

From Tweets to Trends

Predicting Stock Volumes Using X Sentiment

Spring 2025

Team Members

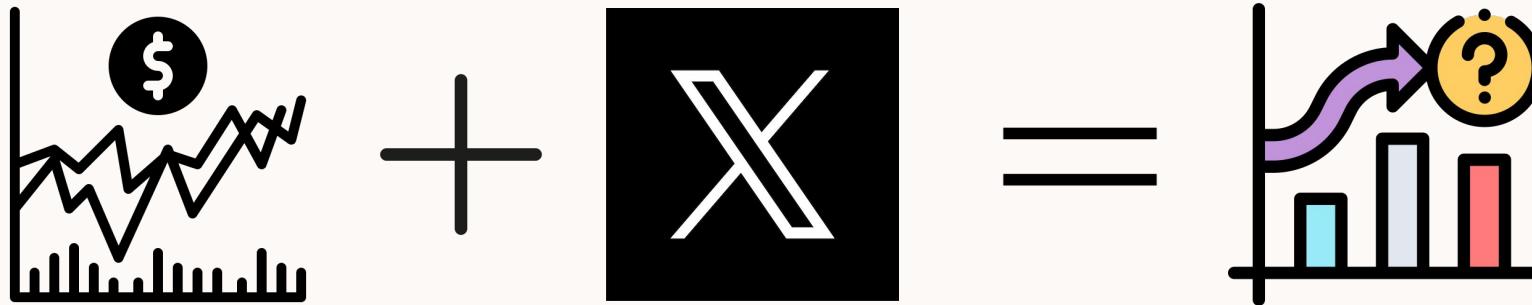
Yuchen Li

Syeda Aqeel

Hsiang Yu Huang

Introduction

- This project aims to predict stock price movements based on public sentiment expressed on X



Business Applications

- **Smart Trading**
Leverages real-time sentiment to enhance algorithmic strategies, especially for fast-moving stocks like NVIDIA.
- **Early Risk Detection**
Detects market panic signals through tweet tone shifts—supporting timely hedging.
- **Strategic Investment Insights**
Identifies patterns in public mood and key influencers shaping investor sentiment.
- **Reputation Management**
Helps NVIDIA refine communication by tracking sentiment dips tied to events (e.g., DeepSeek launch reaction).
- **Influencer Targeting for Marketing**
Reveals which verified voices move the market—useful for brand partnerships or PR.
- **R&D and Academic Value**
Offers a scalable blueprint for future applications across fintech and behavioral finance.
- **Real-Time, Low-Cost Forecasting**
Automated insights without human bottlenecks—ideal for daily operational decision-making.

Why NVIDIA? Why Stock Volume

- Frequent Social Media Attention
 - DeepSeek's Launch made NVIDIA stock dropped

Why not stock price

- Non-stationary
- highly volatile nature

yahoo/finance

Nvidia stock plummets, loses record \$589 billion as DeepSeek prompts questions over AI spending

NVIDIA SUFFERS WORST SINGLE-DAY MARKET CAP LOSS IN HISTORY

Largest single-day market capitalization losses in U.S. stock market history

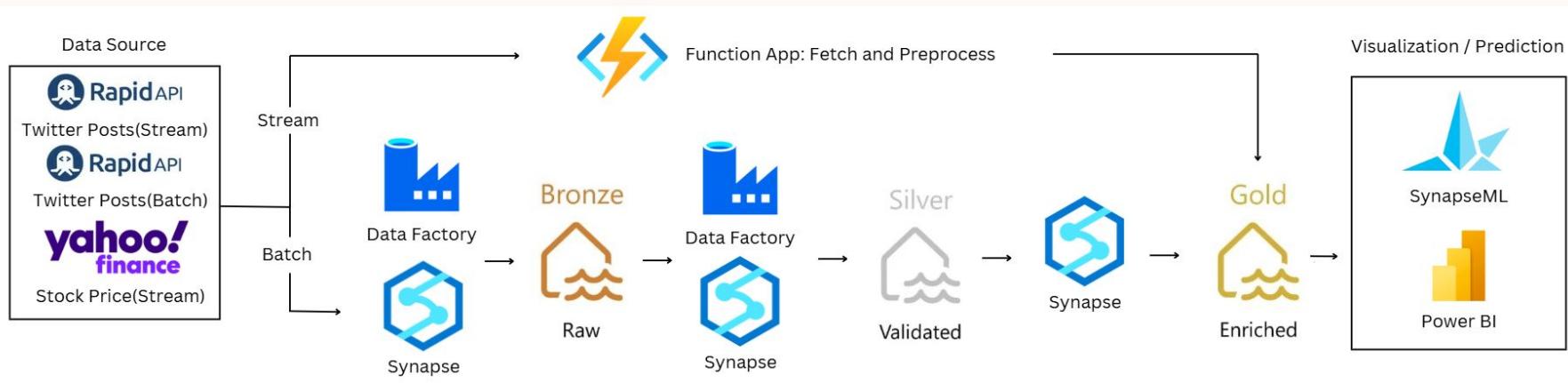
DATE	COMPANY	MARKET CAP LOSS
January 27, 2025	Nvidia	\$589B
September 9, 2024	Nvidia	\$279B
February 3, 2022	Meta	\$251B
January 7, 2025	Nvidia	\$228B
April 19, 2024	Nvidia	\$212B
June 24, 2024	Nvidia	\$208B
April 29, 2022	Amazon	\$206B
July 17, 2024	Nvidia	\$206B
July 24, 2024	Nvidia	\$205B
January 27, 2025	Broadcom	\$200B
April 29, 2024	Nvidia	\$197B
October 31, 2024	Microsoft	\$195B

