Comparative Analysis of Stock Price Prediction Models with Twitter Text Data

Andrew Krumenacker Rafiz Sadique Evan Qiang
University of Pennsylvania

Project Overview and Data Preprocessing

Datasets

Training: 40,000 Tesla-related tweets from Kaggle (2015-2018)

Testing 1: 10,000 Tesla-related tweets from Kaggle (2019)

Testing 2: 10,000 Tesla-related tweets from May 2024 (using Twitter API)

Goals

Direct Prediction from Raw Text: Develop a model that directly uses raw Twitter data to predict Tesla's stock price movements, bypassing traditional sentiment analysis to avoid the pitfalls of noisy and ambiguous sentiment labels

Temporal Alignment: Align tweets with stock price movements over different time frames (weekly, monthly, quarterly) to understand the impact of social media on stock price predictions

Model Evaluation: Evaluate the performance of various NLP techniques and supervised learning models (Logistic Regression, LSTM, DistilBERT) to identify the most effective combination for predicting stock price changes using Twitter data

Creating new dataset: Labeling

Ex: Q1 2015 Tweet

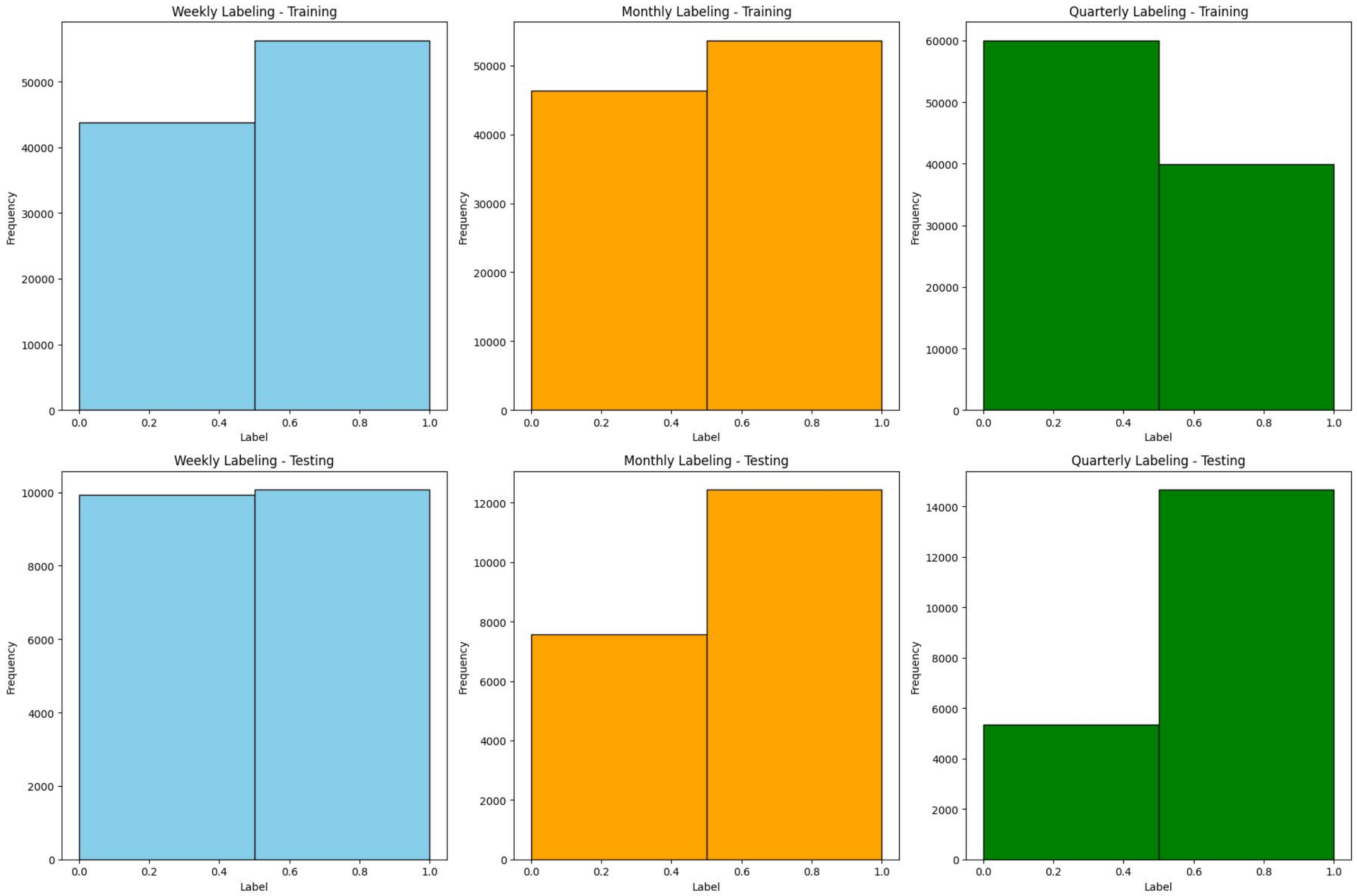
RIP Tesla Model Y sales: BYD has officially rolled out the Sea Lion 07 EV, a pure electric SUV that is the first model based on the e-Platform 3.0 Evo that boasts increased performance. It starts at $26,000, and you know what that means — more $TSLA price cuts are coming! 😂 https://t.co/oLfy5csyvj https://t.co/4d35XP3U1E

a) Q1 2015 - Q2 2015 <u>positive</u> % change -> **1**

a) Q1 2015 - Q2 2015 <u>negative</u> % change -> **0**

New dataset distribution

By incorporating the actual dates of tweets and aligning them with stock price movements over specified periods (weekly, monthly, or quarterly), we aim to create a more robust prediction model



Data Cleaning

1. **Removing ‘@’ mentions:** user-specific and do not contribute to the overall sentiment or context of the tweet. Removing them prevents the models from being confused by irrelevant user handles
2. **Removing URLs:** contain no meaningful linguistic information and can clutter the text. By removing them, we ensure the models focus on the actual content of the tweets.
3. **Removing all non-alphanumeric characters except spaces:** While this is more debatable, especially for complex models like BERT/DistilBERT, we decided to remove non-alpha numeric characters to create a cleaner, more uniform text for the models to process
4. **Converting text to lowercase:** standardizes the input, preventing the models from treating words differently based on capitalization, which can improve consistency and accuracy
5. **Removing redundant spaces:** Removing extra spaces ensures that the text is uniformly formatted, which helps in tokenization and improves model efficiency
6. **Replacing any instance of ‘tsla’ with ‘tesla’:** ensures that variations like 'tsla', which were very common in the tweets we were analyzing, are recognized as the same entity (same tokenization), which improves the model's ability to correctly identify and learn from relevant mentions

Example:

RIP Tesla Model Y sales: BYD has officially rolled out the Sea Lion 07 EV, a pure electric SUV that is the first model based on the e-Platform 3.0 Evo that boasts increased performance. It starts at $26,000, and you know what that means — more $TSLA price cuts are coming! 😂 https://t.co/oLfy5csyvj https://t.co/4d35XP3U1E

RIP Tesla Model Y sales: BYD has officially rolled out the Sea Lion 07 EV, a pure electric SUV that is the first model based on the e-Platform 3.0 Evo that boasts increased performance. It starts at $26,000, and you know what that means — more $TSLA price cuts are coming! 😂 https://t.co/oLfy5csyvj https://t.co/4d35XP3U1E

