Tesla Stock Movement Based on Musk’s Tweet

San Jose State University DATA226 Group Project

Jie Heng (018321914) | Savitha Vijayarangan (018315986) | Lakshmi Bharathi Kumar (017613414) | Daniel Kim (014641497) | Andreah Cruz (013468910)

Abstract

This project explores the relationship between Elon Musk’s tweet sentiment data [1] and Tesla stock price [2] fluctuations. We collect both historical and real-time data and integrate them into a Snowflake data warehouse. Apache Airflow is used to schedule automated pipelines, and ELT transformations are handled using DBT. Visualization is performed through Superset dashboards. A linear regression model is built to predict Tesla’s short-term stock price trends based on sentiment signals and historical patterns. This end-to-end pipeline provides a foundation for actionable insights and data-driven financial forecasting.

Keywords

Tesla stock price prediction, ELT, DBT, Airflow, Snowflake, sentiment, Data Visualization.

I. Problem Statement

The financial market is highly sensitive to public sentiment, particularly when it involves influential figures. Tesla, one of the most closely watched tech companies, frequently experiences stock price volatility in response to statements made by its CEO, Elon Musk. Despite widespread speculation, investors often lack a structured, data-driven framework to assess whether Musk’s tweets genuinely influence short-term stock movements.

This project is cantered on analysing the impact of Elon Musk’s tweets on Tesla’s stock price. By integrating tweet data with stock market data, we aim to build a comprehensive data analytics pipeline using tools such as Snowflake, Apache Airflow, and dbt. Through automated sentiment analysis, data transformation, and in-database statistical modelling, we seek to identify measurable patterns and correlations between tweet sentiment and stock price behaviour.

The results of this analysis will be visualized in an interactive BI dashboard, providing actionable insights into how social media influence - specifically from Elon Musk - may shape Tesla’s stock trends and inform short-term investment decisions.

II. Dataset Description

This project combines Tesla stock price data with sentiment analysis of Elon Musk’s tweets to explore the relationship between public sentiment and stock market movements.

Tesla stock data (real-time dataset) is sourced from **Yahoo Finance** [3] using the yfinance API. It includes updated daily data such as open, close, high, low prices, and trading volume. This dataset provides a reliable foundation for time-series and volatility analysis across different timeframes.

Elon Musk tweet sentiment data (historical) covers tweets from 2023 to 2025 [4], initially collected in all_musks.csv. Sentiment analysis was conducted using the VADER model in Google Colab [5], producing a daily summary file (daily_sentiment.csv).

By integrating these two datasets into the date field, we can investigate correlations between sentiment fluctuations and Tesla’s stock performance. This combined dataset supports trend analysis, anomaly detection, and the development of predictive models for financial decision-making.

III. Technical Requirements

3.1 Technical Requirements

This system architecture consists of a complete ETL and ELT pipeline for analyzing the relationship between Tesla's stock price and sentiment derived Elon Musk's tweets. The system integrates data extraction, sentiment classification, correlation analysis, and dashboard visualization. The main components are as follows:

1. Data Ingestion Layer (ETL)

Yahoo Finance API is used to extract real-time Tesla stock price data (open, close, volume, etc.).
Elon Musk tweet sentiment data is used to extract historical tweet sentiment data related to Tesla.
Tesla stock data and sentiment data are merged by date.
A scheduled Airflow DAG is used to automate and orchestrate the data extraction process.
Extracted raw data is stored in Snowflake staging tables for further processing.

2. Data Processing and Sentiment Analysis Layer (ELT)

Data are retrieved from Snowflake and processed using dbt models (input and output).
Extracting new features from existing features for predictive modelling.
Transformed results are stored in analytical tables within Snowflake.

3. Correlation and Statistical Analysis Layer

Linear Regression model is used to explore the correlation between tweet sentiment and stock price fluctuations.
Key patterns and trends (e.g., whether negative sentiment correlates with price drops) are analyzed.

4. Visualization and Insight Delivery Layer

Processed data is visualized using Apache Superset.
Dashboards display trends such as stock price movements, prediction and their correlations.
This modular architecture ensures scalability, automation, and reproducibility. By leveraging modern data platforms like Snowflake and orchestration tools like Airflow, the system efficiently handles real-time and historical data to support financial insight generation.

3.2 Overall System Diagram

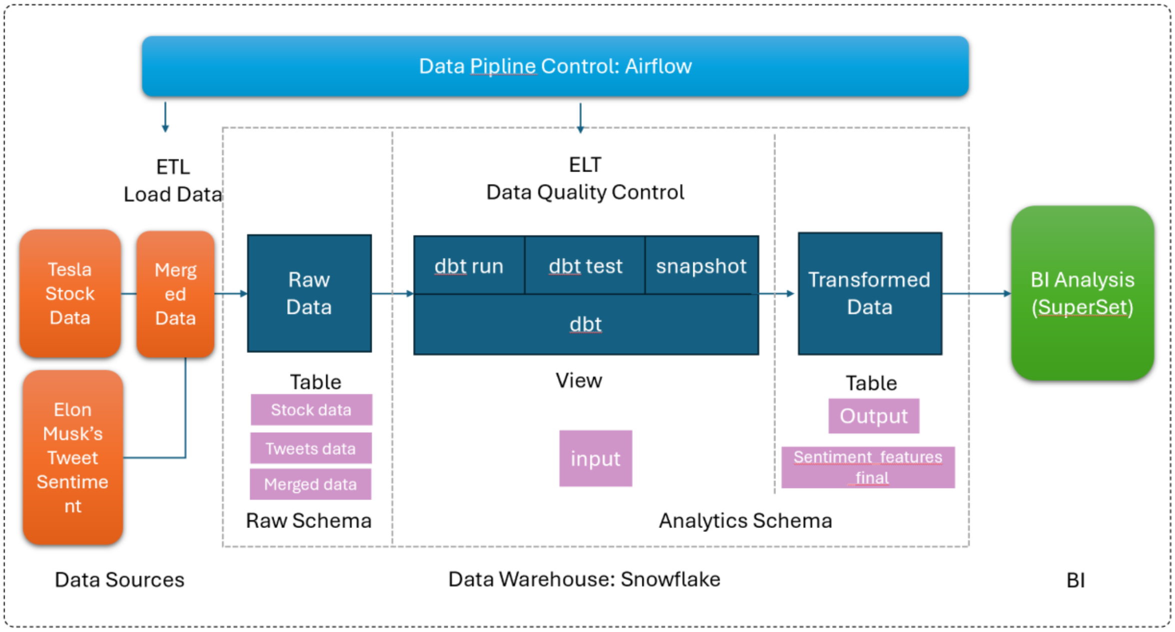


Figure 1: Overall System Architecture

IV. Data Warehouse Design

Table 1: raw.final_raw_data

Field Name	Data Type	Attributes	Constraints	Description
date	DATE	Not null	Primary Key with SYMBOL	Trading date
sentiment_score	FLOAT			
weighted_sentiment	FLOAT			
tweet_count	NUMBER			
total_likes	NUMBER			