SOURCE: YAHOO FINANCE

yahoo/finance

Project Architecture

- Medallion Structure (Bronze → Silver → Gold)
 - Modular, Scalable Processing



Data Sources

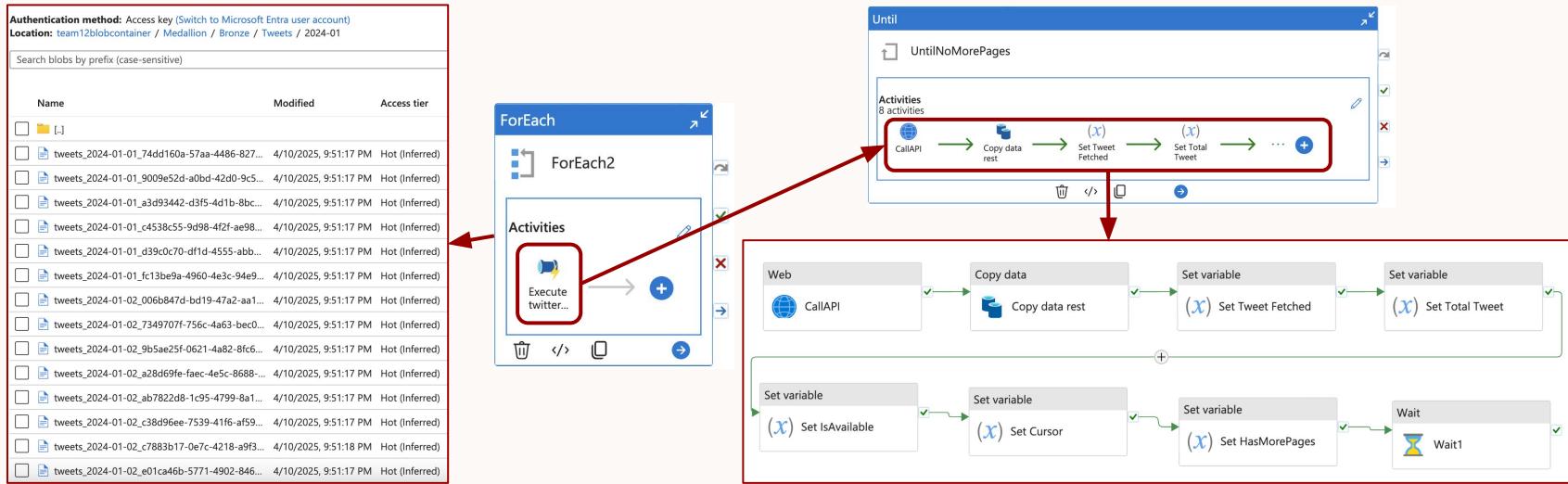
- Tweets via RapidAPI's Tweet V2 API (Batch + Streaming)
- Stock data via Yahoo Finance (`yfinance` in Synapse)



Batch Processing (Tweet)

Bronze (via ADF)

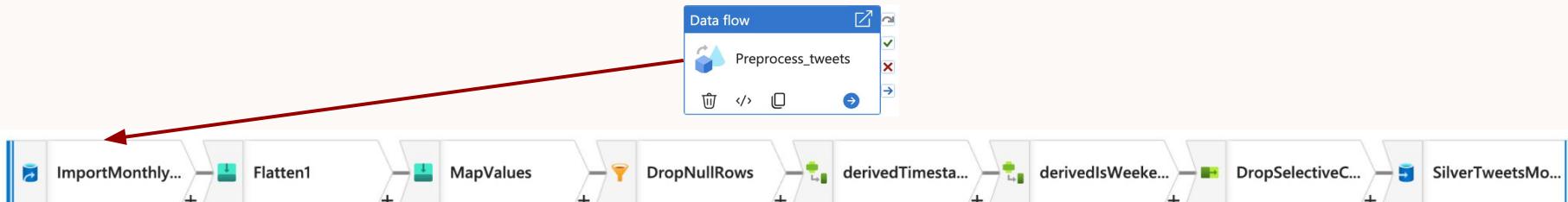
- Collect daily tweets with hashtag #\$NVDA via RapidAPI
- Use bottom cursor for pagination across multiple calls
- ForEach Loop → Until Loop → Stores JSON output to Blob Storage (Bronze Layer)



Batch Processing (Tweet)

Silver (via ADF Data Flow)

- Executed in monthly units
- Read raw JSON tweet files from Bronze Layer
- Steps Performed
 - Flatten for nested fields → Map Values → Drop Null Rows → Derive New Columns → Drop Redundant Columns
- Output: Cleaned dataset (20 columns)
 - Store in Silver Layer



Stock Pipeline (Bronze)

Bronze (via Synapse)

- Collect daily stock information through python library finance
- Set Ticker as “NVDA”
- No data on non-market day

```
1 import yfinance as yf
2 import pandas as pd
3 from datetime import datetime, timedelta
4 # Pull data from Yahoo Finance
5 ticker = "NVDA"
6 start_date = "2024-01-01"
7 end_date = (datetime.strptime("2025-04-28", "%Y-%m-%d") + timedelta(days=1)).strftime('%Y-%m-%d')
8 df = yf.download(ticker, start=start_date, end=end_date)
```

✓ - Command executed in 250 ms on 7:03:32 PM, 5/02/25

	Price	Date	Close	High	Low	Open	Volume
Ticker			NVDA	NVDA	NVDA	NVDA	NVDA
0	2024-01-02	48.149918	49.276493	47.577135	49.225514	411254000	
1	2024-01-03	47.551144	48.165911	47.302237	47.467176	320896000	
2	2024-01-04	47.979984	48.481795	47.490167	47.749068	306535000	

Stock Pipeline (Silver)

Silver (via Synapse)

- Check and drop Duplicate rows
- Transform Date column to string time format, align with Tweet date format
- Save as Parquet format

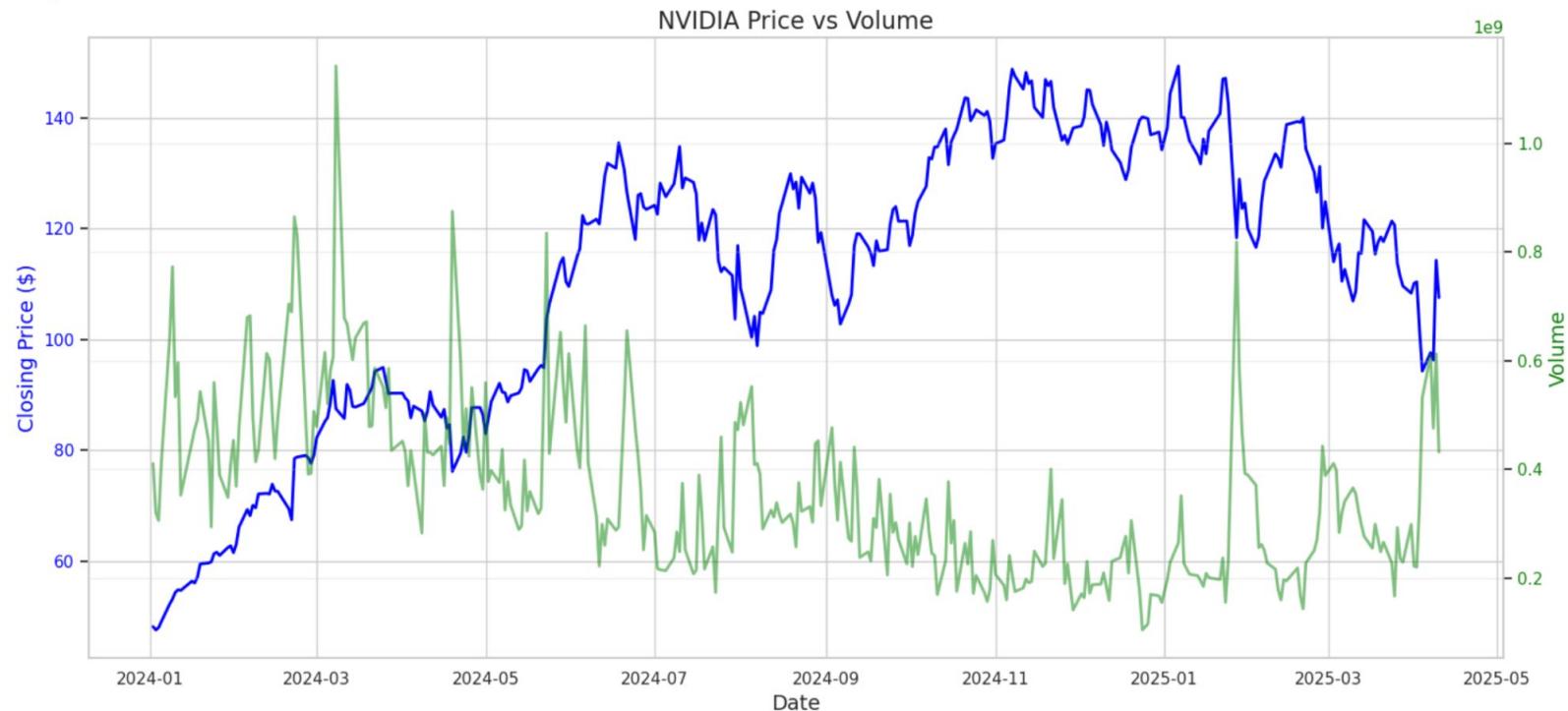
```
Index(['Date', 'Close', 'High', 'Low', 'Open', 'Volume'], dtype='object')
      Date      Close      High      Low      Open      Volume
0  2024-03-01 00:00:00    82.248108    82.269104   79.405174   79.969964   479135000
1  2024-03-04 00:00:00    85.205002    87.662079   83.687574   84.098415   615616000
2  2024-03-05 00:00:00    85.935760    86.068720   83.389600   85.241988   520639000
3  2024-03-06 00:00:00    88.670860    89.694526   87.001411   87.993089   582520000
4  2024-03-07 00:00:00    92.638550    92.736517   89.572556   90.128373   608119000
```