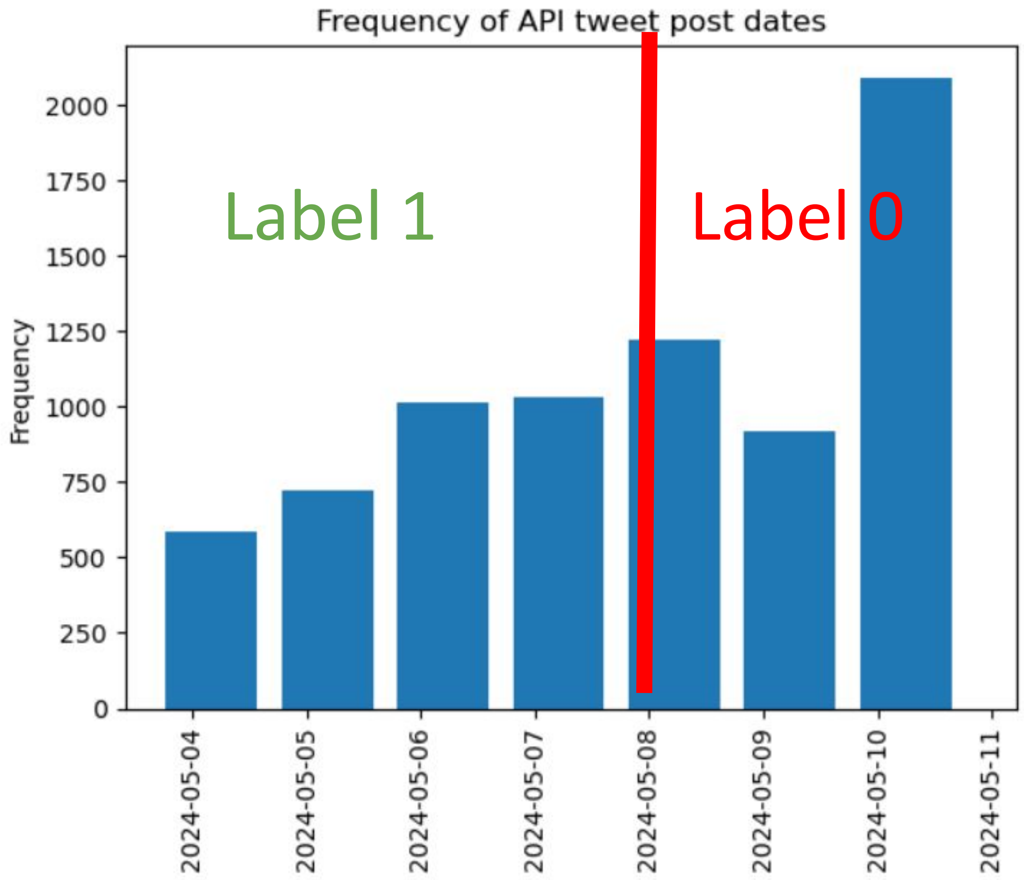
Collecting Tweets from Twitter API

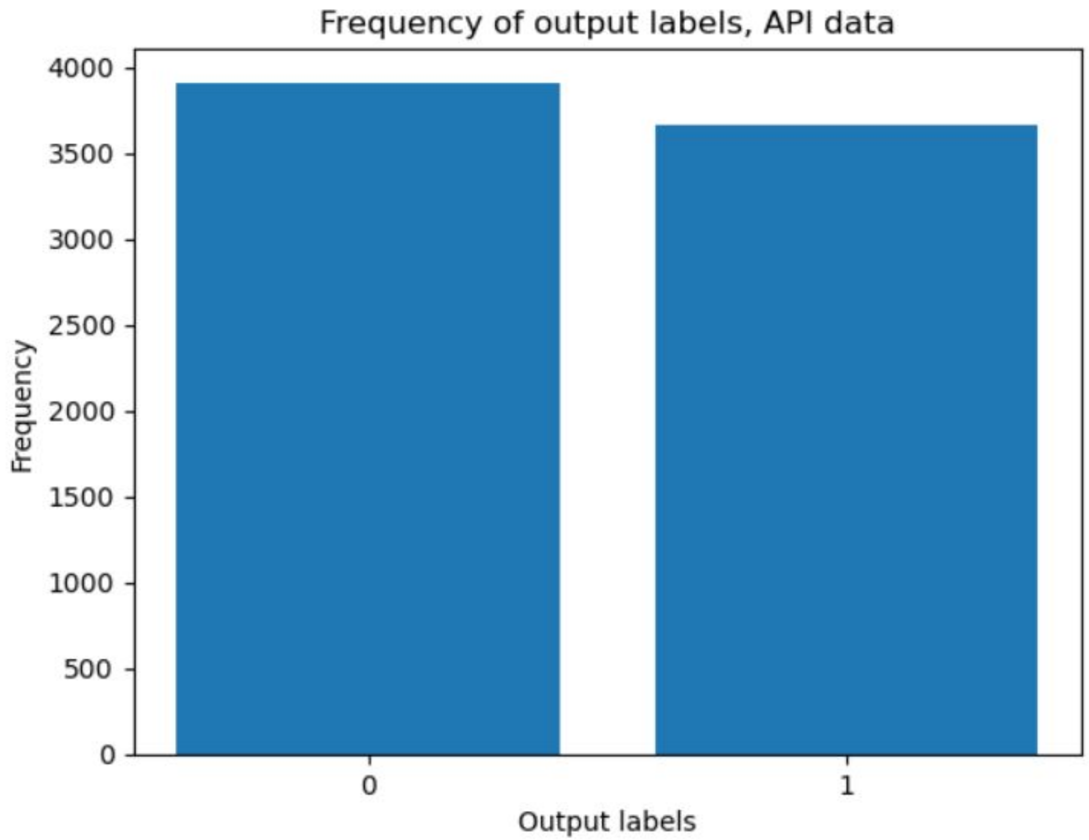
Searching for Tweets

“TSLA (Model X OR Model 3 OR Model X OR…) lang:en -is:retweet”

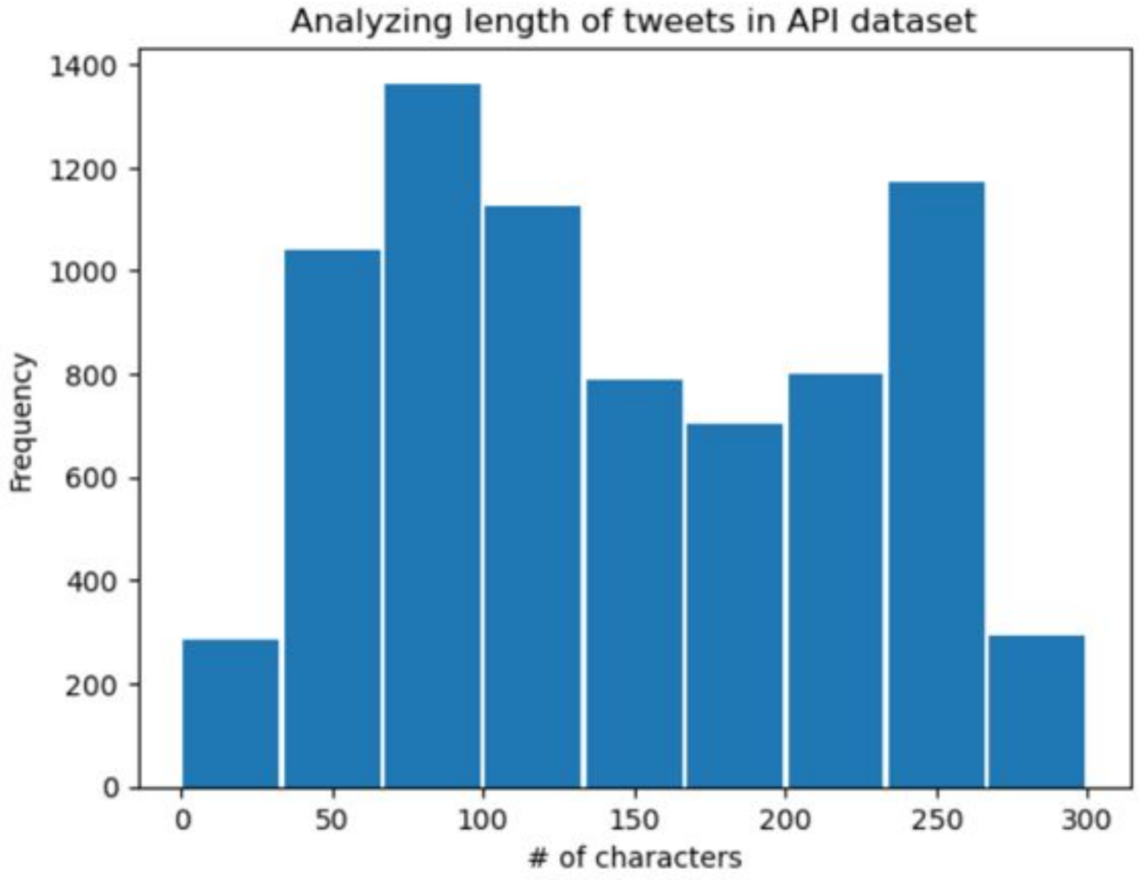
“TSLA Tesla lang:en -is:retweet”