Table 2: raw.final_raw_data

Field Name	Data Type	Attributes	Constraints	Description
date	DATE	Not null	Primary Key with SYMBOL	Trading date
open	FLOAT			Opening price
close	FLOAT			Closing price
Low	FLOAT			Lowest price
high	FLOAT			Highest price
volume	NUMBER			volume

Table 3: raw.final_raw_table

Field Name	Data Type	Attributes	Constraints	Description
date	DATE	Not null	Primary Key with SYMBOL	Trading date
sentiment_score	FLOAT			
weighted_impact	FLOAT			
tweet_count	NUMBER			
total_likes	NUMBER			
open	FLOAT			Opening price
close	FLOAT			Closing price
Low	FLOAT			Lowest price
high	FLOAT			Highest price
volume	NUMBER			volume
Price_change_pct	FLOAT			(current close – last close) / last close

Table 4: analytics.input

Field Name	Data Type	Attributes	Constraints	Description
date	DATE	Not null	Primary Key with SYMBOL	Trading date
sentiment_score	FLOAT			
weighted_sentiment	FLOAT			
tweet_count	NUMBER			
total_likes	NUMBER			
lag_1_sentiment	FLOAT			
avg_3d_sentiment	FLOAT			
open	FLOAT			Opening price
close	FLOAT			Closing price
volume	NUMBER			volume
price_change_pct	FLOAT			(current close – last close) / last close

avg_7d_close	FLOAT			Average of the closing prices over the past 7 days

Table 5: analytics.output

Field Name	Data Type	Attributes	Constraints	Description
date	DATE	Not null	Primary Key with SYMBOL	Trading date
close	FLOAT			Closing price
predicted_close_price	FLOAT			Predicted price

Table 6: analytics.sentiment_features_final

Field Name	Data Type	Attributes	Constraints	Description
date	DATE	Not null	Primary Key with SYMBOL	Trading date
sentiment_score	FLOAT			
tweet_count	NUMBER			
lag_1_sentiment	FLOAT			
close	FLOAT			Closing price
volume	NUMBER			volume
avg_7d_close	FLOAT			Average of the closing prices over the past 7 days
weighted_impact	FLOAT			
avg_3d_close	FLOAT			Average of the closing prices over the past 3 days

V. Technical Solution

5.1 ETL Process (Airflow)

This project uses Airflow DAG to define a ETL pipeline that automates the integration of Tesla stock data and Elon Musk tweet sentiment data into a Snowflake data warehouse. It runs daily at 2:30 AM and consists of three main tasks. First, it extracts Tesla stock price data from Yahoo Finance using yfinance, formats it, and loads it into the dev.raw.stock_data table in Snowflake. Second, it reads a preprocessed sentiment dataset (daily_sentiment.csv)—which was generated in Google Colab by applying VADER sentiment analysis on the all_musks.csv tweet dataset—and loads the daily aggregated results (including sentiment scores, tweet counts, and engagement metrics) into the dev.raw.tweet_data table. The daily_sentiment.csv file was placed under the project's /data directory, from which it is accessed during the ETL process. Finally, the pipeline merges both datasets on the date field into a unified dev.raw.final_raw_table, calculating the daily percentage price change (price_change_pct). The pipeline uses SQL MERGE logic for incremental updates and ensures transactional consistency with BEGIN, COMMIT, and ROLLBACK statements.

This DAG enables reliable, scheduled ETL operations and lays the groundwork for subsequent dbt-based modeling and analytics.

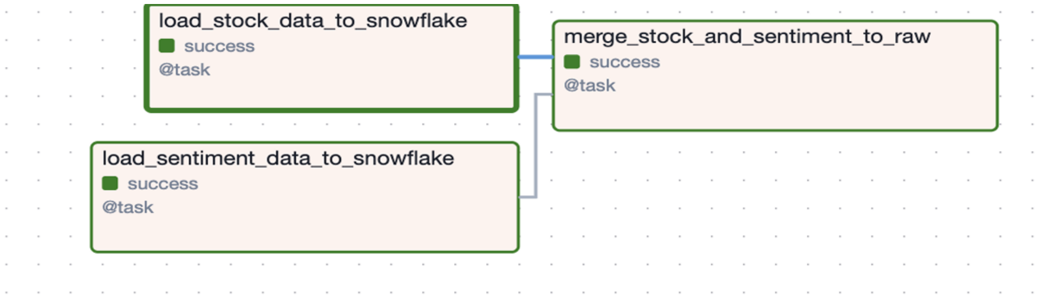

load_stock_data_to_snowflake
success
@task
merge_stock_and_sentiment_to_raw
success
@task
load_sentiment_data_to_snowflake
success
@task

Figure 2: ETL tasks as Airflow DAGS

5.2 ELT Process (dbt)

After the ETL pipeline loads cleaned stock and sentiment data into Snowflake, the ELT process begins using dbt. The dbt project, initialized via CLI, contains structured directories for models, seeds, macros, tests, and snapshots. The transformation logic is defined in a series of modular SQL models under the models/ directory. First, input.sql reads from the final_raw_table (produced by the ETL process) and performs feature engineering by generating lagged values, rolling averages, and other predictors relevant to stock price movements. This enriched dataset is passed to sentiment_features_final.sql, which finalizes the feature set, ensuring it's clean and ready for modeling. Finally, output.sql manually applies a linear regression formula—developed and trained in Colab—to generate predicted_price_change. It also joins with a regression_metrics table (seeded from a CSV containing RMSE, MAE, and R² values) to append model performance scores. This layered approach follows dbt's best practices of modularity and traceability, producing a reliable and analytics-ready prediction table as the final output.