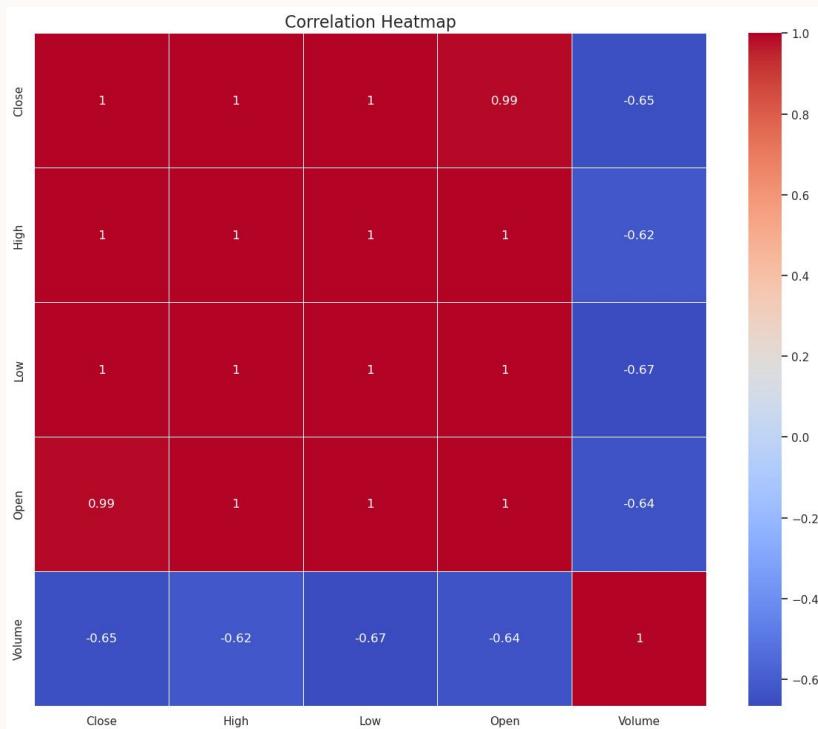
part-00000-1e09fdd7-b250-4929-b911-f38458a8deac-c000.snappy.parquet

Exploratory Data Analysis

sharp drop in NVIDIA's stock price and surge in trading volume in early 2025 aligns with the market's reaction to DeepSeek



