorised into Trend (SMA & EMA), Signal (MACD & RSI), and others (TNX & Volume). For Trend indicators, from the models we ran, we discovered that EMA generally works better for simpler models such as Linear Regression and Logistic Regression, while SMA is preferred for more sophisticated models such as XGB and multivariate LSTM. This could be due to the fact that more recent price data gives a stronger trend indication for simpler models, but as XGB and multivariate LSTM involves deeper learning, the exponentially weighted prices might have skewed the learning process.

On a whole, training with Signal indicators generally gives rather significant improvements to the model for both categorical (in terms of AUC) and regression models (in terms of MSE). This is reasonable as Signal indicators are often used to gauge momentum in the market and are often combined with Trend indicators to give a complete technical view of the stock market. Also, the improvements from TNX & Volume indicators are more significant in regression models than categorical models. Nonetheless, this seemed to only have applied to AAPL, but not FB or PG.

There are also other popular indicators such as Fibonacci retracement & extension tools, Bollinger bands, self-dictated trading support/resistance levels, etc. that were difficult to be included in the models, but possess the potential to improve the model prediction performances.

9.6 Sentiment Analysis Insights

The model performance after incorporating sentiment scores exhibits the limitation in assuming stock market is affected by the news, at least for the regression models. Incorporating sentiment scores generated using NLTK did help improve the AUC of the XGB model by 0.014. However, Neural Network's MSE increased from 0.60 to 0.83 and Multivariate LSTM's MSE increased from 5.72 to 14.05 while Linear Regression and SARIMAX's MSE remained almost the same. Therefore, the sentiment scores of today had significantly less predictive power and indication on the price movement & closing price of tomorrow than we have expected.

This could be due to the general nature of news being more reactive to stock market movements as compared to providing deep and impactful insights, that can significantly affect prices. The effects of such sentiment could also have been captured in the price of the stock, thus leaving little potential for the sentiment scores to contribute to the model. Nevertheless, the news headlines and articles were still acquired from reputable sources such as Market Insider, CNBC news, the Kaggle dataset which was in turn collated from investing.com.

9.7 Potential correlation between Google Trends and market data

Google Trends were included to capture more information about the market beyond technical analysis of stock prices. However, this did not manage to bring the intended effect of improving the model as compared to Topic Modelling. Nonetheless, it is also worth noting that the Google

Trends data used were only on the name of the company. Hence, the Google Trends data could have been similar to sentiment or Volume of trades, and there was not much further improvement to predictions that could be derived from including it in the model.

9.8 Model Performance across Industries

Between FB, AAPL, and PG, there were observable differences in model performance. Firstly, TNX and Volume data does not appear to improve predictions for FB and PG, even for the regression models which previously improved predictions for AAPL. This could potentially be attributed to AAPL having the second-largest market cap of all companies in the world (behind MSFT) with FB and PG ranked 7th and 21st respectively as of Nov 21. AAPL is therefore easily included in many market indexes, ETFs or funds which allows it to react to more general market data such as the TNX which in some ways reflects interest and inflation rates expectations.

Secondly, Topic Modelling appears to increase model performance for PG but not for FB across both categorical and regression models. This was seen by the increase in AUC from 0.571 to 0.623 in the XGB model and a reduction in MSE from 0.512 to 0.490 in the Linear Regression model. On the other hand, FB has the best model performance without including sentiment or Topic Modelling for both categorical and regression models. This could be due to the stock price of FB having faster market reaction and efficiently reflecting public information available (Semi-Strong Form Efficiency - Efficient Market Hypothesis) and as such the prices have accounted for the effects of news sentiments and Topic Modelling.

10. Conclusion

10.1 Process

In conclusion, extensive model testing was conducted to build a base model based on a combination of technical indicators (Phase 1). This was then attempted to be enhanced with sentiment analysis (Phase 2), Topic Modelling and Google Trends (Phase 3). The model was then tested across industries (Phase 4).

In exploring the past 5 years of stock prices, we introduced an extensive dataset that consists of daily economic and sentiment data. We utilised state-of-the-art tools for Natural Language Processing and corporated concepts of Topic Modelling and Google Trends to identify trends of volatility in the market. This was done using categorical, regression, time-trends, and text-classification models to various degrees of success. In the process, we covered the technical indicators to use, the significance (or the lack of) when including sentiment analysis, topic modelling and Google Trends data, impact of Covid-19 on model performance as well as pros & cons of each model and how predictions fared across industries.

10.2 Outcome

For the models tested for this project, XGB performs the best for categorical predictions while Linear Regression achieves the best results amongst the regression models. SARIMAX and CNN models did not give relatively good model performance. The robustness of the models were also found to be heavily dependent on whether Covid-19 data was included in training the models, especially for regression models.

When measuring price predictions at a one-day interval, we found that including Technical Indicators improves the model predictions significantly - specifically, EMAs to be preferred for simpler models while SMAs works better for models with deeper learning; MACD & RSI was also good Signal indicators to be used in conjunction with the Moving Averages; the Treasury yield seems to have only benefited the regression models and larger market-cap stocks.