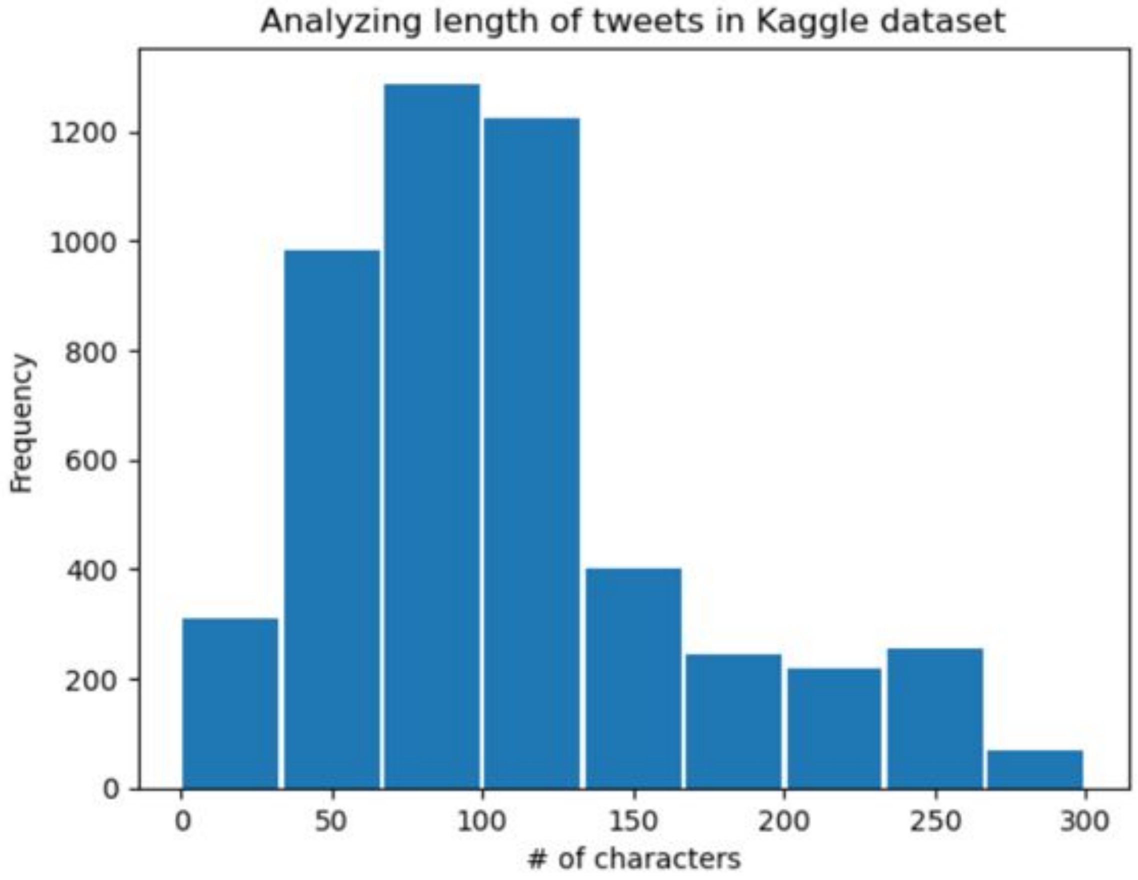
Dataset Visualization



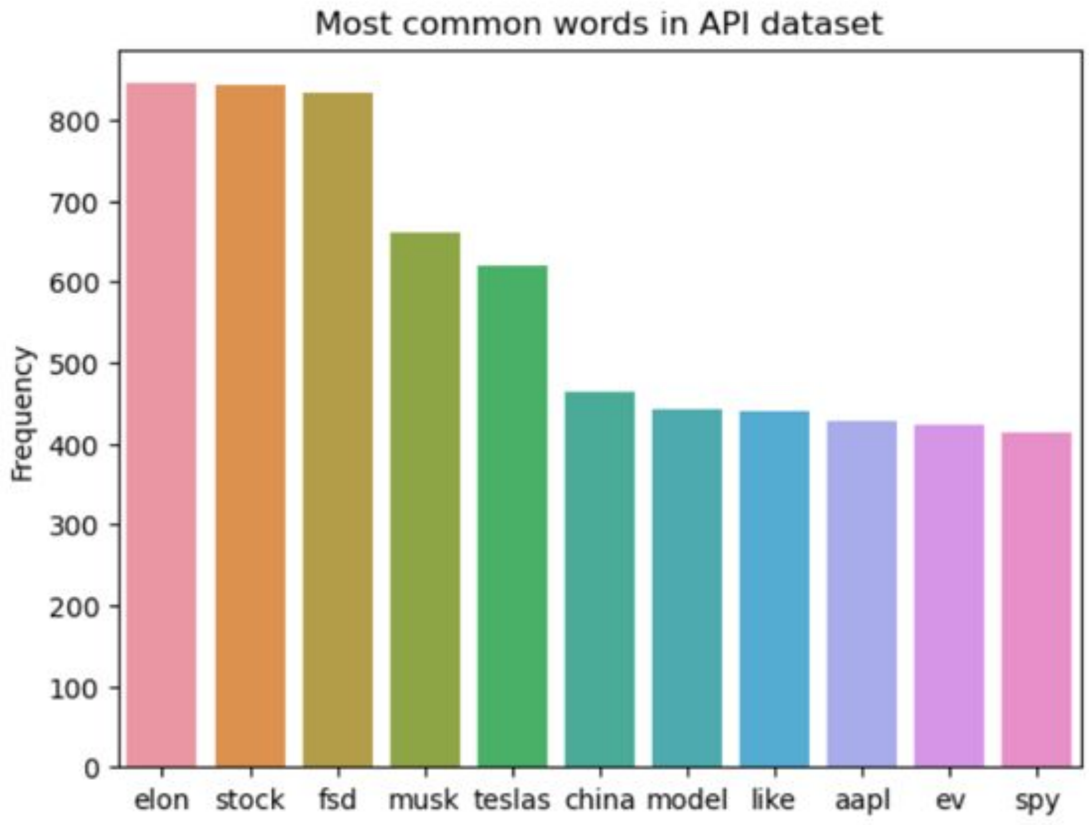


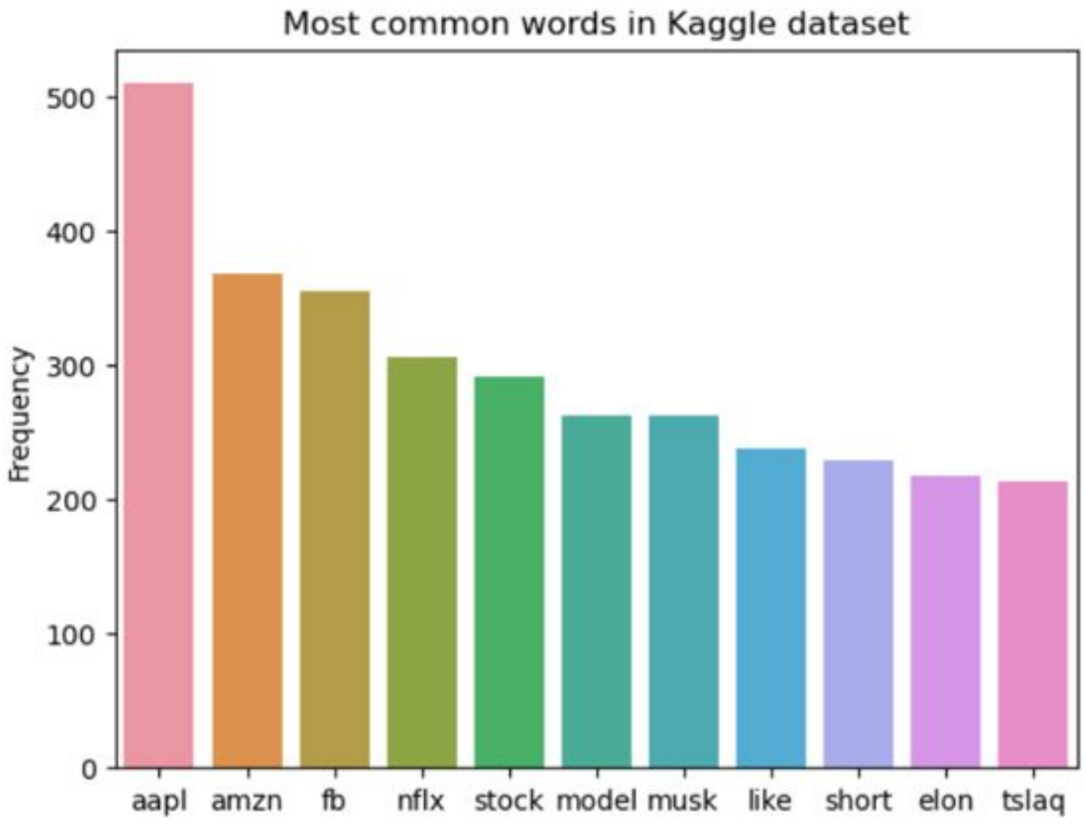
Comparing API with Kaggle



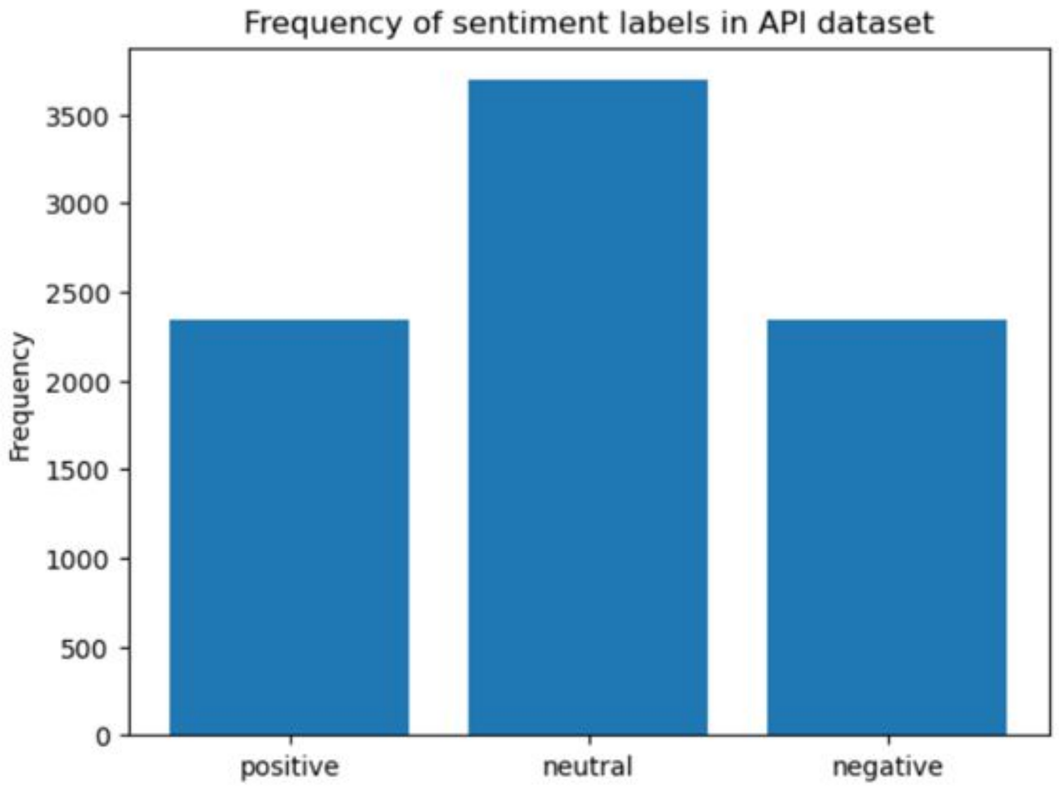


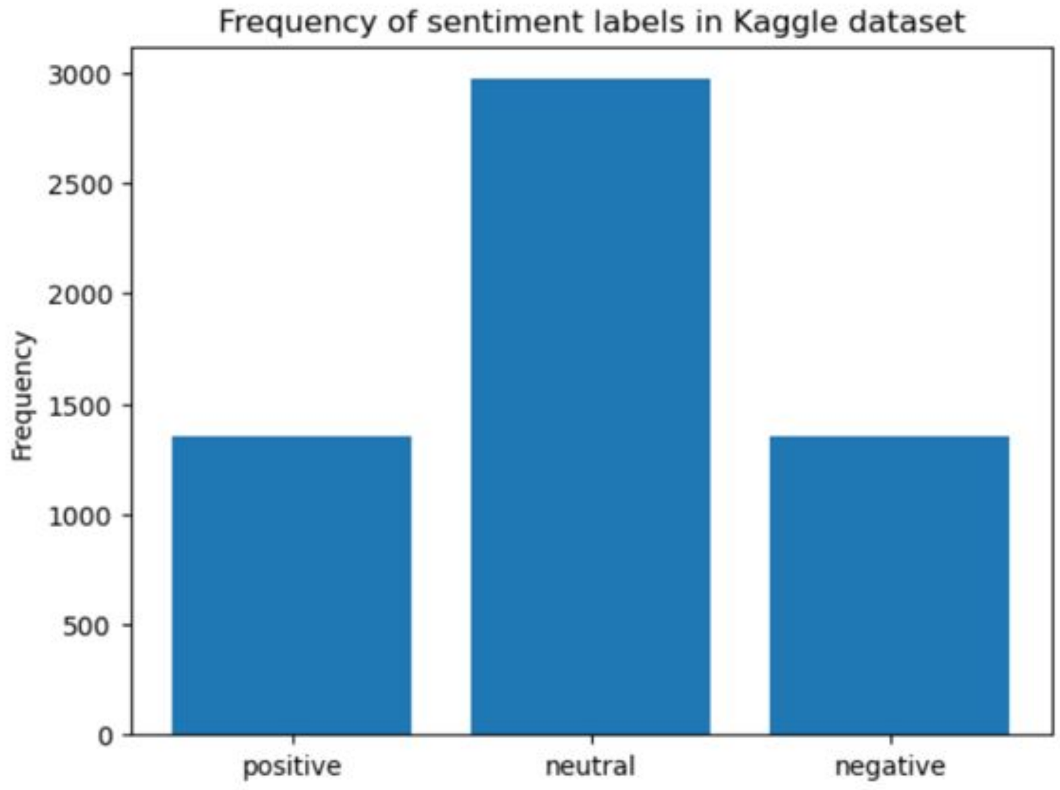