Correlation Between NVIDIA Stock Metrics



--- Summary of Stock Performance ---

Period: 2024-01-02 00:00:00 to 2025-04-10 00:00:00

Starting Price: \$48.15

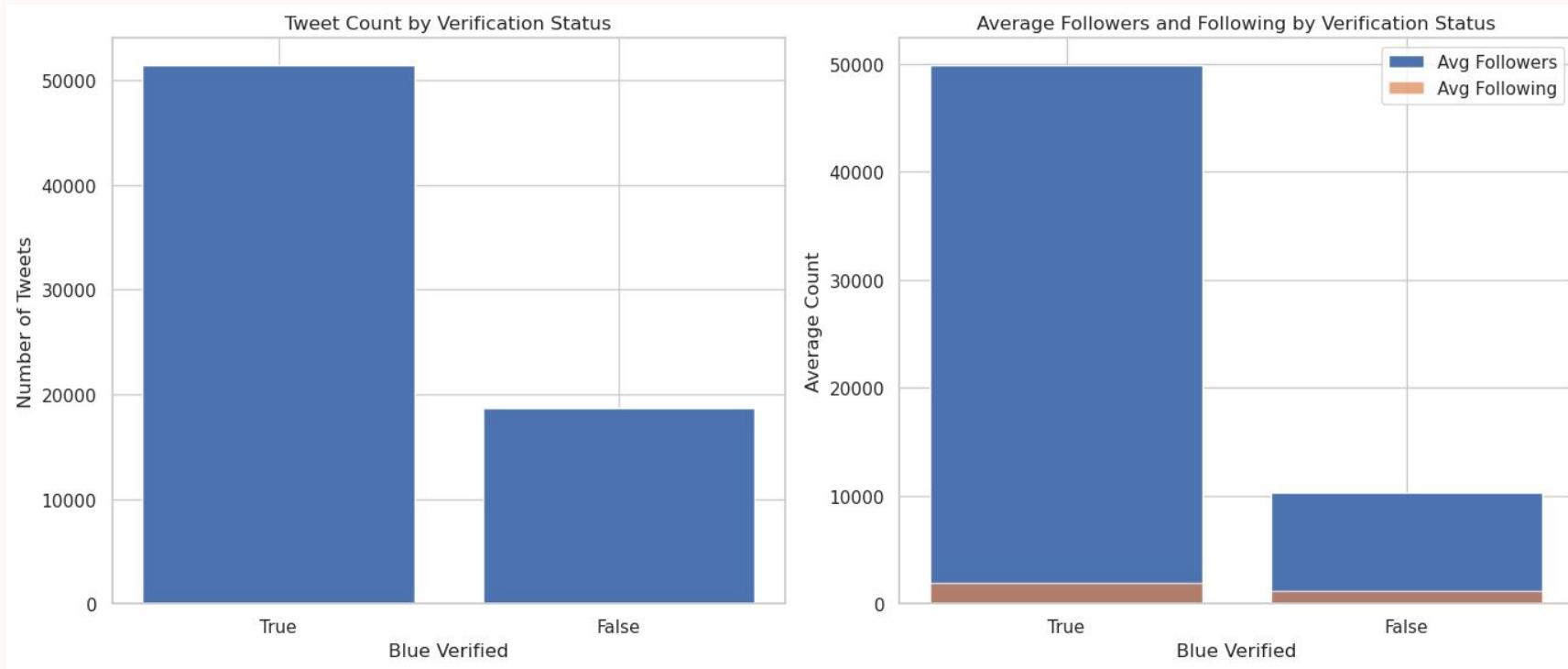
Ending Price: \$107.57

Overall Price Change: 123.41%

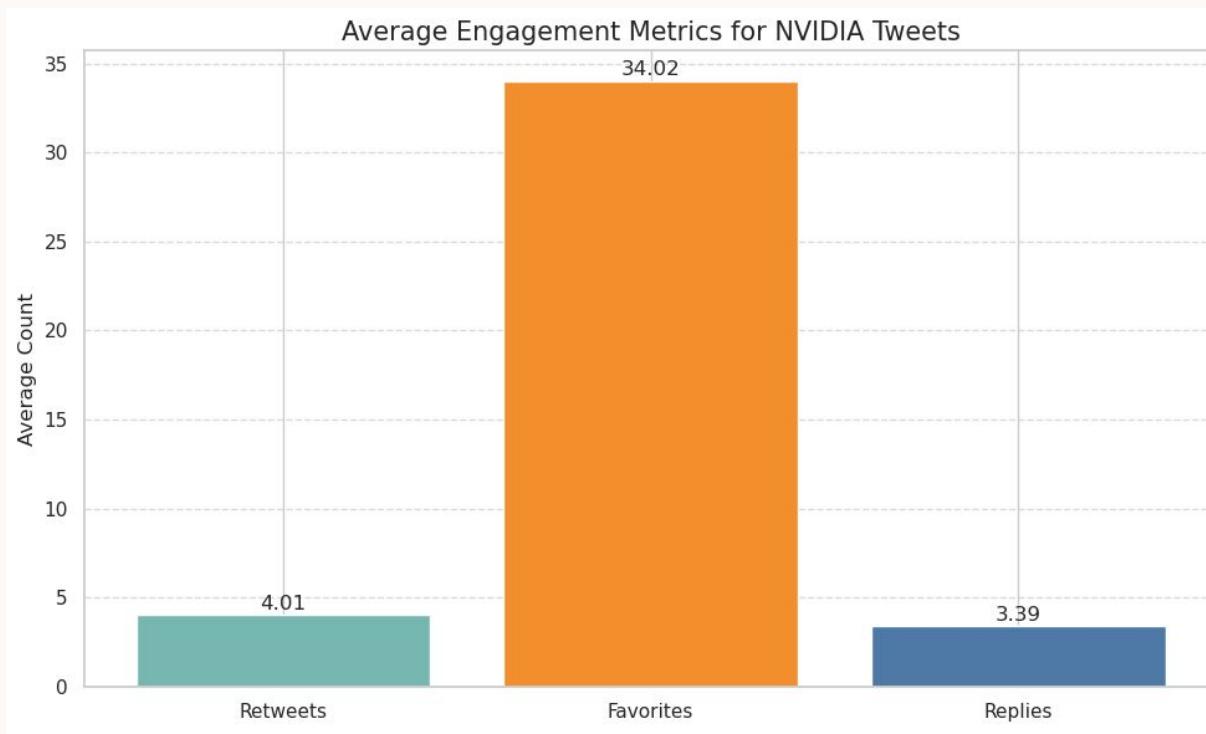
Average Trading Volume: 360698947

Maximum Volume: 1142269000

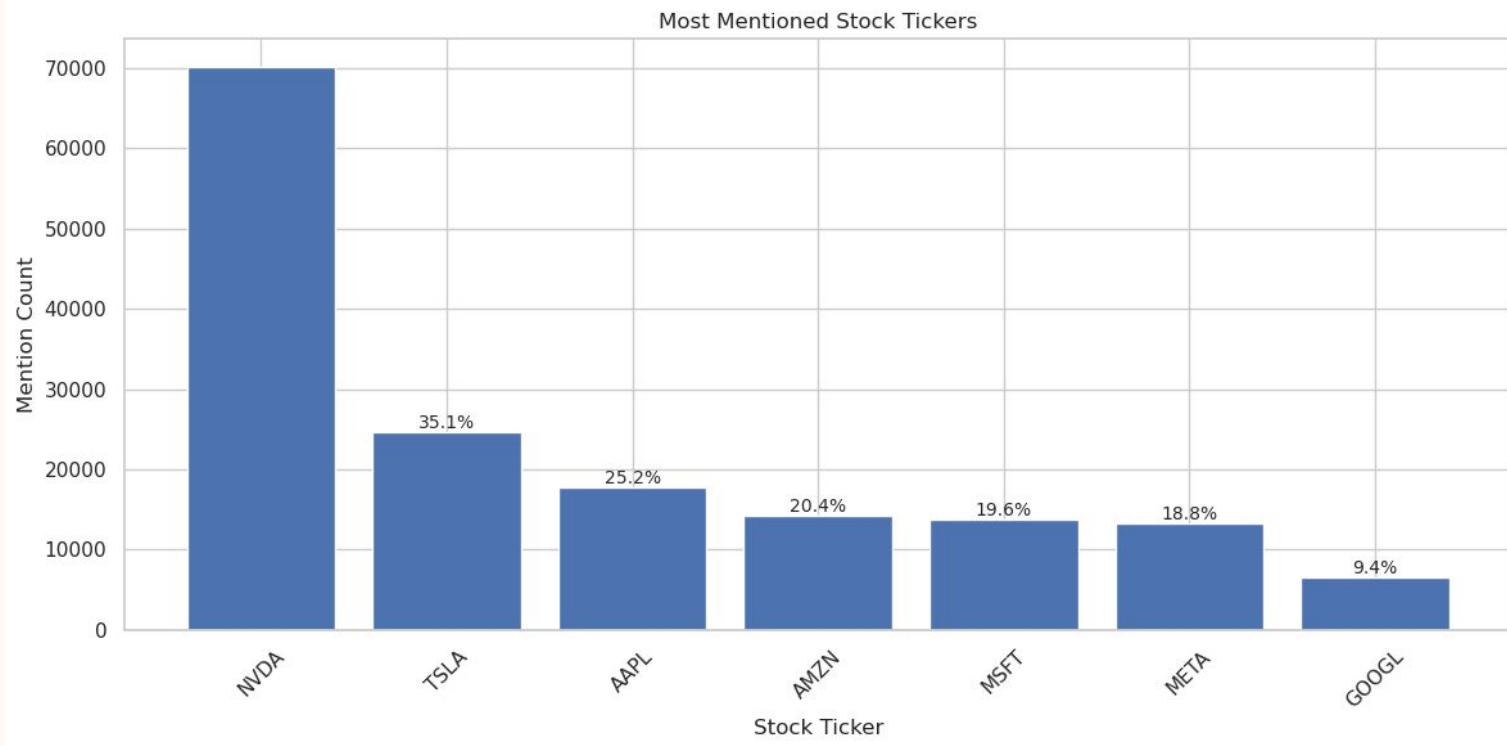
Engagement and Reach of Blue-Verified Users Tweeting About #NVIDIA



