

# STOCK PRICE PREDICTION USING SENTIMENT ANALYSIS

## Hrishita Bapuram(22204557), Soujanya Hassan Prabhakar(22202225)

### INTRODUCTION: REDDIT and ALTERNATIVE DATA in STOCK PERFORMANCE ANALYSIS

- In recent times, Reddit has surfaced as a favored platform for engaging in conversations and deliberations spanning a broad array of subjects, which encompass financial matters and investments.
- With a multitude of users sharing their viewpoints and understandings concerning diverse stocks and financial markets, Reddit has transformed into a valuable information pool for gauging sentiments within the financial domain.
- Additionally, with the emergence of Alternative Data as a growing source to evaluate Market Performance, studying the impact of market sentiment could add for an invaluable measure while studying possible vulnerabilities of a stock.
- This project delves into the application of sentiment analysis for assessing the fluctuations of widely-held stocks, employing Reddit headlines and comments as a wellspring of market sentiment.
- This project aims to scrutinize pre-existing studies that leverage sentiment analysis derived from Reddit to approximate market volatility, and consequently, future gains.

### KEY OBJECTIVES

- Assessing Market Movement through Sentiment Analysis:** To investigate the feasibility and effectiveness of utilizing sentiment analysis on Reddit headlines and comments to evaluate the correlation between public sentiment and the movement of popular stocks.
- Reviewing Existing Reddit Sentiment Analysis Research:** To conduct a comprehensive review and synthesis of existing research studies that employ sentiment analysis derived from Reddit to predict market volatility, estimate future stock returns, and assess the reliability of these approaches.
- Comparing Social Media vs. Traditional Analysis:** To perform a comparative analysis that assesses the relative effectiveness and predictive capabilities of social media sentiment analysis in contrast to conventional methods of market analysis, such as technical and fundamental analysis.
- Sentiment Analysis Methodology:** To implement a sentiment analysis methodology that computes the sentiment polarity of public discussions on a given day within the studied time frame, leveraging data extracted from the selected social media platform (Reddit).
- Pattern Analysis for the Sentiment-Stock Relationship:** To systematically examine recurrent patterns in the connection between market sentiment outcomes and stock price movements, investigating whether specific sentiment patterns correlate with stock price highs and lows.
- Develop a predictive model for Stock Returns/Prices:** To design and construct a predictive model, preferably utilizing a Recurrent Neural Network (RNN), that combines sentiment data from social media with historical stock prices to forecast future stock returns or prices within the chosen time frame.

### SENTIMENT ANALYSIS IN FINANCIAL MARKETS

- Analysis of Stock market returns has been an eon-old field of research, with multiple approaches evolving over time to accurately guess market returns.
- Some of these include Fundamental and Technical Analysis, Time Series, and Forecasting using traditional methods and machine learning approaches.
- The use of Sentiment Analysis is a relatively new approach that uses unstructured textual data to extract public opinion.
- Sentiment and Lexicon-based analyses thus provide a novel, more direct approach to obtaining data that truly represents individual opinions.
- In light of this, there have been many case studies of use cases as well as public data repositories on platforms like Kaggle and GitHub that have made clean and well-organized data more accessible for everyone to contribute to creating a more sophisticated model.
- The idea vests in the understanding that public opinion drives market dynamics and a "negative" or "positive" sentiment could trigger a bearish or a bullish trend, as illustrated in the graphic below:

### DATA SNAPSHOT

SELECTED COMPANIES	DATE RANGE	RESPONSE VARIABLE
APPLE INC. (AAPL) GOOGLE (GOOG) MICROSOFT (MSFT) NVIDIA (NVDA) TESLA (TSLA)	TRAIN: 01/03/23 TO 09/06/23 TEST: 10/06/23 TO 01/07/23	ADJ_CLOSE PRICE AT TIME T+1
FINANCIAL PREDICTORS		
OPEN, HIGH, LOW, VOLUME		
SENTIMENT PREDICTORS	TEXT DATA STRUCTURE (POST PREPROCESSING & AGG)	
WEIGHTED SUBJECTIVITY WEIGHTED COMPOUND SCORE	DATE	SCORE
	W_SUBJ	W_COMP
FINANCIAL DATA STRUCTURE		
	DATE	OPEN
	HIGH	LOW
	ADJ_CLOSE	VOLUME

### REDDIT DATA PROCESSING

EXTRACTED FILES FROM REDDIT USING TICKER NAME and WWDIC

APPLE INC. (AAPL)

GOOGLE (GOOG)

MICROSOFT (MSFT)