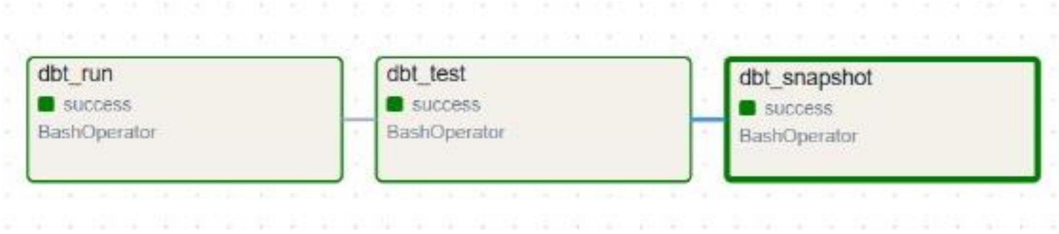
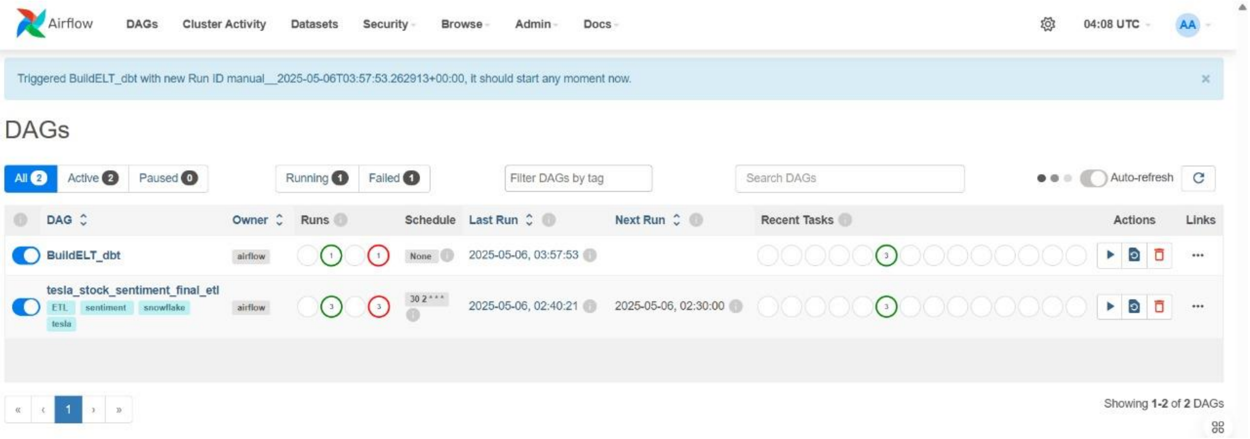


Figure 3: dbt tasks as Airflow DAGS

1. **Source Data**
   final_raw_table (merged stock + sentiment data) is loaded into Snowflake's raw schema using an Airflow ETL pipeline.
2. **Feature Engineering** – input.sql
   - Reads from final_raw_table via {{ source('raw', 'final_raw_table') }}
   - Calculates:
     - Lagged sentiment
     - 3-day avg sentiment
     - 7-day avg stock price
     - Price change percentage
3. **Weighted sentiment impact**
   Model-Ready Features – sentiment_features_final.sql
   - Filters the engineered data to remove NULLs in price_change_pct
   - Creates a clean dataset for modeling
4. **Prediction Logic** – output.sql
   - Applies a hardcoded linear regression equation (trained externally in Python)

   - Predicts stock price change based on sentiment and market features

5. **Output Table**
   - Contains actual vs predicted price changes
   - Used for performance evaluation and dashboard visualizations
6. **Materialization Strategy:**
   - All models (input, sentiment_features_final, output) are materialized as tables for performance and caching



Triggered BuildELT_dbt with new Run ID manual__2025-05-06T03:57:53.262913+00:00, it should start any moment now.

DAGs

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Actions	Links
BuildELT_dbt	airflow	1 1	None	2025-05-06, 03:57:53		3		...
tesla_stock_sentiment_final_etl ETL sentiment snowflake tesla	airflow	3 3	30 2 * * *	2025-05-06, 02:40:21	2025-05-06, 02:30:00	3		...

Showing 1-2 of 2 DAGs

Figure 4: dbt run in airflow

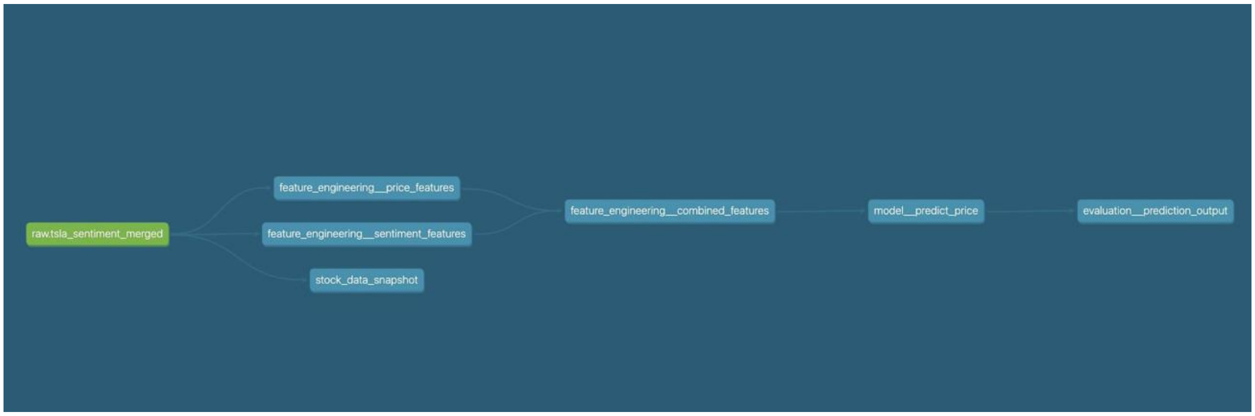

raw.tsla_sentiment_merged
feature_engineering__price_features
feature_engineering__sentiment_features
stock_data_snapshot
feature_engineering__combined_features
model__predict_price
evaluation__prediction_output

