Project Title: **Apple Insights: Linking News Sentiment with Stock Performance**

Group Members:

Name	Student-ID
Chuheng Xiao	**16356625**

Abstract
The stock market is deeply intertwined with public sentiment and major global events, reflecting a complex interplay of human psychology and market behavior. Analyzing stock prices solely based on historical and current values overlooks the broader influences that shape market dynamics. This project presents a preliminary analysis of historical news sentiment related to Apple Inc. and explores its potential impact on Apple's stock performance. By assessing sentiment scores of past news articles and correlating them with stock movements, this study lays the groundwork for more in-depth research on the effects of media sentiment on market behavior.

1. Introduction

a. Project Scope

- Converting unstructured data to structured format using NLP
- A preliminary exploration of the impact of sentiment analysis results on the stock market.

b. Project Limitations & Constraints

1. Limited News Sources: This project relies solely on Apple's official news archive, which lacks diversity and objectivity. While other sources, such as Yahoo News, were considered, financial constraints prevented the use of professional APIs needed to access them.

2. Simplified Analysis of Sentiment Impact on Stock Market: The exploration of sentiment analysis effects on the stock market is basic. This limitation arises because the project prioritizes large-scale data collection and storage, laying the groundwork for deeper stock market analysis in the future.

c. Feasibility Study [Technical | Operational]

1.Technical Feasibility

Google Cloud Platform offers a wide range of services for securing, storing, serving, and analyzing data[1]. This project employs big data processing and sentiment analysis techniques, leveraging Google Cloud Platform (GCP) services for efficient data storage and processing. Data is collected from Apple's news archive, and natural language processing (NLP), which stands halfway between computer science and computational linguistics, is used to evaluate the sentiment trends in the news over time. NLP is dedicated to the conversion of written and spoken natural human languages into structured, mineable data[2]. Through GCP's scalability and computational power, the technical requirements of the project are met effectively, enabling large-scale data processing and the analysis of potential links between news sentiment and the stock market.

Positive sentiment shocks are believed to enhance consumer confidence, increase consumption, output, and interest rates, while dampening inflation[3]. As such, this project aims to explore how shifts in news sentiment influence stock market fluctuations, uncovering potential relationships between sentiment trends and economic indicators.

2.Operational Feasibility

From an operational perspective, the project environment is feasible for users familiar with cloud platforms and NLP techniques. GCP ' s Dataproc clusters support large-scale data processing, and if access to other news sources' APIs can be secured in the future, the project ' s architecture can accommodate similar sentiment analysis tasks.

2. System Requirement Specifications (SRS) [MDRE | Bespoke]

1.System Overview

The system consists of components for data collection, storage, sentiment analysis, and visualization.

Functional Requirements

2.Data Collection:

Collect data from Apple Inc. news archives, with future support for other sources.

3.Data Storage:

Use Google Cloud Storage to store news articles and sentiment scores.

Sentiment Analysis: Calculate sentiment scores for each news article.

4.Data Processing:

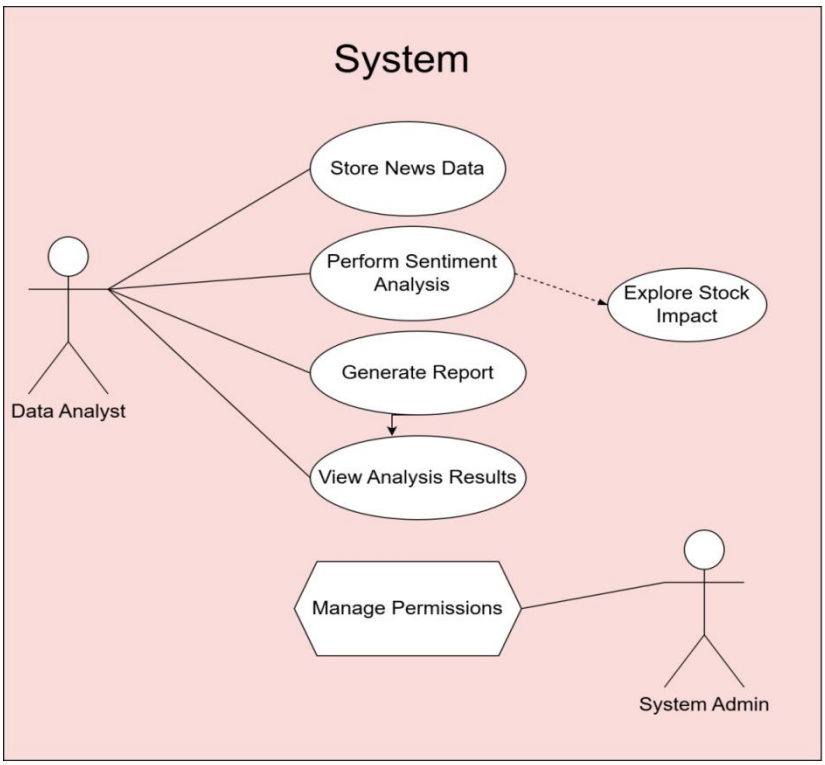
Integrate with Apache Spark to ensure efficient data analysis.