Favorites Drive Engagement on #NVIDIA Tweets, Outpacing Retweets and Replies



Most Co-Mentioned Stock Tickers in #NVIDIA Tweets



Gold Merge (Tweet + Stock)

Gold Layer (via Synapse Spark)

- Perform Sentiment Analysis with VADER
- Feature Engineering
 - Engagement Scope
 - User Metrics
 - Flags
- Join & Output
 - Merge two datasets on date
 - Add next_available_volume (target for ML)
 - Output to Gold/dataset_updated/

The screenshot shows two code snippets in the Microsoft Azure Synapse Analytics workspace:

Top Snippet (TweetStockGold):

```
1 import re
2 From pyspark.sql.functions import udf, col
3 From pyspark.sql.types import StringType, DoubleType
4 import nltk
5 from nltk.sentiment import SentimentIntensityAnalyzer
6
7 # Download VADER (only once needed)
8 nltk.download('vader_lexicon')
9
10 # Initialize VADER
11 sia = SentimentIntensityAnalyzer()
12
13 # Clean text function (remove URLs and extra spaces)
14 def clean_text(text):
15     if text:
16         text = re.sub("https://\S+", "", text) # Remove any URL
17         text = re.sub("\s+", ' ', text).strip() # Remove extra spaces
18     return text
19
20     return None
21
22 # Create UDF for cleaning
23 clean_text_udf = udf(clean_text, StringType())
24
25 # (1) Overwrite full_text column with cleaned text
26 tweet_df = tweet_df.withColumn("full_text", clean_text_udf.col("full_text")))
27
28 # (2) Define UDF for sentiment analysis
29 @udf(returnType=DoubleType())
30 def polarity_score(text):
31     if text:
32         score = sia.polarity_scores(text)[“compound”]
33     else:
34         score = 0.0
35
36     return score
37
38 # Create New Column
```

Bottom Snippet (TweetStockGold):

```
1 # Step 1: Perform the left join
2 tweet_df = tweet_df.join(
3     stock_df,
4     col("date"), alias("match_date"),
5     col("volume"), alias("current_volume"),
6     col("open"), alias("current_open"),
7     col("close"), alias("current_close"),
8     col("high"), alias("current_high"),
9     col("low"), alias("current_low")
10    ),
11    tweet_df[“tweet_created_at”] == col(“match_date”), how="left"
12 )
13
14
15 # Step 2: Clean up (remove the helper match_date column)
16 tweet_df = tweet_df.drop("match_date")
17 display(tweet_df.limit(10))
```

Below the code snippets, a table view shows the resulting dataset with columns: user_id, is_blue_verified, account_created_at, followers_count, friends_count, and account_favourites_count. The data includes rows for various users with their respective details.

user_id	is_blue_verified	account_created_at	followers_count	friends_count	account_favourites_count
VXNlclg01NjISOTUSQte+	true	2012-02-25	44327	169	5660
VXNlclg01NjM1Ng2q07Aw	true	2013-07-31	16558	605	8255
VXNlclg01NjM2aBUD6H6NTQwO...	true	2023-04-04	36346	182	2664
VXNlclg0xMDY0NzAwMuA4...	true	2013-01-06	104852	2352	51833
VXNlclg0xMzE3MDA4MTI1Nz20...	true	2020-10-16	5216	217	3343

Gold Aggregate for Visualization

- **User Profile Table:** Captures metrics like follower count, credibility, influencer flags
- **Daily Tweet Summary:** Avg. sentiment, engagement totals, viral count by date
- **Market Summary per Day:** Merges tweet stats with daily stock performance
- **Viral Tweet Log:** Stores all high-impact viral tweets
- **User Type Distribution:** Verified vs. Unverified, Influencer vs. General, New vs. Established
- **Top Users by Tweet Count / Viral Tweets:** Highlights most active and influential contributors
- **Influencer Monthly Engagement:** Tracks influencers' impact over time

```
1 from pypark.sql.functions import max as spark_max, avg, first
2 user_profile_df = df.groupby("user_id").agg(
3     first("is_blue_verified").alias("is_blue_verified"),
4     first("account_created_at").alias("account_created_at"),
5     spark_max("account_age_days").alias("account_age_days"),
6     first("is_new_account").alias("is_new_account"),
7     first("is_influencer").alias("is_influencer"),
8     spark_max("friends_count").alias("friends_count"),
9     spark_max("account_favourites_count").alias("account_favourites_count"),
10    spark_max("listed_count").alias("listed_count"),
11    spark_max("credibility_score").alias("credibility_score"),
12    avg("credibility_score").alias("credibility_score"),
13    avg("follower_activity_score").alias("follower_activity_score")
14 )
15 )
16
17 display(user_profile_df.limit(10))
```

View Table Chart Export results ↗

user_id	is_blue_verified	account_created_at	account_age_days	is_new_account	is_influencer
VXNljojNTYMTU0MTY2NzM4...	true	2022-08-23	985	false	true
VXNljojNTYMTw0MDx3MjU2RjyN...	true	2017-01-05	3041	false	true
VXNljojNTYMTxk100khj0j0M...	false	2022-09-07	970	false	false
VXNljojNTYODawQD1NTC=	true	2011-04-10	5138	false	true
VXNljojNTQ2ODY=	true	2012-11-07	4561	false	true

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[...]					-	...
daily_tweet_summary	5/5/2025, 6:56:01 PM				-	...
dataset_updated	5/5/2025, 5:16:20 PM				-	...
influencer_monthly_engagement	5/5/2025, 6:56:11 PM				-	...
market_summary_per_day	5/5/2025, 8:10:28 PM				-	...
Stream	5/4/2025, 7:00:08 PM				-	...
top_users_by_tweet_count	5/5/2025, 6:56:07 PM				-	...
top_users_by_viral_tweets	5/5/2025, 6:56:09 PM				-	...
user_profile	5/5/2025, 6:56:14 PM				-	...
user_type_distribution_per_day	5/5/2025, 6:56:05 PM				-	...
viral_tweet_log	5/5/2025, 6:56:02 PM				-	...