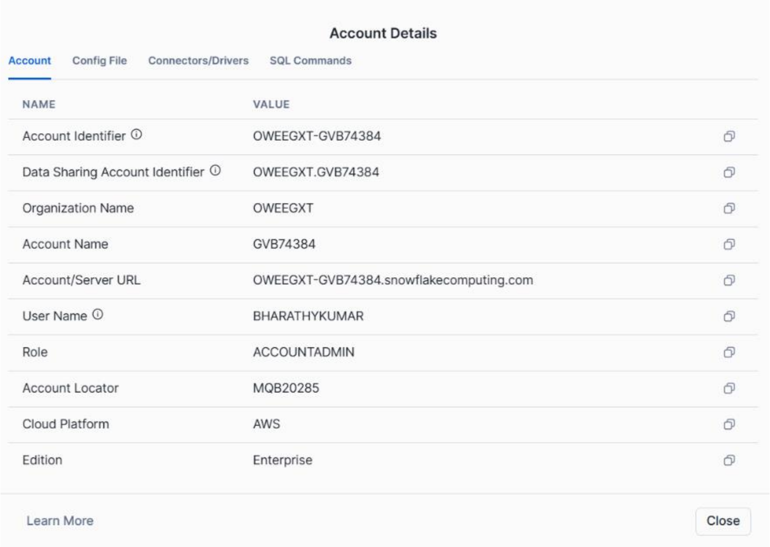
Figure 5: Data lineage Graph

5.3 Snowflake

Use this account to access the Snowflake:

User name: Bharathykumar

Pw: bharathy@sjsu25



Account Details

NAME	VALUE
Account Identifier ⓘ	OWEEGXT-GVB74384
Data Sharing Account Identifier ⓘ	OWEEGXT.GVB74384
Organization Name	OWEEGXT
Account Name	GVB74384
Account/Server URL	OWEEGXT-GVB74384.snowflakecomputing.com
User Name ⓘ	BHARATHYKUMAR
Role	ACCOUNTADMIN
Account Locator	MQB20285
Cloud Platform	AWS
Edition	Enterprise

We used the connection and variables in airflow to connect snowflake. And the tables after ETL process are:

RAW.STOCK_DATA

RAW.TWEET_DATA

RAW.FINAL_RAW_TABLE

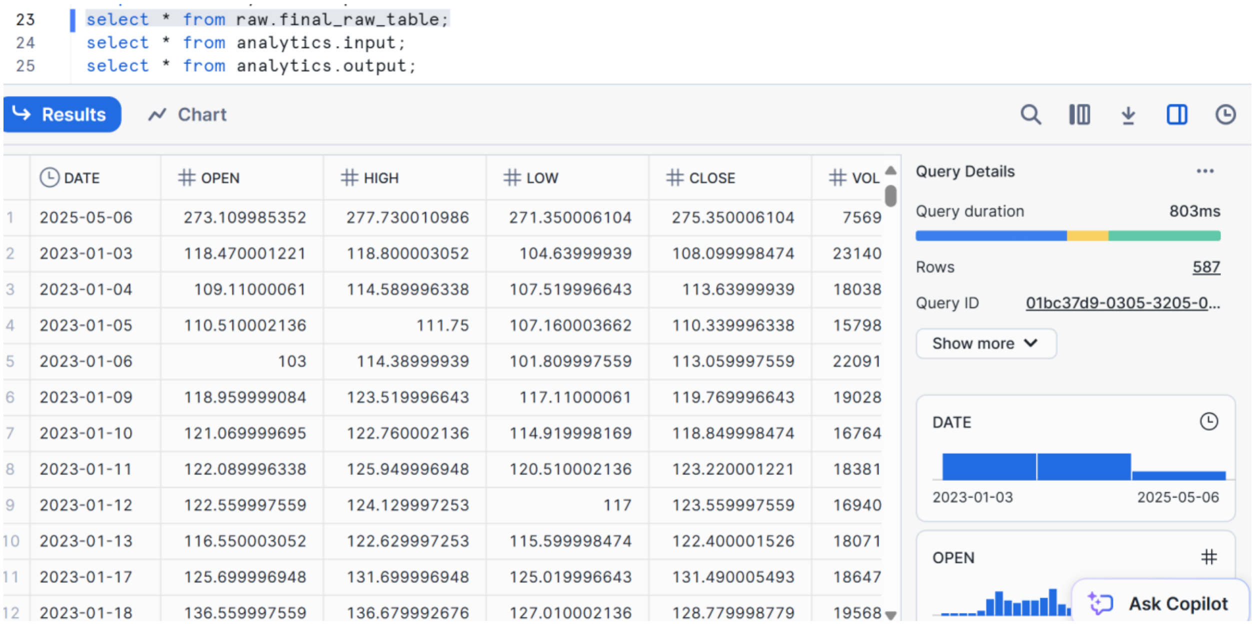
Tables after ELT process are:

ANALYTICS.INPUT

ANALYTICS.OUTPUT

ANALYTICS.SENTIMENT_FEATURES_FINAL

The following are screenshots of some tables stored in snowflake:



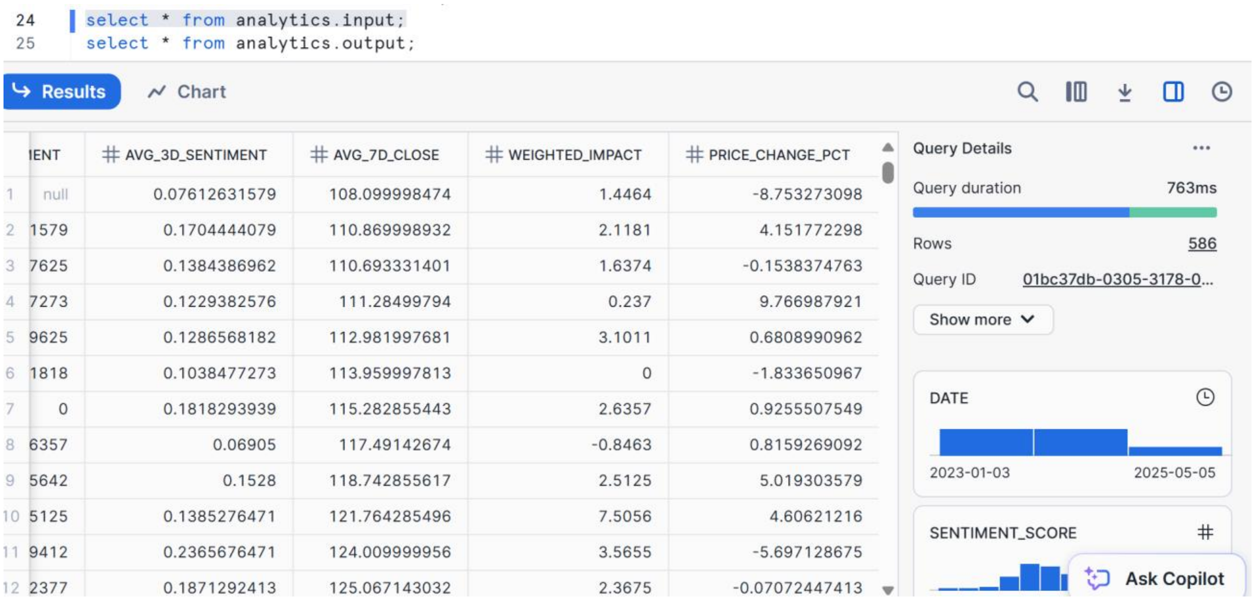
```
select * from raw.final_raw_table;
select * from analytics.input;
select * from analytics.output;
```