The sentiment scores of news headlines from Market Insider, CNBC news and Investing.com (through the Kaggle dataset) were shown to be only marginally useful in improving predictions. This is contrary to our previous assumptions that the market moves on the news, and goes to show that the market is potentially very efficient in pricing in sentiments.

Incorporating Topic Modelling has shown potential to improve stock price predictions, although relatively more for PG and less for FB. Thus, there is a possibility that the effectiveness depends on the market capitalisation of companies, although this is to be studied further. On the other hand, the historical search volume of the company has shown lower effectiveness in improving price predictions as they are likely to be correlated with other market data such as sentiment scores or volume of trades and may have already been priced into the stock prices as suggested by the Efficient Market Hypothesis.

10.3 Further Areas to Explore

Further research is still needed to uncover more relationships between stock prices and economic, sentiment, and financial data over different time periods as well as for more companies. An interesting extension of our study would be to explore predictive power of the technical indicators, sentiment scores and topic modelling at higher time intervals, at the intraday level - to observe even greater real-world application use for real-time trading. Another extension could also be to explore investment sentiment by scraping web data from community websites such as Reddit or StockTwits to capture investor sentiments in addition to the current news sentiments already used.

References

1. Deng, J. Stock Prediction by Analyzing Financial News Sentiment and Investor Mood of Social Media. 2020. UC Santa Cruz. ProQuest ID: Deng_ucsc_0036N_12177. Merritt ID: ark:/13030/m5kq3kkp. Retrieved from https://escholarship.org/uc/item/5bq0c1kx

2. Joshi K, H. N B, Rao J. Stock Trend Prediction Using News Sentiment Analysis. International Journal of Computer Science and Information Technology. 2016;8(3):67-76. doi:10.5121/ijcsit.2016.8306

3. Mehta P, Pandya S, Kotecha K. Harvesting social media sentiment analysis to enhance stock market prediction using deep learning. PeerJ Computer Science. 2021;7:e476. doi:10.7717/peerj-cs.476

4. Kala S. Stock Market Prediction using News Sentiments. Medium. https://medium.com/@kala.shagun/stock-market-prediction-using-news-sentiments-f9101e5ee1f4. Published 2020. Accessed November 18, 2021.

5. Historical financial news archive. Kaggle.com. https://www.kaggle.com/gennadiyr/us-equities-news-data. Published 2019. Accessed November 18, 2021.

6. xgboost/demo at master · dmlc/xgboost. GitHub. https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions. Published 2017. Accessed November 18, 2021.

7. Ranjan A. Advantages and Disadvantages of Logistic Regression - GeeksforGeeks. GeeksforGeeks. https://www.geeksforgeeks.org/advantages-and-disadvantages-of-logistic-regression/. Published 2020. Accessed November 18, 2021.

8. Zucchi K. Why the 10-Year U.S. Treasury Yield Matters. Investopedia. https://www.investopedia.com/articles/investing/100814/why-10-year-us-treasury-rates-matter.asp. Published 2021. Accessed November 18, 2021.

9. Bhattacharya S. Top 7 Python NLP Libraries and Their Applications in 2021. Analyticsinsight.net. https://www.analyticsinsight.net/top-7-python-nlp-libraries-and-their-applications-in-2021/. Published 2021. Accessed November 18, 2021.

10. Nair G. Text Mining 101: Topic Modeling - KDnuggets. KDnuggets. https://www.kdnuggets.com/2016/07/text-mining-101-topic-modeling.html. Published 2016. Accessed November 18, 2021.

11. Czakon J. F1 Score vs ROC AUC vs Accuracy vs PR AUC: Which Evaluation Metric Should You Choose? - neptune.ai. neptune.ai. https://neptune.ai/blog/f1-score-accuracy-roc-auc-pr-auc. Published 2021. Accessed November 18, 2021.

Project Contribution

Name	Contributions
Ang Guo Xiong	*Regression models*: Neural Network, Multivariate LSTM models, Linear Regression (5.2, 6.2, 7.4, 8.2), 9.0 Insights Analysis, 10.0 Conclusion
Ching Zheng Ing	1.0 Introduction, 3.0 Stocks and Model Used, 4.0 Methodology, Script, Slides
Darius Seah Kuan Wei	2.0 Data Preparation, 7.1 Topic Modelling Web scraping of news headlines & Sentiment, Proofreading
Lim Huai Xing	*Categorical models*: XGB, Logistic Regression (5.1, 6.1, 7.3, 8.1) *Text classification models*: CNN (6.4, 7.6)
Leung Hoi Kit Alvin	*Time-series model*: SARIMAX model, (6.3, 7.5) Web scraping of Google Trends data (7.2), Video

The project was mainly split by models, with each of us focusing on training and tuning separate models, and then subsequently combined together for model comparisons. The other parts of the reports were not clearly split, and that everyone had significant inputs to the report.