NVIDIA (NVDA)

TESLA (TSLA)

TRAIN: 01/03/23 TO 09/06/23

TEST: 10/06/23 TO 01/07/23

FINANCIAL PREDICTORS

OPEN, HIGH, LOW, VOLUME

SENTIMENT PREDICTORS

WEIGHTED SUBJECTIVITY  
WEIGHTED COMPOUND SCORE

TEXT DATA STRUCTURE (POST PREPROCESSING & AGG)

DATE	SCORE	W_SUBJ	W_COMP
2022-12-04	0.000000	0.000000	0.000000
2022-11-24	0.314167	-0.0258	0.000000
2022-09-13	0.000000	0.2500	0.000000

LEMATIZATION

WORDNETLEMMATIZER() USING NLTK

PREPROCESSING STEPS preprocess\_text() function

CHECK IF 'TEXT' IS A NON-NULL STRING

REMOVE PUNCTUATION

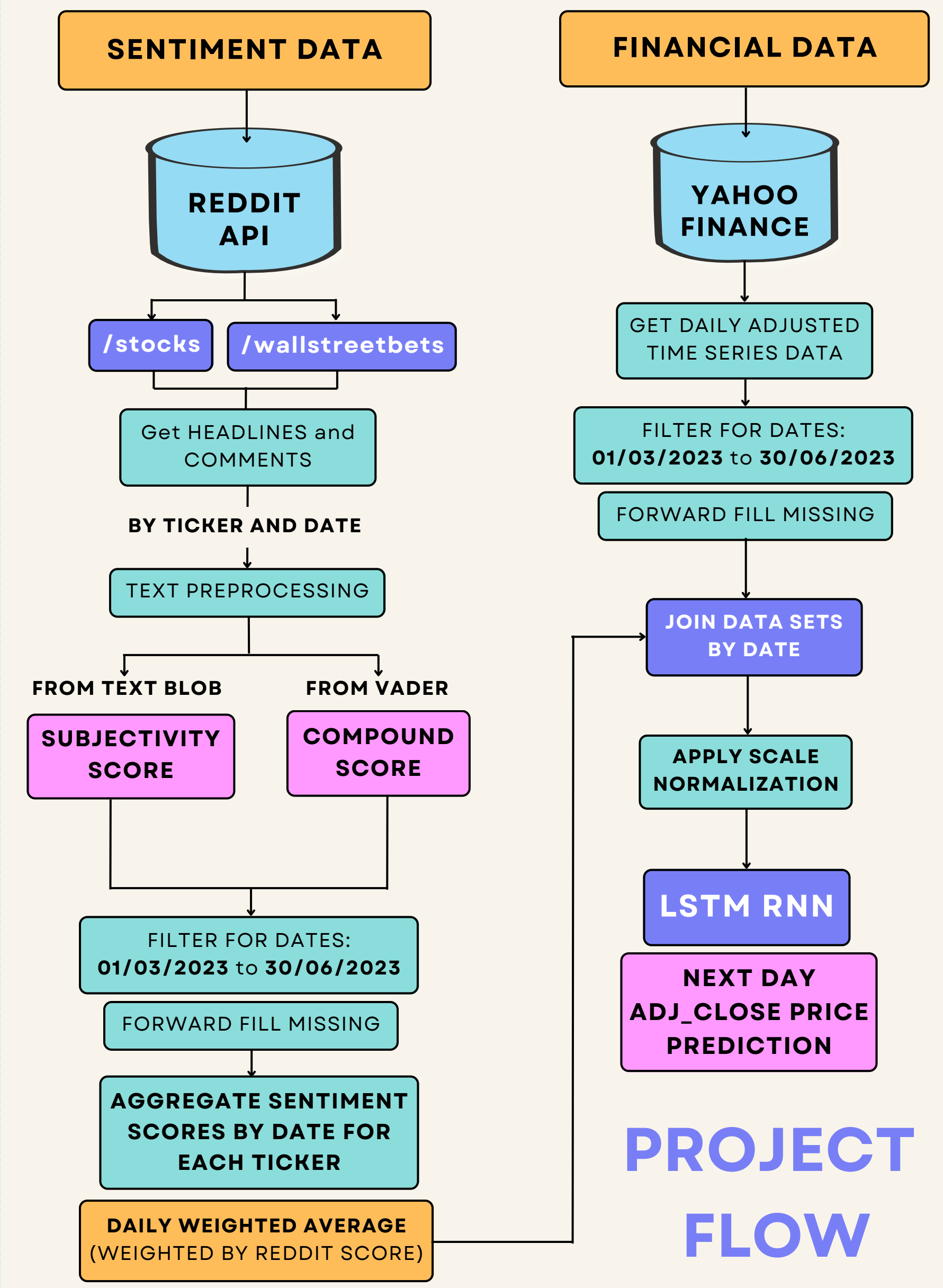
CONVERT TO LOWERCASE

TOKENIZE TEXT

REMOVE STOPWORDS

FILTER OUT SHORT WORDS

JOIN TOKENS BACK INTO A STRING



### SENTIMENT ANALYSIS

SENTIMENT ANALYSIS

```
#Create a function to get the subjectivity
def getSubjectivity(text):
    return TextBlob(text).sentiment.subjectivity

#Create a function to get the sentiment scores
def getSIA(text):
    sia = SentimentIntensityAnalyzer()
    sentiment = sia.polarity_scores(text)
    return sentiment