	DATE	OPEN	HIGH	LOW	CLOSE	VOL
1	2025-05-06	273.109985352	277.730010986	271.350006104	275.350006104	7569
2	2023-01-03	118.470001221	118.800003052	104.63999939	108.099998474	23140
3	2023-01-04	109.11000061	114.589996338	107.519996643	113.63999939	18038
4	2023-01-05	110.510002136	111.75	107.160003662	110.339996338	15798
5	2023-01-06	103	114.38999939	101.809997559	113.059997559	22091
6	2023-01-09	118.959999084	123.519996643	117.11000061	119.769996643	19028
7	2023-01-10	121.069999695	122.760002136	114.919998169	118.849998474	16764
8	2023-01-11	122.089996338	125.949996948	120.510002136	123.220001221	18381
9	2023-01-12	122.559997559	124.129997253	117	123.559997559	16940
10	2023-01-13	116.550003052	122.629997253	115.599998474	122.400001526	18071
11	2023-01-17	125.699996948	131.699996948	125.019996643	131.490005493	18647
12	2023-01-18	136.559997559	136.679992676	127.010002136	128.779998779	19568

Query Details

Query duration	803ms
Rows	587
Query ID	01bc37d9-0305-3205-0...

Figure 6: raw.final_raw_table



```sql
select * from analytics.input;
select * from analytics.output;
```

	IENT	AVG_3D_SENTIMENT	AVG_7D_CLOSE	WEIGHTED_IMPACT	PRICE_CHANGE_PCT
1	null	0.07612631579	108.099998474	1.4464	-8.753273098
2	1579	0.1704444079	110.869998932	2.1181	4.151772298
3	7625	0.1384386962	110.693331401	1.6374	-0.1538374763
4	7273	0.1229382576	111.28499794	0.237	9.766987921
5	9625	0.1286568182	112.981997681	3.1011	0.6808990962
6	1818	0.1038477273	113.959997813	0	-1.833650967
7	0	0.1818293939	115.282855443	2.6357	0.9255507549
8	6357	0.06905	117.49142674	-0.8463	0.8159269092
9	5642	0.1528	118.742855617	2.5125	5.019303579
10	5125	0.1385276471	121.764285496	7.5056	4.60621216
11	9412	0.2365676471	124.009999956	3.5655	-5.697128675
12	2377	0.1871292413	125.067143032	2.3675	-0.07072447413

Query Details

Query duration 763ms

Rows 586

Query ID 01bc37db-0305-3178-0...