Comparing API with Kaggle





Comparing API with Kaggle





Logistic Regression

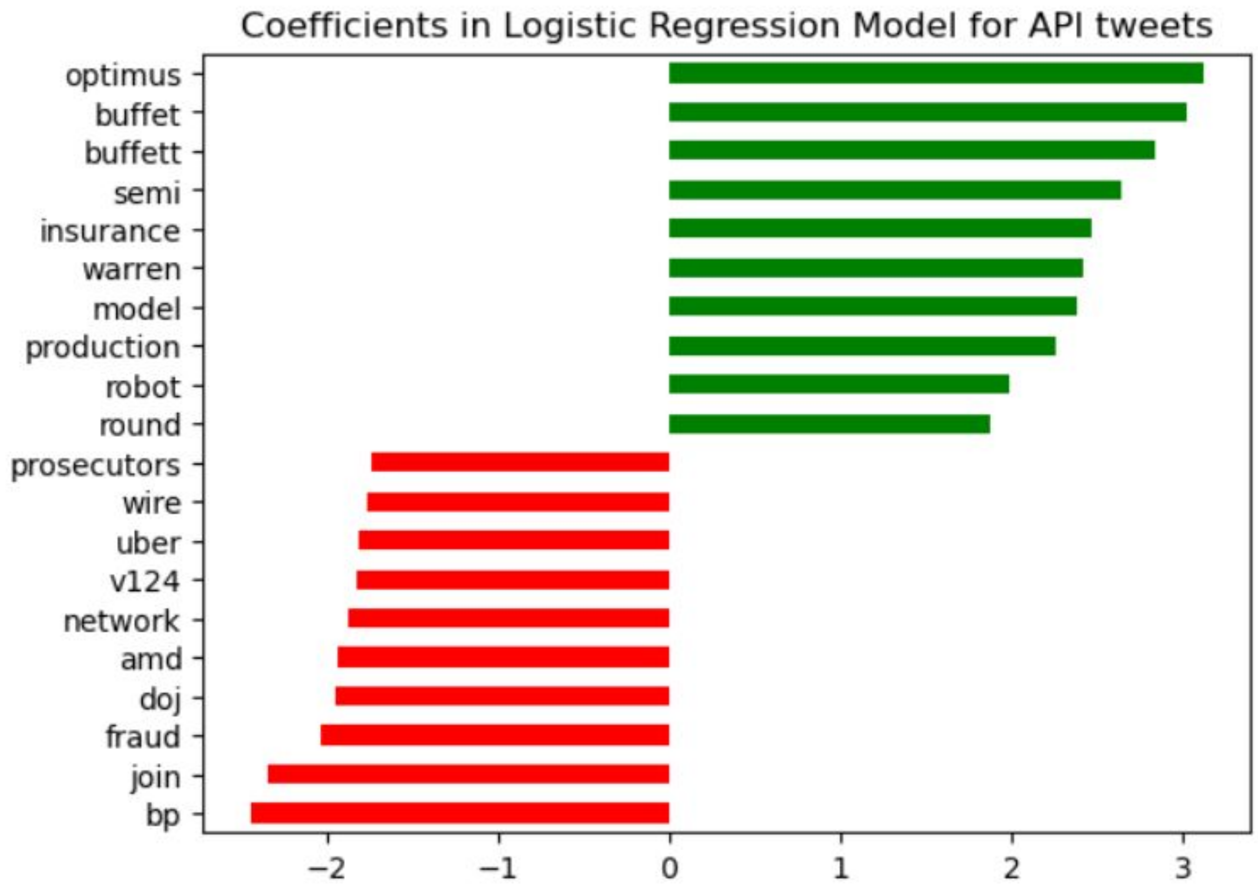
Bag of Words Accuracies

Weekly: 51%

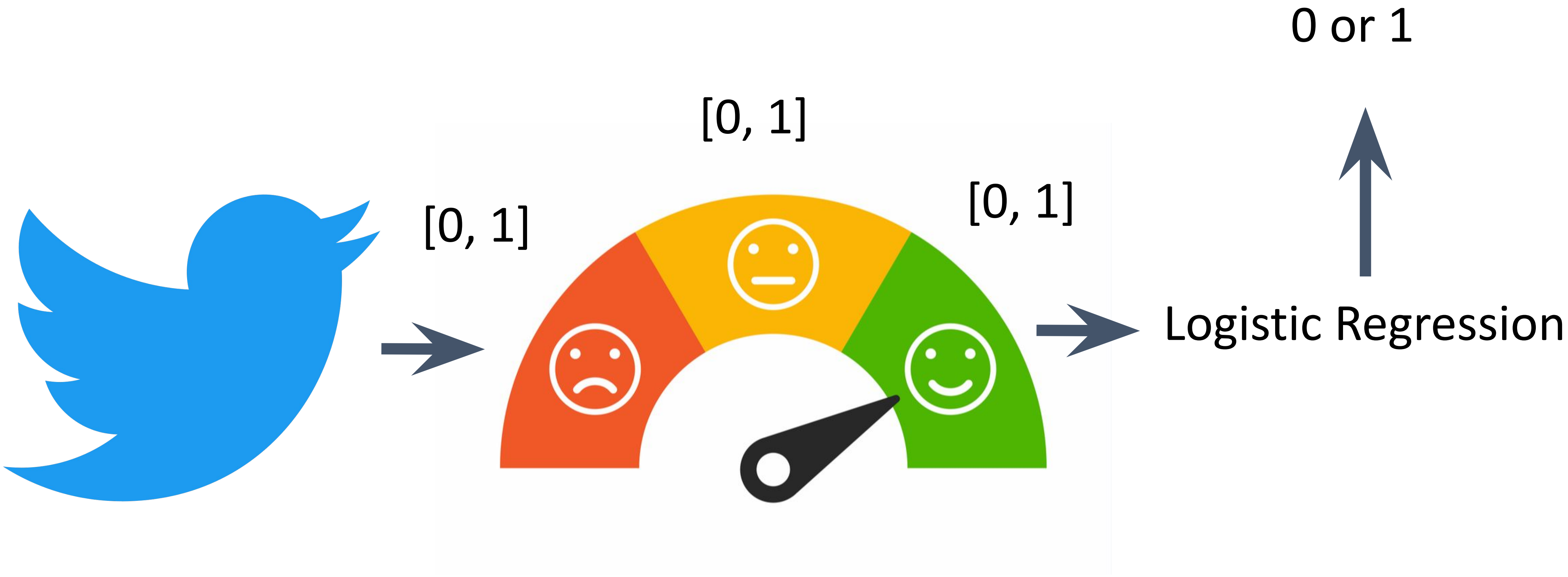
Monthly: 49%