SUBJECTIVITY FROM TEXTBLOB
COMPOUND SCORE FROM SIA (VADER)
```

Compound Score by Date

Subjectivity Score by Date

date	score	Subjectivity	Compound
0	2023-06-06	6842	0.600000
1	2023-06-06	5802	0.000000
2	2023-05-09	5734	0.314167
3	2023-03-03	4954	0.000000

date	score	w_subj	w_comp
0	2023-03-01	3	0.166667
1	2023-03-02	3600	0.155467
2	2023-03-03	8677	0.185994
3	2023-03-04	4358	0.391205

Sentiment measures do not have a sequential dependence

Most text samples are subjective in nature

w_comp	w_subj	volume	open	high	low	
0	-0.168600	0.166667	55479000.0	146.830002	147.229996	145.009995
1	-0.355207	0.155467	52238100.0	144.380005	146.710007	143.899994
2	0.186974	0.185994	70732300.0	148.039993	151.110001	147.330002

RMSE	2.9831
MAE	2.5370
MAPE	0.0135

Train Loss	0.0022
Test Loss	0.0149

ADAM OPTIMIZER (LR = 0.01)

EPOCHS = 60

EARLY STOPPING: PATIENCE = 15

REDUCE LR ON PLATEAU: PATIENCE = 15

Layer (type)	Output Shape	Param #
lstm_1 (LSTM)	(None, 64)	16896
dropout_1 (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 1)	65