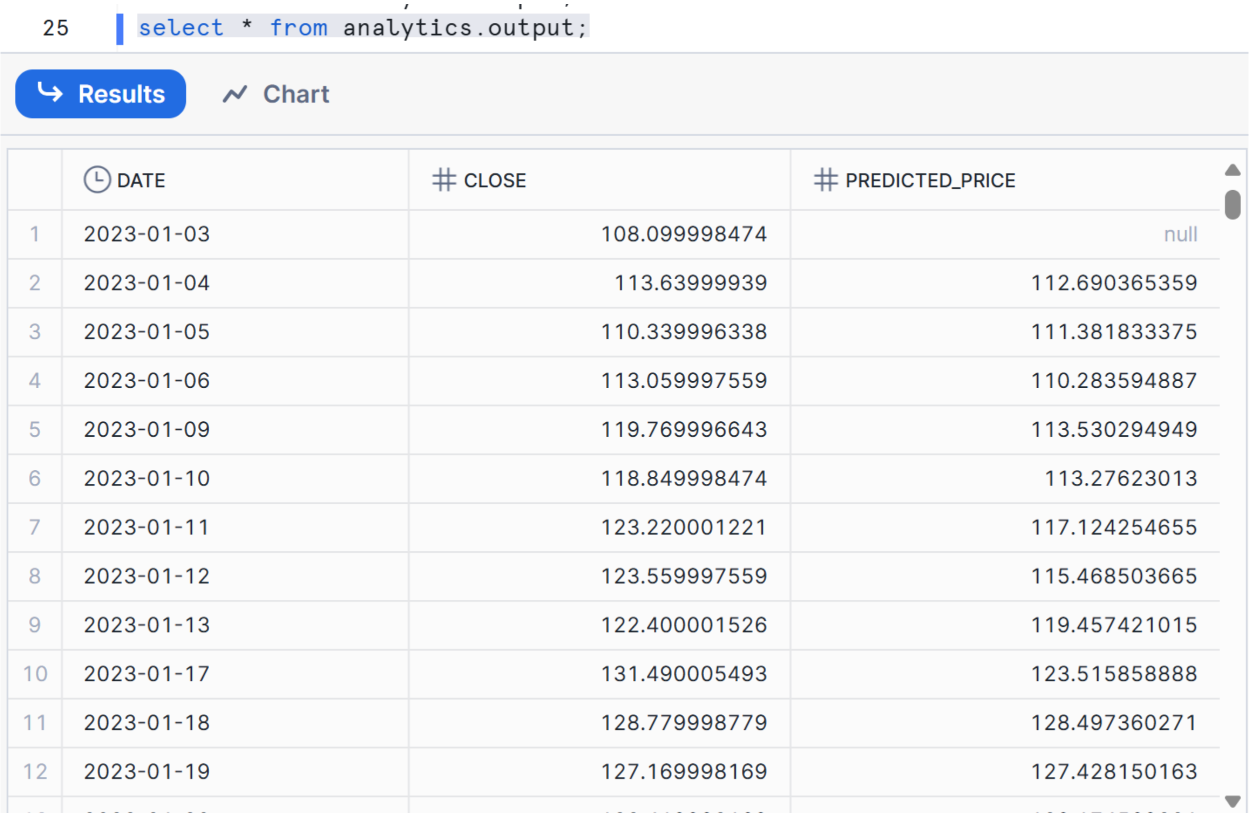
Show more

DATE

2023-01-03 2025-05-05

SENTIMENT_SCORE

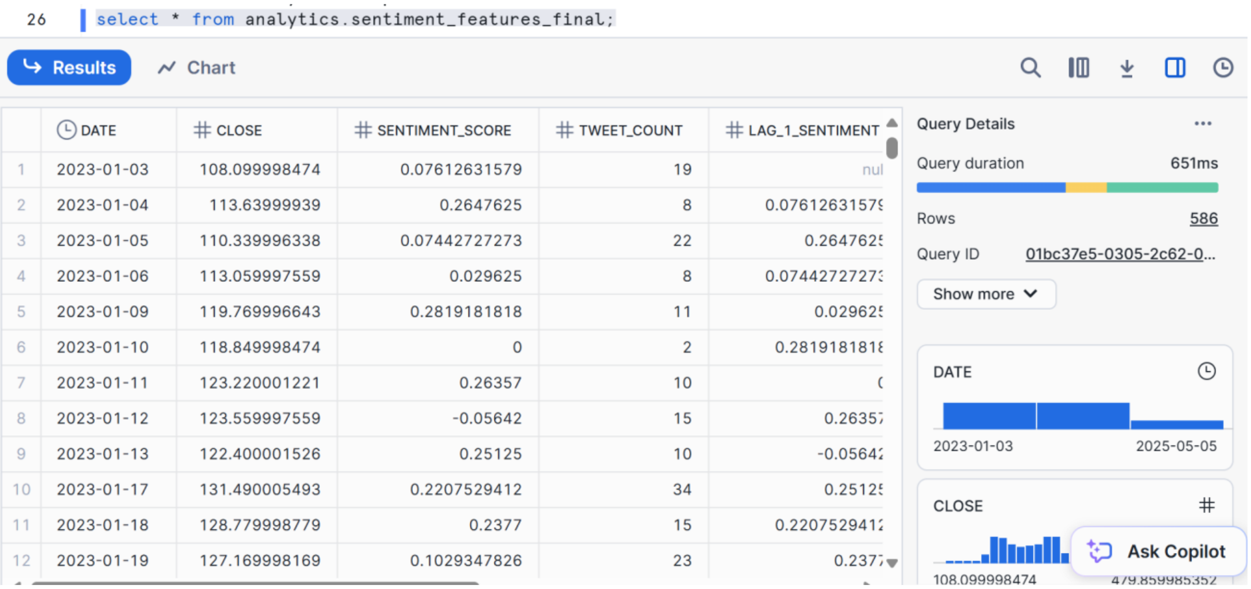
Figure 7: analytics.input



```sql
select * from analytics.output;
```

	DATE	CLOSE	PREDICTED_PRICE
1	2023-01-03	108.099998474	null
2	2023-01-04	113.63999939	112.690365359
3	2023-01-05	110.339996338	111.381833375
4	2023-01-06	113.059997559	110.283594887
5	2023-01-09	119.769996643	113.530294949
6	2023-01-10	118.849998474	113.27623013
7	2023-01-11	123.220001221	117.124254655
8	2023-01-12	123.559997559	115.468503665
9	2023-01-13	122.400001526	119.457421015
10	2023-01-17	131.490005493	123.515858888
11	2023-01-18	128.779998779	128.497360271
12	2023-01-19	127.169998169	127.428150163

Figure 8: analytics.output



```
select * from analytics.sentiment_features_final;
```

	DATE	CLOSE	SENTIMENT_SCORE	TWEET_COUNT	LAG_1_SENTIMENT
1	2023-01-03	108.099998474	0.07612631579	19	nul
2	2023-01-04	113.63999939	0.2647625	8	0.07612631579
3	2023-01-05	110.339996338	0.07442727273	22	0.2647625
4	2023-01-06	113.059997559	0.029625	8	0.07442727273
5	2023-01-09	119.769996643	0.2819181818	11	0.029625
6	2023-01-10	118.849998474	0	2	0.2819181818
7	2023-01-11	123.220001221	0.26357	10	(
8	2023-01-12	123.559997559	-0.05642	15	0.26357
9	2023-01-13	122.400001526	0.25125	10	-0.05642
10	2023-01-17	131.490005493	0.2207529412	34	0.25125
11	2023-01-18	128.779998779	0.2377	15	0.2207529412
12	2023-01-19	127.169998169	0.1029347826	23	0.2377

Query Details

Query duration 651ms

Rows 586

Query ID 01bc37e5-0305-2c62-0...

Figure 9: analytics.sentiment_features_final

5.4 Predictive Model

We build an updated linear regression model whose goal is to predict Tesla's daily closing stock price using sentiment-driven and market-based features. The model was trained using a linear regression algorithm and achieved a strong performance with an R^2 score of 0.916, indicating that over 91% of the variation in closing prices is explained by the features. The Root Mean Square Error (RMSE) of 19.03 and Mean Absolute Error (MAE) of 14.58 reflect relatively low average prediction error given the scale of Tesla's stock prices. The features used in the model include: SENTIMENT_SCORE, TWEET_COUNT, LAG_1_SENTIMENT, AVG_3D_SENTIMENT, and AVG_7D_CLOSE. The most impactful coefficient was for AVG_3D_SENTIMENT ($\beta \approx 28.56$), indicating that short-term sentiment trends significantly influence stock prices. The model intercept (β_0) is -6.90. All features were selected based on domain relevance, and the model was evaluated using a time-based train-test split to preserve the integrity of the time series data. This high R^2, combined with low RMSE and MAE, justifies the model's effectiveness for capturing both sentiment and market-driven signals in predicting Tesla's closing stock price.

1. Prediction Logic and Formula used instead of snowflake inbuilt ML

We trained a linear regression model externally (e.g., in Colab) and then manually implemented the model's coefficients within SQL code in dbt. This method allows you to perform predictions directly in Snowflake, leveraging the database's processing capabilities without relying on external machine learning services.

2. Predicted Close Price

This is the model's estimate of how Tesla's price is expected to change based on:

- Tweet sentiment
- Tweet volume
- Market trends (e.g., average closing prices)
- Lagged sentiment from previous days

3. Trained a Linear Regression model in Python on historical data

using features:

- sentiment_score

- tweet_count
- lag_1_sentiment
- avg_3d_sentiment

4.Linear Regression Equation: Coefficients and Intercepts

Table 1: Coefficients and Intercepts (Intercept (β_0): -6.904662)

Feature	Coefficient	Interpretation
LAG_1_SENTIMENT	(β): 4.921648	Strong positive effect → yesterday's sentiment increases today's price change
AVG_3D_SENTIMENT	(β): 28.557330	Strong inverse → higher recent sentiment averages may precede drops
SENTIMENT_SCORE	(β): 6.224183	Small but negative correlation — could indicate public mood is lagging, not leading
TWEET_COUNT	(β): 0.002950	
AVG_7D_CLOSE, VOLUME	(β): 1.016338	

VI. BI (Superset) & Visualization

Chart 1: Sentiment Data Time Trend Analysis

Purpose:
This line chart compares Tesla's stock closing prices with the 3-day moving average sentiment scores derived from Elon Musk's tweets, providing insights into the relationship between sentiment and stock price movements.