Stream Data

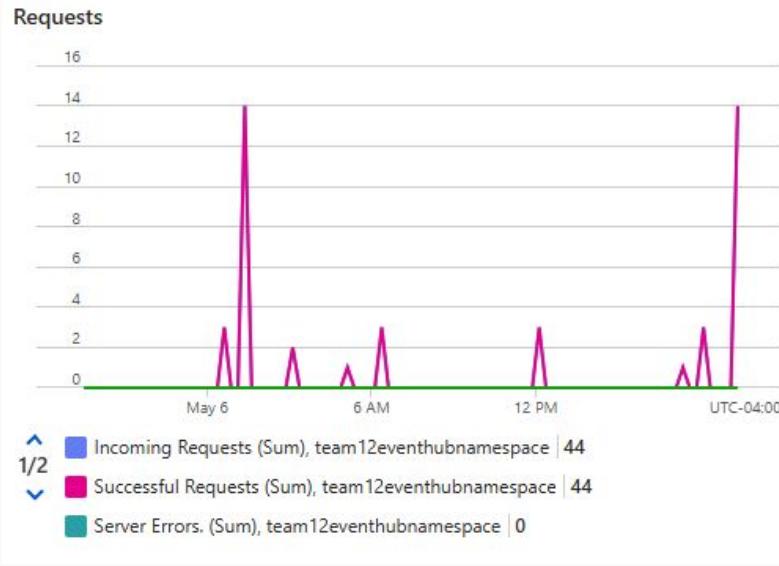
- Triggers Every day at 5am
- Process directly to Gold Layer Schema
- Save to Blob Storage
- Skip non-market day data

Why not Event Hub?

- Limited Message size of 1MB

```
@app.timer_trigger(  
    schedule="0 0 * * *",  
    arg_name="myTimer",  
    run_on_startup=False,  
    use_monitor=False  
)
```

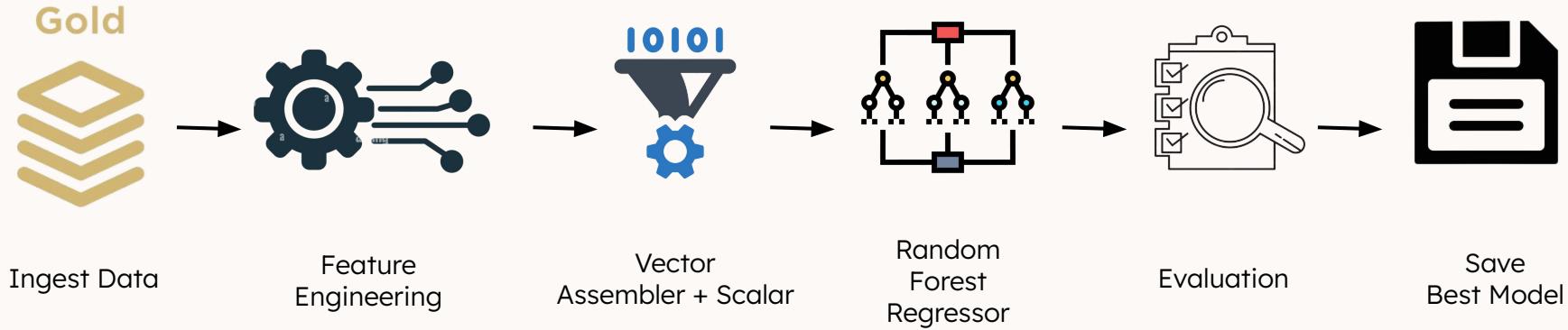
```
def save_to_blob(data: dict, date: datetime) -> bool:  
    """Save data to Azure Blob Storage"""  
    try:  
        date = date.strftime('%Y-%m-%d')  
        blob_path = f"Medallion/Gold/Stream/{date}.json"  
        data_json = json.dumps(data, indent=4)  
        blob_client = blob_service_client.get_blob_client(container=container, blob=blob_path)  
        blob_client.upload_blob(data_json, overwrite=True)  
        return True
```



Machine Learning Pipeline

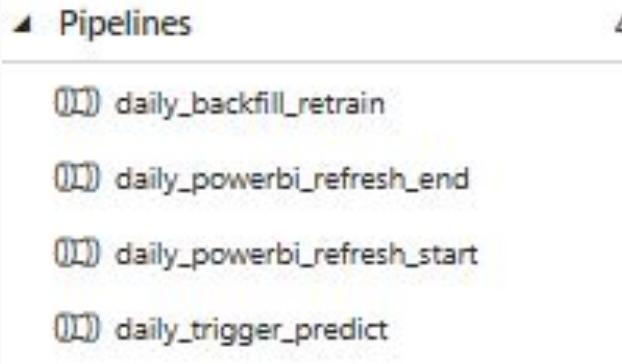


Goal: Predict next-available NVIDIA stock volume from tweet sentiment



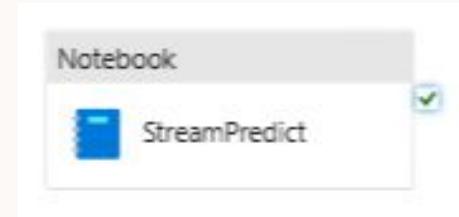
Daily Trigger Routine for Automatic Prediction