Quarterly: 48%

API: 68%



Sentiment Analysis



Sentiment Accuracies

Weekly: 53%

Monthly: 49%

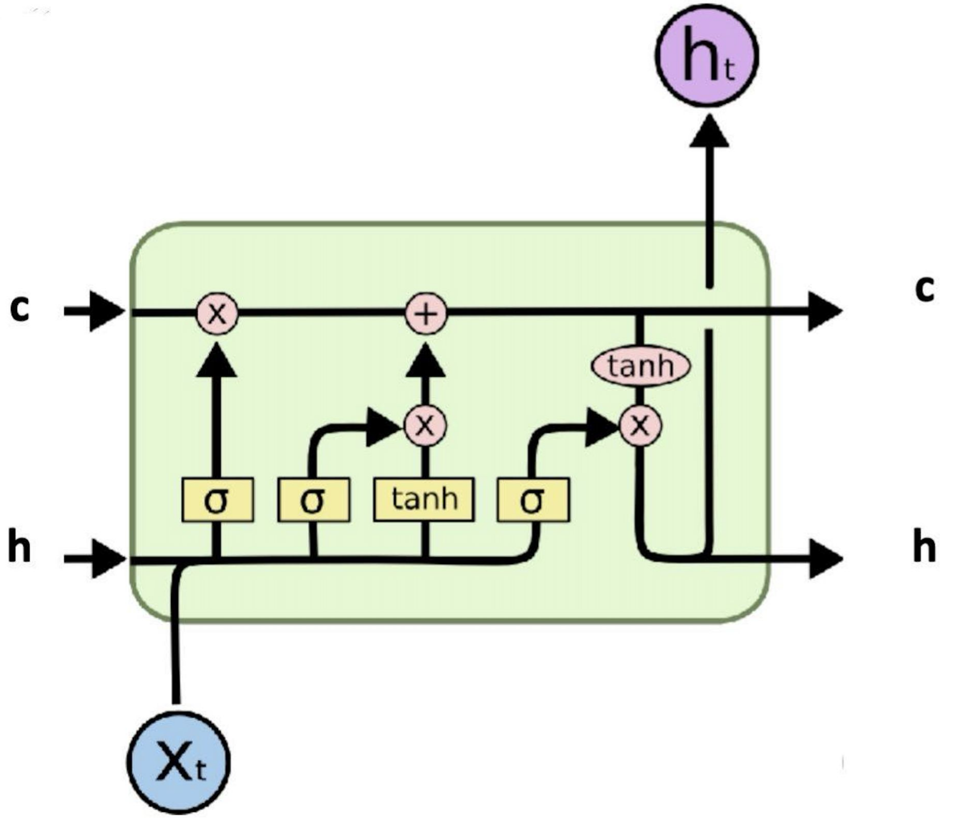
Quarterly: 49%

API: 52%

LSTM Models

Introduction to LSTM Models

- A type of recurrent neural network (RNN).