Usage:
The 3-day moving average helps to smooth out short-term fluctuations in sentiment data, making it easier to identify longer-term trends and correlations with stock price movements.

Dataset: analytics.sentiment_features_final

Chart 1: Average_3D_Sentiment

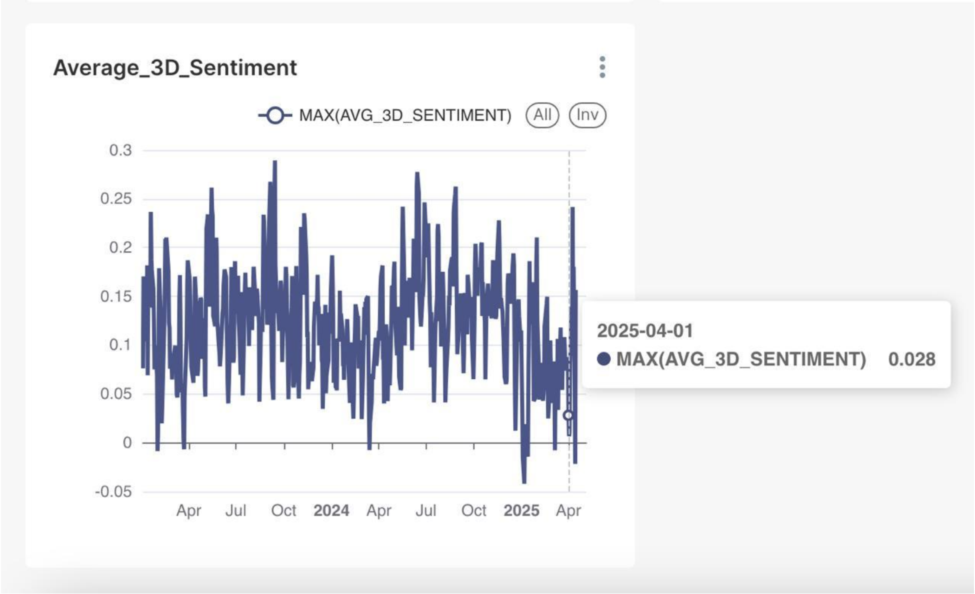


Chart 2: Sentiment and Tesla’s Stock Close Price Correlation Analysis

Purpose: This chart compares the 3-day moving average sentiment from Elon Musk's tweets with Tesla's stock price over the past 7 days, offering insights into the relationship between sentiment and stock price.

Usage: It highlights how short-term sentiment trends (3-day average) correlate with stock price movements over a 7-day period, helping to identify potential patterns.

Dataset: analytics.sentiment_features_final

Chart 2: 3-days sentiment score vs 7-days moving average

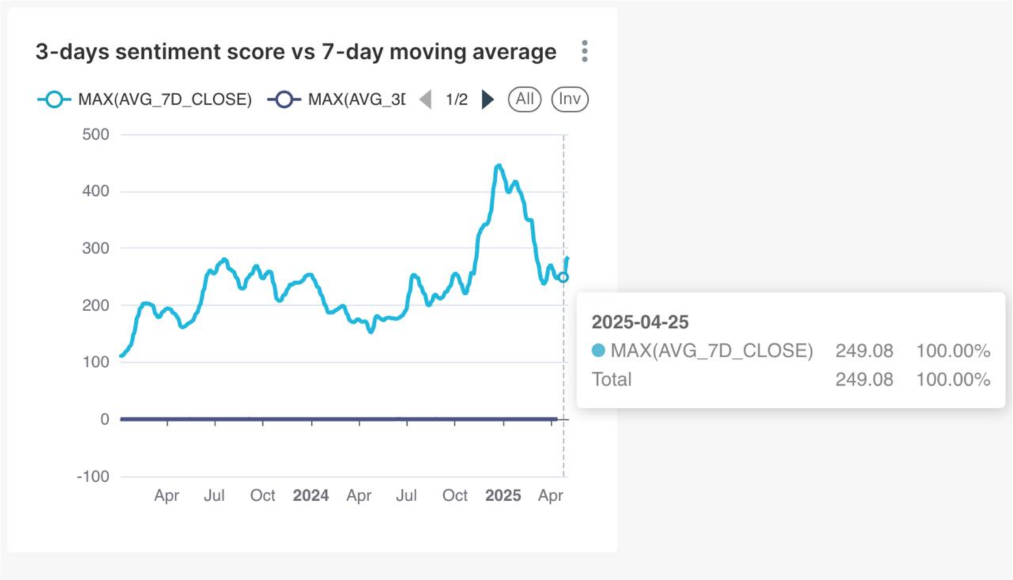


Chart 3: Crowd Interest VS Market Activity

Purpose: This chart compares the maximum tweet count (representing crowd interest) with the maximum trading volume (representing market activity) to explore their relationship.

Usage: It helps to analyze how peaks in public interest, as measured by tweet volume, correlate with market activity, represented by trading volume, to identify trends in how crowd interest influences stock market behavior.