5.Data Visualization:

Provide visual displays of sentiment scores and stock price changes.

3. System Design

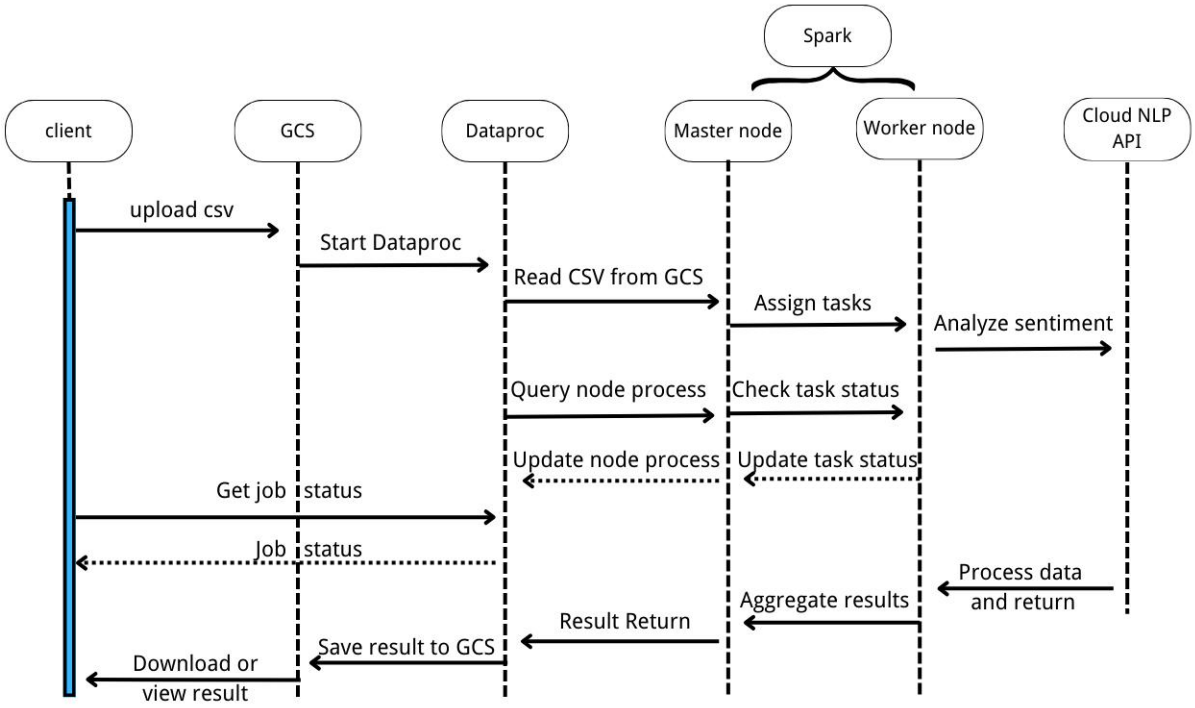
a. Use-Case Diagram



Possible Actors in Data Science Projects:

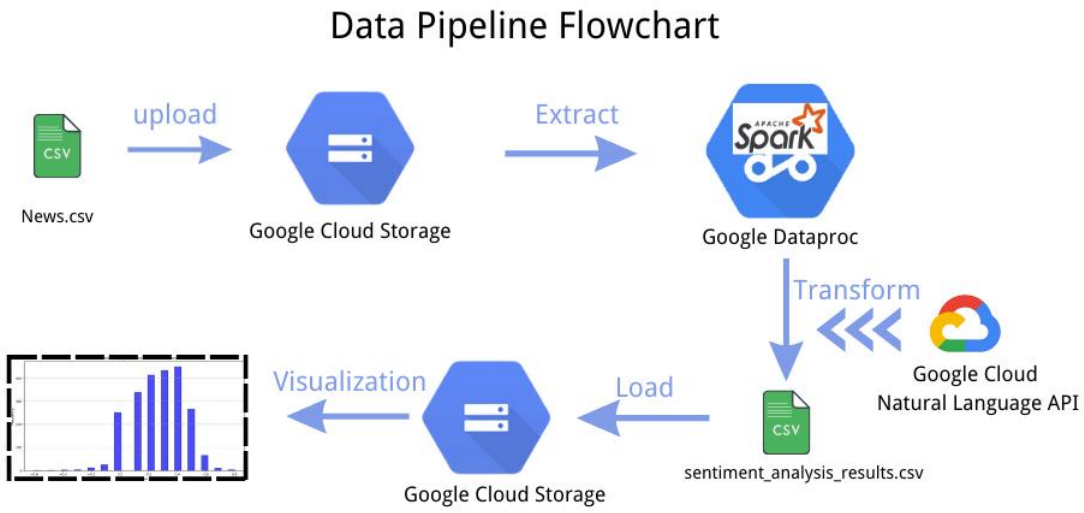
- Data Analyst
- System Administrator

b. Sequence Diagram

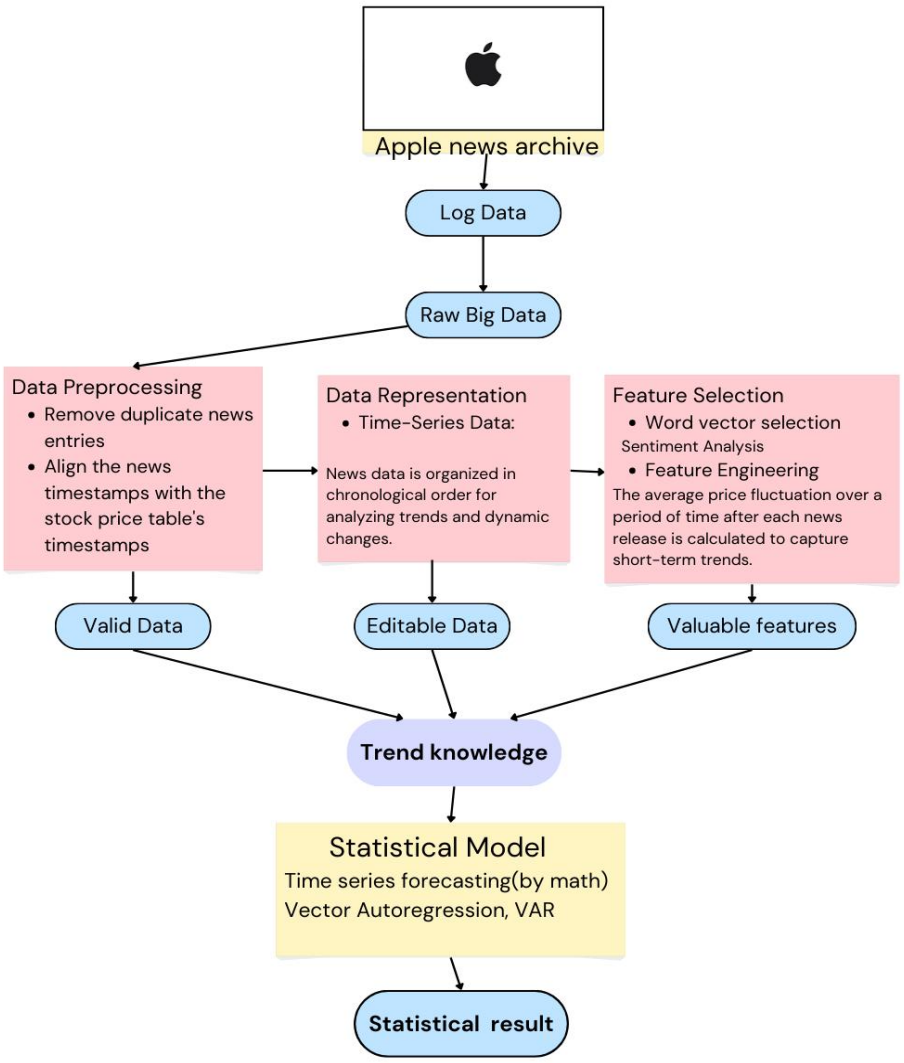


4. Data Design

a. ETLV Process (Explanation)



b. Data Management



- What data is being selected for this project?

The selected data includes news articles related to Apple, including the content of the articles and their publication times.

- Why was this data selected (and other datasets excluded)?

Apple news data was selected because it directly relates to understanding the impact of news sentiment on Apple's stock price. Other types of news or data from unrelated companies were excluded.

- How was the data obtained?

The news data was scraped from the Apple news archive using my own web crawler.

- What are known issues in the data?

Known issues in the data include potential noise in the news content (irrelevant or speculative information).

- What does the data look like? (Nominal, Ordinal, Discrete, Continuous)

News content: Nominal (text data).

Publication time: Ordinal or DateTime.

- How did you alter the data (transformations, imputations, other data cleaning techniques applied, etc.)

The news timestamps were aligned with the stock price data, and sentiment analysis was performed on the articles.

- Where is the data located?

The Apple news data is stored in Google Cloud Storage (GCS).

- How frequently is the data refreshed?

The news data is periodically updated, and may require manual scraping by staff and uploading to the system.

- Is selected data can be good input for yielding correct business plans?

Yes, this data helps analyze the impact of news sentiment on Apple’ s stock price, which can inform business decisions related to market trends.

5. Data Analytics and Modeling

- Diagnostic

Sentiment and Stock Price Relationship: Analyzing the correlation between sentiment scores (from news articles) and stock price changes to diagnose how news impacts stock movements.

Feature Analysis: Identifying which features (e.g., sentiment scores, time-based features) are most influential in stock price changes.

- Statistical or ML

Sentiment Analysis and Stock Price Calculation: Using mathematical methods to calculate the relationship between sentiment scores from news articles and stock price movements, such as regression analysis or correlation studies.

Statistical Models: Applying statistical techniques to model and quantify the impact of sentiment on stock price fluctuations.

Future LSTM Application: These analyses will provide the foundation for more advanced time-series forecasting, where LSTM (Long Short-Term Memory) networks will be used to predict future stock prices based on sentiment over time.