- Handles sequence data effectively.
- Ideal for Twitter data due to memory of past inputs.

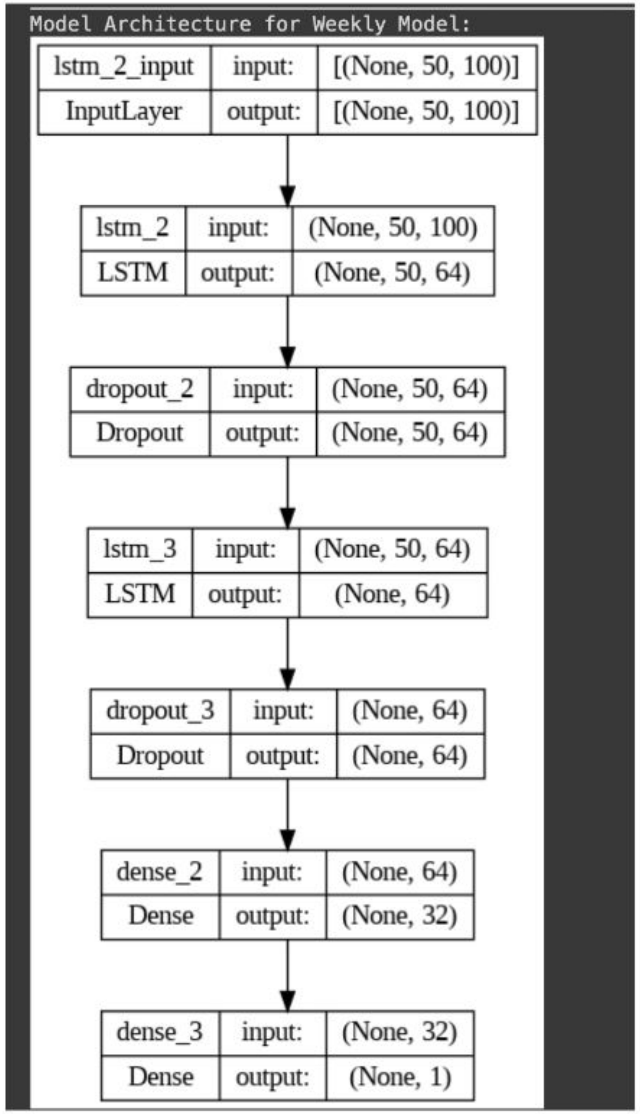


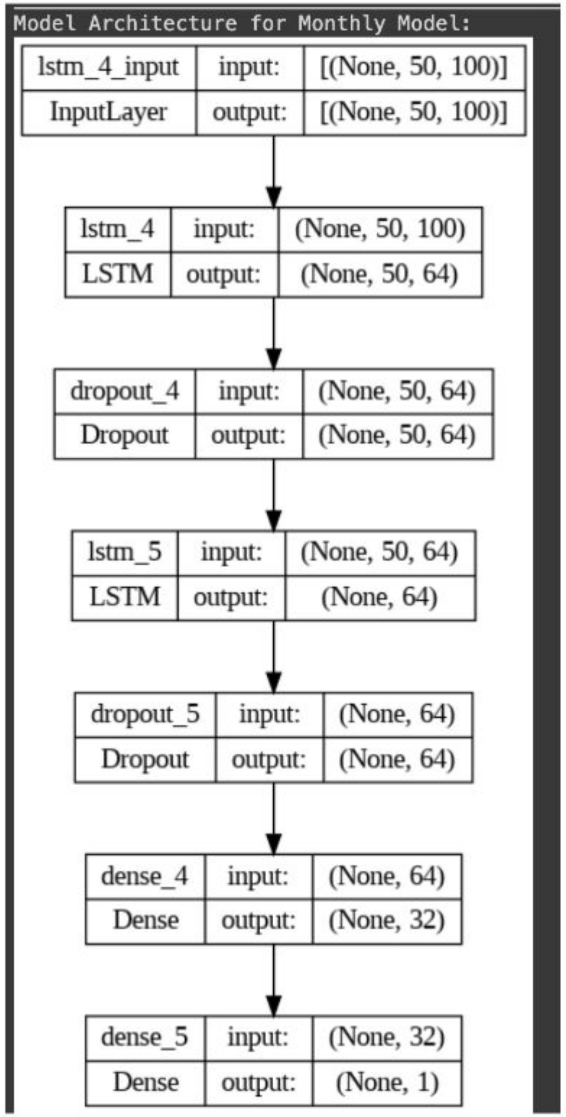
LSTM Model Design

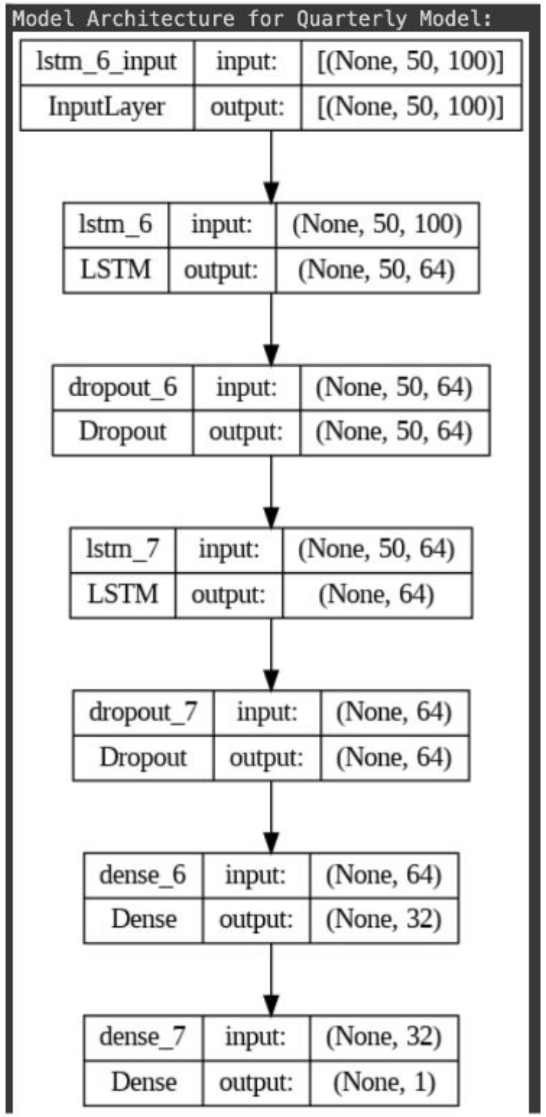
.