Dataset: analytics.sentiment_features_final

Chart 3: Crowd Interest vs Market Activity

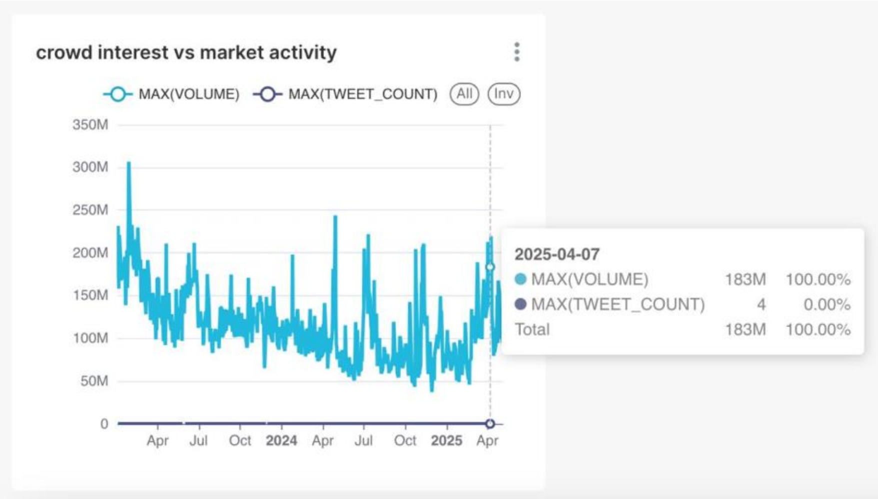


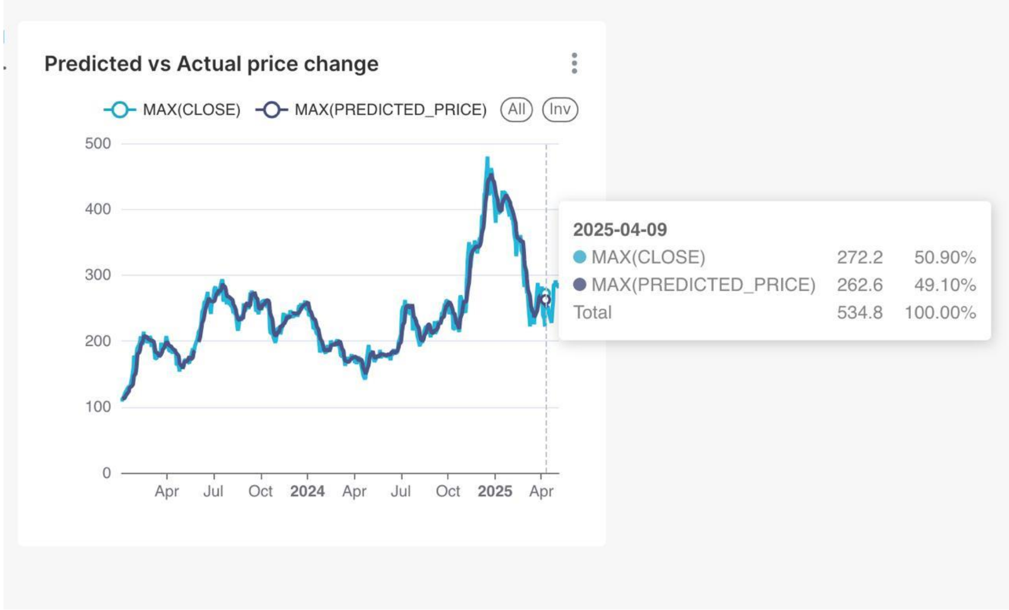
Chart 4: Tesla Stock Close Price Prediction

Purpose: To predict Tesla's stock closing prices based on historical data and sentiment analysis, illustrating potential future price movements.

Usage: This line chart helps visualize the predicted closing prices alongside the actual stock closing prices, allowing for easy comparison of the forecast accuracy and identifying trends or patterns in Tesla's stock price behavior. The chart aids in understanding the impact of sentiment and other factors on the stock's future performance.

Dataset: analytics.output

Chart 4: Predicted VS Actual Price Change



2025-04-09

MAX(CLOSE)	272.2	50.90%
MAX(PREDICTED_PRICE)	262.6	49.10%
Total	534.8	100.00%

***** Our Superset is local so cannot share using a public link, please refer to the screenshots. Refer to github in the next part repo to view the dashboard screenshot.

VII. Implementation

We use SQL transaction along with try/catch to implement idempotency. In dbt part we implemented models, tests, and snapshot, integrating dbt models in airflow.

Github url: https://github.com/Lakshmibharathy11/Elon_musk-s_tweet_impact

VIII. Analysis and Recommendation

8.1 Key findings

Negative Sentiment Leads to Price Decline

When sentiment shifts negatively, whether due to tweets, Tesla's stock typically experiences a dip the following day. This suggests that negative sentiment can act as a catalyst for short-term price drops, as market participants react to the potential risks associated with unfavorable tweets.

Positive Sentiment Drives Modest Price Increase

Positive sentiment does lead to price increases, but the effect is relatively small. This indicates that while optimism can influence stock movements, other larger market forces continue to have a stronger impact on stock price dynamics.

Delayed Reaction to Sentiment Changes

The most significant price movements are observed the day after the sentiment shift, indicating a delayed market reaction. This delay suggests that traders and investors may need some time to process new information and adjust their strategies accordingly, rather than reacting instantaneously.

8.2 Recommendation

Sentiment as a Predictor: While sentiment analysis can serve as a useful predictor of stock price movements, it is not sufficient on its own. Other external factors—such as broader market trends and economic indicators—remain essential for accurately predicting price changes.

Efficiency in Data Management: Conducting data cleaning, testing, and modeling within Snowflake using dbt has proven to be an efficient approach. This method not only kept the workflow organized but also ensured that the pipeline was repeatable and scalable.

Data Provenance and Transparency: The use of versioned snapshots and data lineage in the pipeline is invaluable for maintaining the integrity and traceability of the analysis. This practice ensures that each figure in the model can be traced back to its source, providing a clear audit trail—whether for validation purposes or compliance.

IX. Conclusion

In this project, we explored the correlation between tweets sentiment and Tesla’s stock price fluctuations by building a data pipeline using Airflow and Snowflake. We extracted stock data from Yahoo Finance and Elon Musk’s tweets from twitter, performed sentiment classification using Snowflake's machine learning capabilities, and integrated the results with stock prices for analysis. Our correlation analysis indicated a measurable relationship between sentiment and short-term price movements, where negative tweets often aligned with price drops. A line chart visualization and predictive model further demonstrated that sentiment could offer early signals of price direction. While our results show promise, future improvements could include integrating more granular real-time data, testing additional sentiment models, and expanding analysis to multiple stocks or sectors for broader applicability.

References

[1] Elon Musk’s tweet sentiment data. Available: https://twitter.com/elonmusk.
[2] Yahoo Finance, "Tesla Inc. (TSLA) Stock Historical Data." Available: https://finance.yahoo.com/quote/TSLA/history.
[3] yfinance package, "Yahoo Finance API for stock data retrieval." Available: https://pypi.org/project/yfinance/.
[4] Elon Musk tweets dataset, "all_musks.csv," collected 2010–2025.
[5] H. P. Liu, "VADER Sentiment Analysis on Tweets," Google Colab. Available: https://colab.research.google.com.