6. Data Visualization

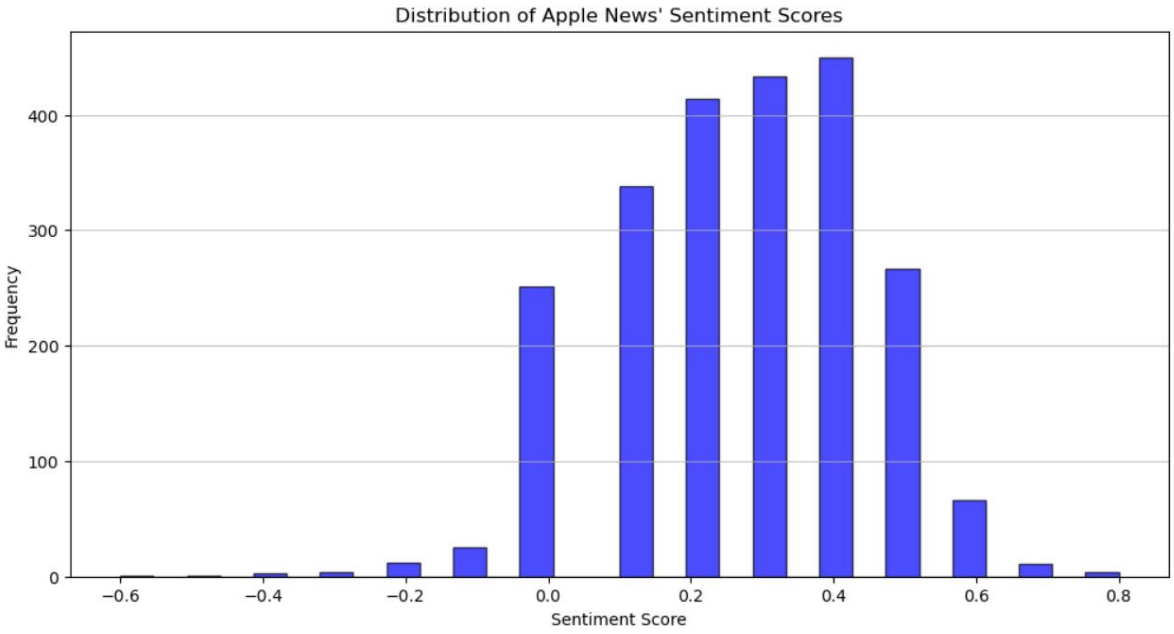


Figure 1. Distribution of Apple News' Sentiment Score

Figure 2. Average Future Stock Price Change by Sentiment Score

7. Code (Only main modules in report | submit code separately)

```
from pyspark.sql import SparkSession

from pyspark.sql.functions import col

from google.cloud import language_v1

import pandas as pd


# Create SparkSession

spark = SparkSession.builder.appName("SentimentAnalysis").getOrCreate()
```

```
# Read data from GCS

df = spark.read.csv("gs://datakimmy1/apple_news.csv", header=True, inferSchema=True)
```

```
# Show the first few rows of data

df.show(5)
```

```
# Define the sentiment analysis function

def analyze_sentiment(text):

    client = language_v1.LanguageServiceClient()

    document = language_v1.Document(content=text, type_=language_v1.Document.Type.PLAIN_TEXT)

    response = client.analyze_sentiment(request={'document': document})

    return response.document_sentiment.score   # Return sentiment score
```

```
# Use UDF to apply sentiment analysis to the Content column

from pyspark.sql.functions import udf

from pyspark.sql.types import FloatType
```

```
# Create UDF

sentiment_udf = udf(analyze_sentiment, FloatType())
```

```
# Add a new column to store sentiment scores

df_with_sentiment = df.withColumn("sentiment", sentiment_udf(col("Content")))
```

```
# Progress tracking

num_rows = df.count()  # Get total number of rows

for i in range(0, num_rows, 10):
```

```
    print(f"Processing rows {i + 1} to {min(i + 10, num_rows)}...")
```

```
# Retain only the "Time" and "sentiment" columns
df_final = df_with_sentiment.select("Time", "sentiment")
```

```
# Convert the DataFrame with sentiment scores to a Pandas DataFrame for local saving
pandas_df = df_final.toPandas()
```

```
# Print the first few rows of the final result
print("Sentiment score result:")
print(pandas_df.head())
```

```
# Save to GCS
output_path = "gs://datakimmy1/sentiment_analysis_results.csv"
df_final.write.csv(output_path, header=True, mode='overwrite')
```

```
# Close Spark Session
spark.stop()
```

8. Test Cases

https://umsystem.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=89e3a7aa-eea7-4b1f-b22c-b229013c51dd

(This is a recording of the main part, not including the scratch part.)

9. References

[1]Bisong E, Bisong E. An overview of google cloud platform services[J]. Building Machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners, 2019: 7-10.

[2]Fanni S C, Febi M, Aghakhanyan G, et al. Natural language processing[M]//Introduction to Artificial Intelligence. Cham: Springer International Publishing, 2023: 87-99.

[3]Shapiro A H, Sudhof M, Wilson D J. Measuring news sentiment[J]. Journal of econometrics, 2022, 228(2): 221-243.