- Automated prediction pipeline for streaming data
- Starting from 5am everyday



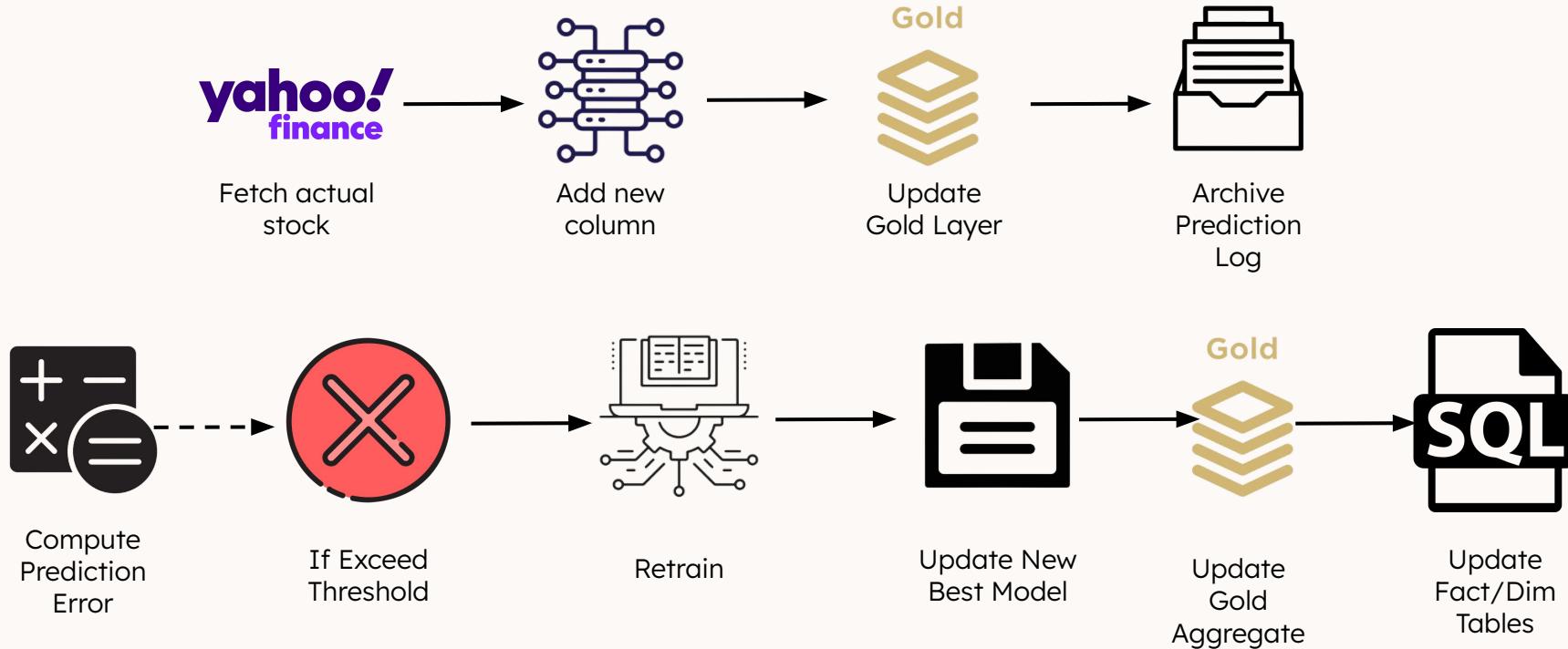
Daily Prediction Workflow

- **Input:** Loads streaming data from Blob Storage.
- **Preprocessing:** Transformation pipeline used for prediction
- **Prediction:** Loads best model and predicts for the end-of-day stock volume
- **Output:** Write to the “Prediction/Streaming” folder.



daily trigger	Schedule trigger	5/6/2025, 7:00:00 AM	Succeeded
daily_trigger_predict	5/6/2025, 7:00:02 AM	5/6/2025, 7:06:32 AM	6m 31s
2025-05-05		5/6/2025, 7:05:59 AM	

Backfill & Retraining Logic



daily backfill retrain Schedule trigger 5/6/2025, 6:30:01 PM Succeeded

[daily_backfill_retrain](#)

5/6/2025, 6:30:01 PM

5/6/2025, 6:54:46 PM

24m 45s

daily backfill retrain

Succeeded

SQL Integration & Pool Management

External Tables: Aggregated Gold Layer data

Fact & Dimension Tables: Built on top of external tables for PowerBI

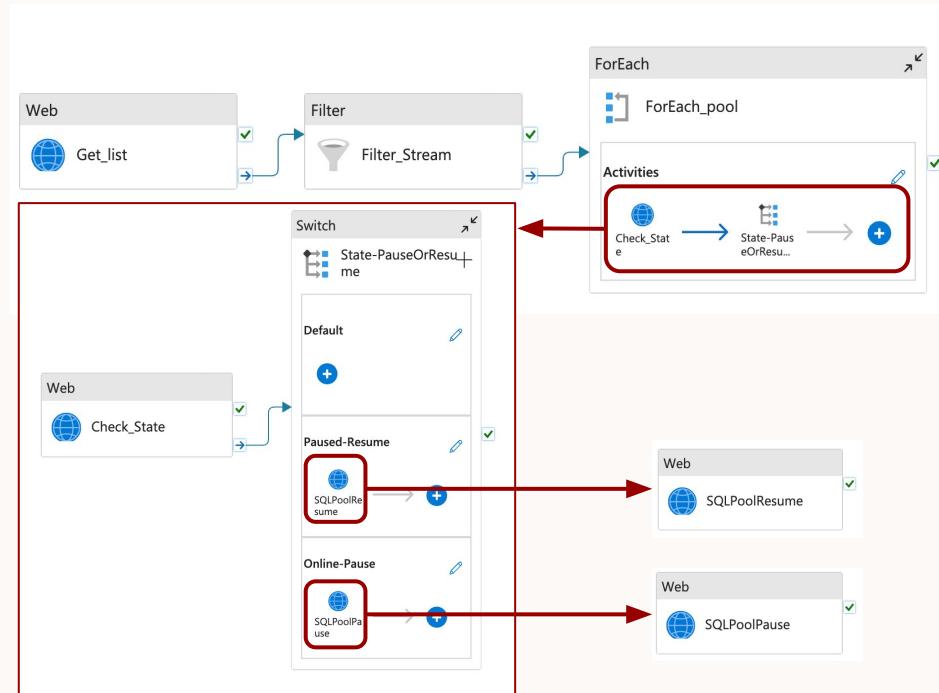
Stored Procedure - DailyRefresh:

- Truncates & Refreshes Fact Tables
- Cast data types as needed
- Applies transformations

Runs **Daily** to sync Power BI dashboards

```
1 --- FACT: Daily Tweet Summary
2 TRUNCATE TABLE FactDailyTweet;
3 INSERT INTO FactDailyTweet SELECT * FROM ExternalDailyTweet;
4
5 --- FACT: Viral Tweet Log
6 TRUNCATE TABLE FactViralTweetLog;
7
8 INSERT INTO FactViralweetLog
9 SELECT
10     user_id,
11     tweet_created_at_date,
12     CAST(full_text AS NVARCHAR(500)),
13     sentiment_score,
14     interaction_score,
15     followers_count,
16     favorite_count,
17     retweet_count,
18     reply_count,
19     CAST(view_count AS INT),
20     credibility_score
21 FROM Externalviralweetlog;
22
23 --- FACT: User Type Distribution
24 TRUNCATE TABLE FactUserTypeDistribution;
25 INSERT INTO FactUserTypeDistribution SELECT * FROM ExternalUserTypeDistribution;
26
27 --- FACT: Top Users by Tweet Count
28 TRUNCATE TABLE FactTopUsersByTweetCount;
29 INSERT INTO FactTopUsersByTweetCount SELECT * FROM ExternalTopUsersByTweetCount;
30
31 --- FACT: Top Users by Viral Tweets
32 TRUNCATE TABLE FactTopUsersByViralTweets;
33 INSERT INTO FactTopUsersByViralTweets SELECT * FROM ExternalTopUsersByViralTweets;
```

SQL Pool Management



daily_powerbi_refresh_end	5/6/2025, 6:53:00 PM	5/6/2025, 6:54:45 PM	1m 45s	0ef642e6-9f10-452b...	✓ Succeeded
daily_powerbi_refresh_start	5/6/2025, 6:39:20 PM	5/6/2025, 6:42:31 PM	3m 11s	47380450-2ea0-41b...	✓ Succeeded