- Model Structure: Dual-layer LSTM with 64 units each for handling sequential tweet data.
- Embedding Layer: Utilized Word2Vec to convert tweets into numerical vectors of size 100.
- Optimizer: Adam with a learning rate of 0.0005 to minimize binary cross-entropy loss.
- Regularization: Dropout layers at 0.25 to prevent overfitting during training.
- Early Stopping: Employed to halt training when validation accuracy ceases to improve, ensuring generalization.
- Input and Output: Processes sequences of maximum 50 tokens and predicts stock price movements (increase/decrease) using a sigmoid activation.

3 different models







LSTM Results

Dataset	Accuracy	Precision	Recall	F1 Score
Weekly	0.4965	0.4977	0.8438	0.6261
Monthly	0.4945	0.4919	0.4457	0.4677
Quarterly	0.4910	0.4846	0.3318	0.3939

Table 1: Optimized performance metrics for test data

Model	Accuracy	Precision	Recall	F1 Score
Weekly	0.4660	0.4700	0.8081	0.5943
Monthly	0.4809	0.4675	0.5215	0.4930
Quarterly	0.5245	0.5077	0.5750	0.5392

Table 2: Performance metrics for curr_tweets dataset

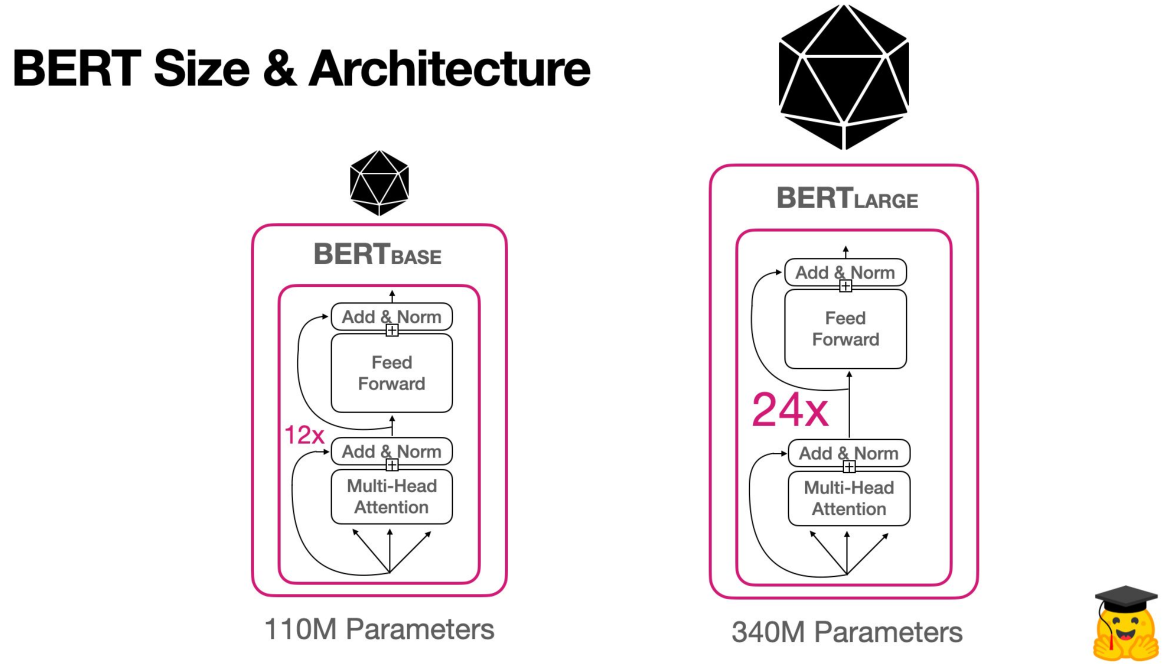
DistilBERT/BERT (“Bidirectional Encoder Representations from Transformers”) Model

Introduction to DistilBERT Model

DistilBERT:

- ~40% smaller than base BERT model
- Quicker training, better for smaller NLP tasks
- ~66 million parameters

Normal BERT model architecture:

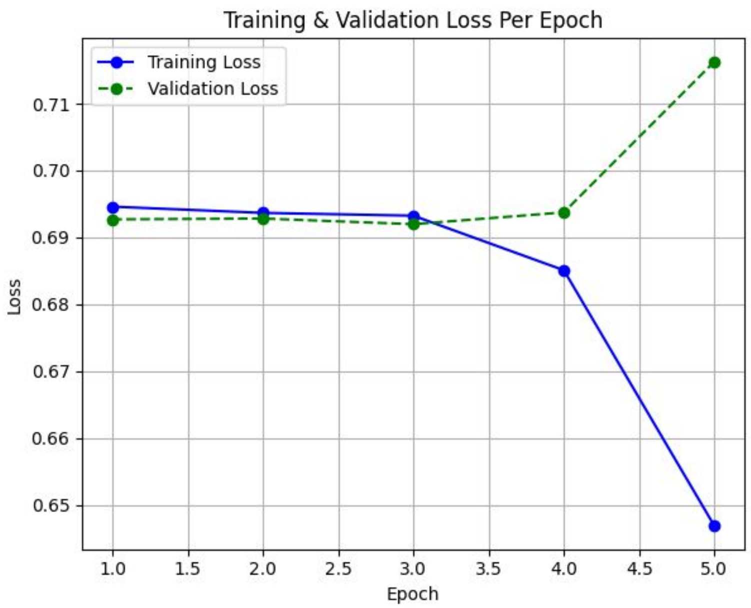


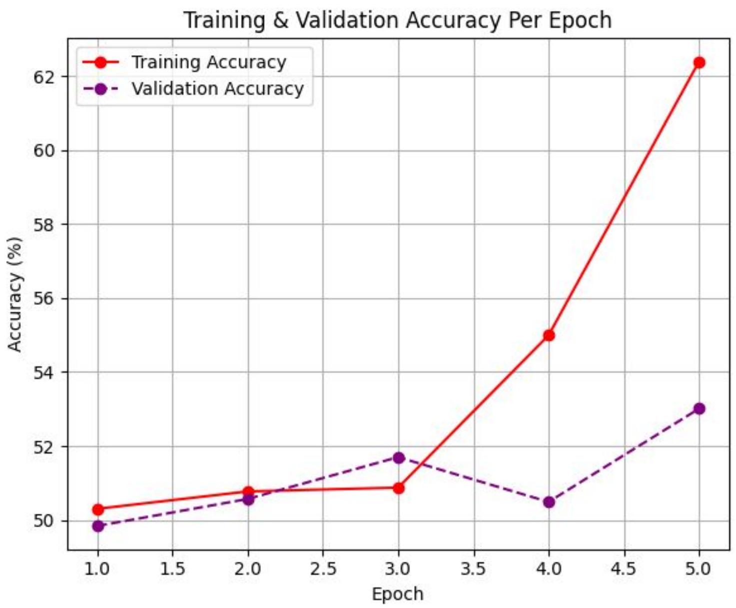
Training Specifications and Model Design

Training:

- Training Dataset Size: 40,000 cleaned, tokenized tweets
- Epochs: 5
- Compute: T4 GPU
- Batch Size: 32
- Dropout: Yes (built into DistilBERT architecture)
- Initial Learning Rate: 0.00005 (A smaller learning rate is recommended for fine-tuning tasks to ensure more precise adjustments to the pretrained model's parameters)
- Learning Rate Scheduler: Utilized PyTorch's 'ReduceLROnPlateau', which reduces the learning rate when validation loss does not significantly change for 2 iterations