Total params: 16,961

Trainable params: 16,961

Non-trainable params: 0

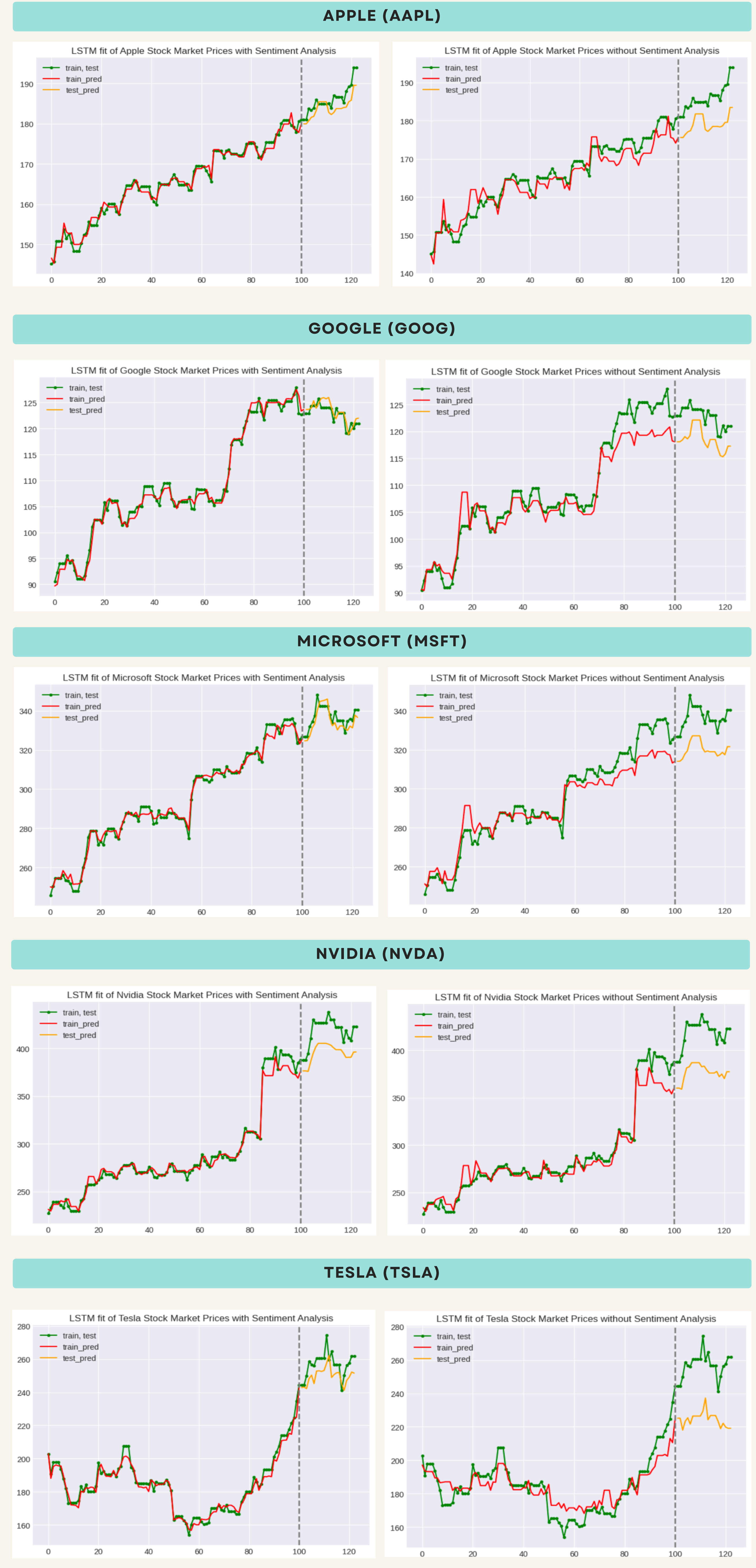
VALIDATION SPLIT = 0.2

### MODEL EVALUATION (RMSE, MAE, MAPE, TEST & VALIDATION LOSS)

Company	Metrics	With Sentiment Metrics	Without Sentiment Metrics
APPLE	RMSE	2.9831	7.4547
	MAE	2.5370	7.0735
	MAPE	0.0135	0.0379
GOOGLE	RMSE	1.2020	4.5130
	MAE	1.0026	4.2552
	MAPE	0.0082	0.0346
MICROSOFT	RMSE	3.7656	16.6647
	MAE	3.4598	16.4653
	MAPE	0.0102	0.0488
NVIDIA	RMSE	23.7111	42.1509
	MAE	22.8826	41.5345
	MAPE	0.0544	0.0989
TESLA	RMSE	8.5815	33.1223
	MAE	7.6951	32.3034
	MAPE	0.0297	0.1252

Company	Metrics	With Sentiment Metrics	Without Sentiment Metrics
APPLE	Validation Loss	0.0022	0.0138
	Test Loss	0.0149	0.0931
GOOGLE	Validation Loss	0.0027	0.0266
	Test Loss	0.0055	0.0582
MICROSOFT	Validation Loss	0.0025	0.0216
	Test Loss	0.0054	0.1060
NVIDIA	Validation Loss	0.0038	0.0125
	Test Loss	0.0504	0.1594
TESLA	Validation Loss	0.0025	0.0191
	Test Loss	0.0202	0.3012

### COMPANY-WISE RESULTS OF FIT WITH AND WITHOUT THE INCLUSION OF SENTIMENT SCORES (TRAIN AND TEST)



### RESULTS AND CONCLUSIONS

- The results obtained upon the implementation of the prediction algorithm for Adjusted Closing Price, we notice that the inclusion of the Sentiment Parameters makes a positive impact on improving performance
- The model fit didn't require a lot of hyperparameter tuning, with only minor adjustments made to the number of input units, number of epochs, and dropout rate.
- The sentiment analysis scores weighted by Reddit Score (Given as Number of Upvotes - Number of Downvotes) as the weight component in calculating the aggregate sentiment score for each day accounted for the level of interaction on a particular comment/headline thereby evaluating the average sentiment for each day appropriately in accordance with the "influence" of the particular text.
- Sentiment features, i.e., Subjectivity Score and Compound score prove to be particularly impactful in predicting future values where there have been huge spikes/dips (as observed in behaviour of Microsoft, Nvidia, and Tesla) since the improvement in the evaluation measures of Root Mean Squared Error, Mean Absolute Error and Mean Absolute Percentage Error has been greater in these cases than the rest.
- As an extended study, evaluating the data and analysing the unpredictable highs and lows might facilitate for the creation of a simulation to generate worst-case scenarios for companies to use while conducting stress tests or developing Business Continuity Plans.