PowerBI Dashboard

From Tweets to Trends: Predicting Stock Prices Using X Sentiment

< Filters

Current Stock Volume

236,121,500

Predicted Next Stock Volume

253,299,694

Actual Next Stock Volume

189,784,700

Prediction Accuracy

66.53%

Date

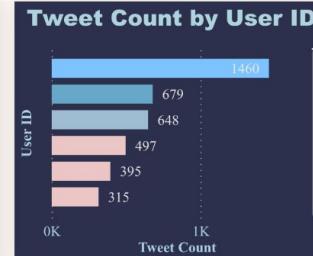
Search

2025-05-01

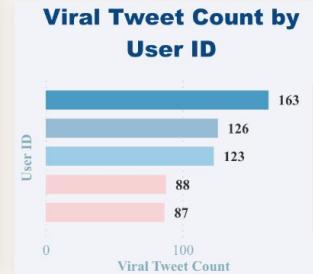
2025-04-30

2025-04-29

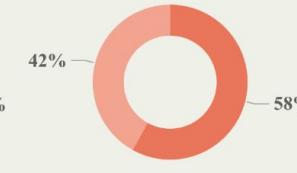
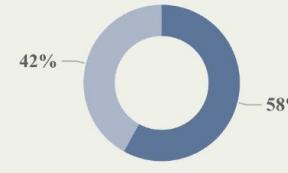
2025-04-28



Viral Tweet Count by User ID



Daily User Type Distribution



Favorite Ratio

Favorite / View

0.00854

0 0.0138

Reply Ratio

Reply / Retweet

1.72285

0 2.3118

Intersection Score

778.48

Sentiment Score

0.11

Viral Tweets

2

Credibility Score

18.05

Power BI Dashboard

From Tweets to Trends: Predicting Stock Prices Using X Sentiment

Search

Select all

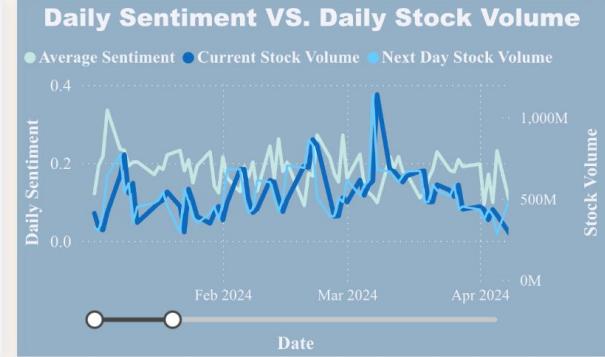
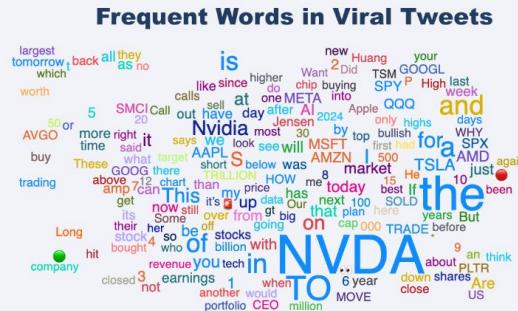
2025-05

2025-04

2025-03

2025-02

2025-01

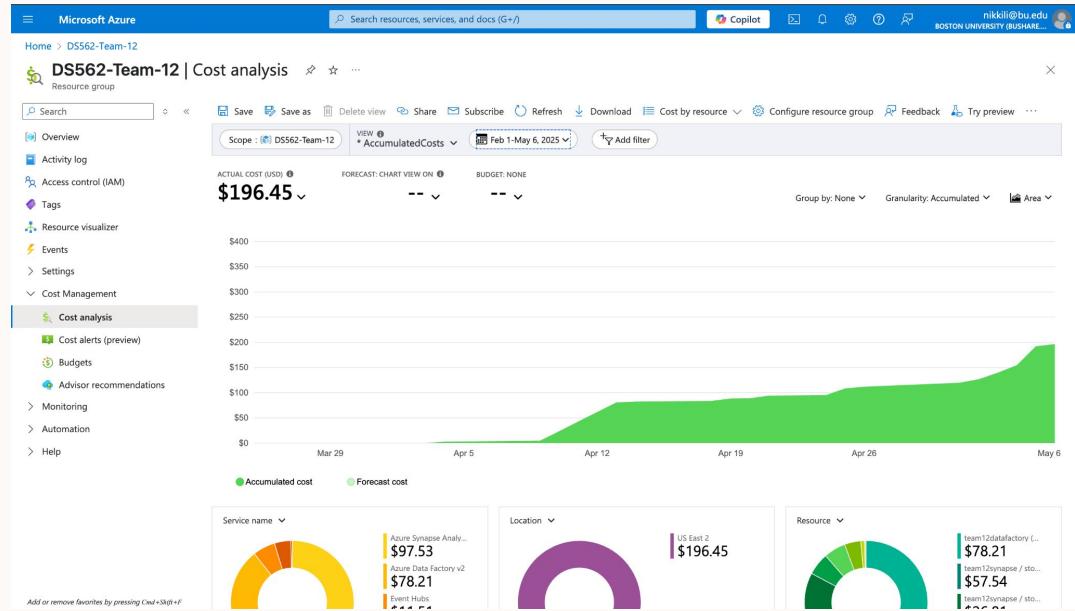


Viral Tweet

Tweet	Reply	Retweet	Like
You can only pick one to invest in for the next 10 years \$TSLA, \$NVDA, \$AAPL, or \$PLTR Which are you choosing?	810	57	1218
Wrong answers only Why was Nvidia \$NVDA down 10% today?	806	85	1573
NVIDIA STOCK WILL BE UNDER \$200 IN A WEEK MARK THIS TWEET \$NVDA	702	203	3832
Which company will be larger in 2030, \$TSLA or \$NVDA?	693	122	1777
WHO'S BUYING THE NEW TARIFF DIP MONDAY? \$NVDA \$153 to \$93 🚨 -39% DISCOUNT \$PLTR \$125 to \$73 🚨 -41% DISCOUNT \$SHOUD \$67 to \$34 🚨 -49% DISCOUNT \$SOFI \$18 to \$9 🚨 -51% DISCOUNT \$TSLA \$488 to \$237 🚨 -51% DISCOUNT \$UPST \$97 to \$34 🚨 -64%...	679	811	7251
To outperform \$NVDA - get on the #Bitcoin Standard.	529	2022	13613

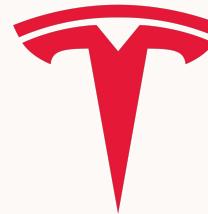
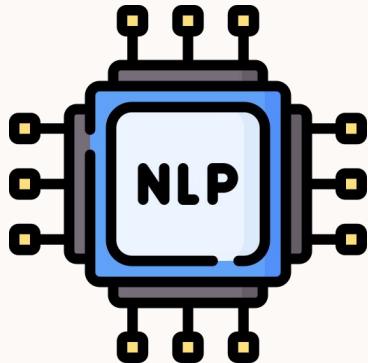
Challenges & Optimizations

- Data skew in modeling
- SQL pool cost management
- Feedback loop tuning
- Efficient API pagination



Future Work

- Weekend Tweet Utilization
- Complete Tweet Coverage
- Multi-stock analysis
- Anomaly detection
- Advanced NLP models



TESLA



Thank You!