Results - Weekly Model





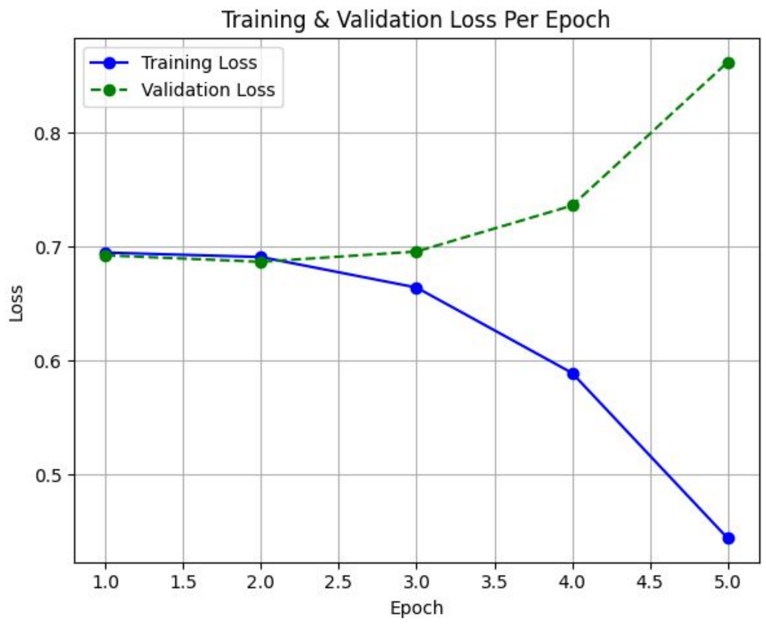
Classification Report (Weekly Model)

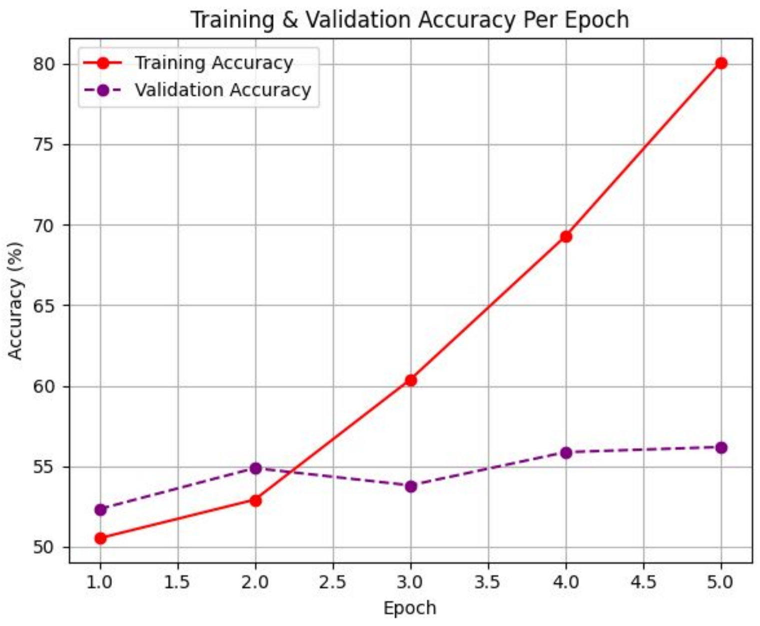
	Precision	Recall	F1
0 (Decrease)	0.50	0.71	0.58
1 (Increase)	0.50	0.29	0.37

Confusion Matrix (Weekly Model)

3510	1490
3527	1473

Results - Monthly Model





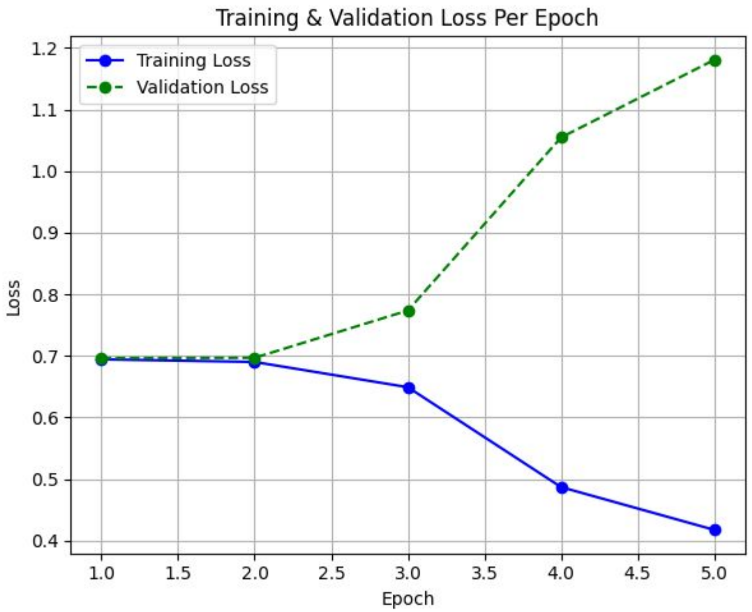
Classification Report (Monthly Model)

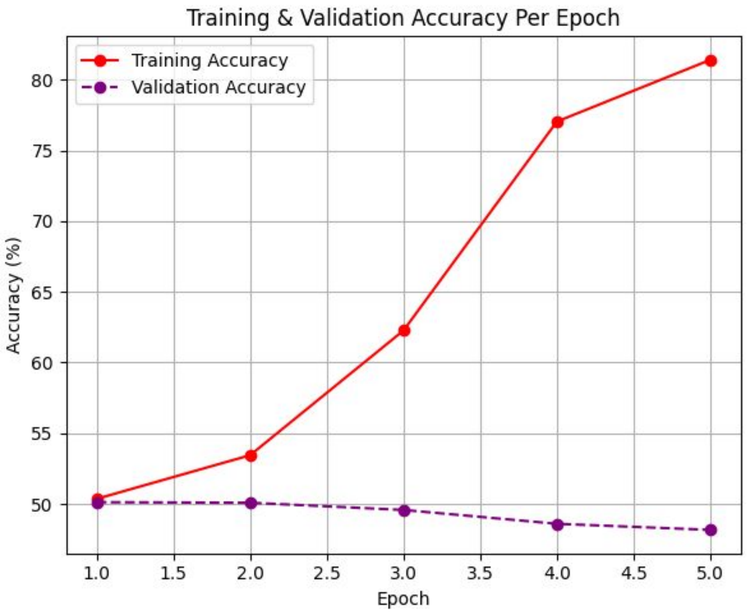
	Precision	Recall	F1
0 (Decrease)	0.48	0.37	0.42
1 (Increase)	0.49	0.60	0.54

Confusion Matrix (Monthly Model)

1858	3142
1997	3003

Results - Quarterly Model





Classification Report (Quarterly Model)

	Precision	Recall	F1
0 (Decrease)	0.49	0.40	0.44
1 (Increase)	0.49	0.58	0.53

Confusion Matrix (Quarterly Model)

2020	2980
2122	2878

Conclusion

Interpretation of Results

.

- Advanced NLP models like LSTM and DistilBERT offer deeper insights into stock trends.
- Accuracy around 50% despite of this indicates models' strengths lie beyond simple metrics.
- Emphasizes the need for balancing sensitivity and specificity in noisy environments like social media.

Challenges faced

.

- Navigated the complexities of processing noisy and unstructured Twitter data for accurate model input.
- Overcame computational constraints on Google Colab, optimizing training times and model efficiency.
- Adapted methodologies based on iterative feedback to refine data cleaning and model tuning processes.

Future Work

.

- Aim to enhance model precision without sacrificing recall by integrating advanced NLP techniques and hybrid models.
- Explore ways to balance true positives with minimizing false positives in predictive modeling.
- Extend models to other companies, and train models on more data from various social media platforms like Reddit and Threads.

Considerations to make

.

- Prioritize ethical considerations to avoid amplifying biases in social media data.
- Address potential impacts of automated predictions on financial markets and investment behaviors.
- Ensure future developments are responsible and contribute positively to the broader economic landscape.

The End