CSYE7200 Project Real-Time Social Media Keywords Sentiment Analysis System

Burning Crusade

Team member: Weifan Guo, Xuanli Liu, Zijie Zhou

Project Description

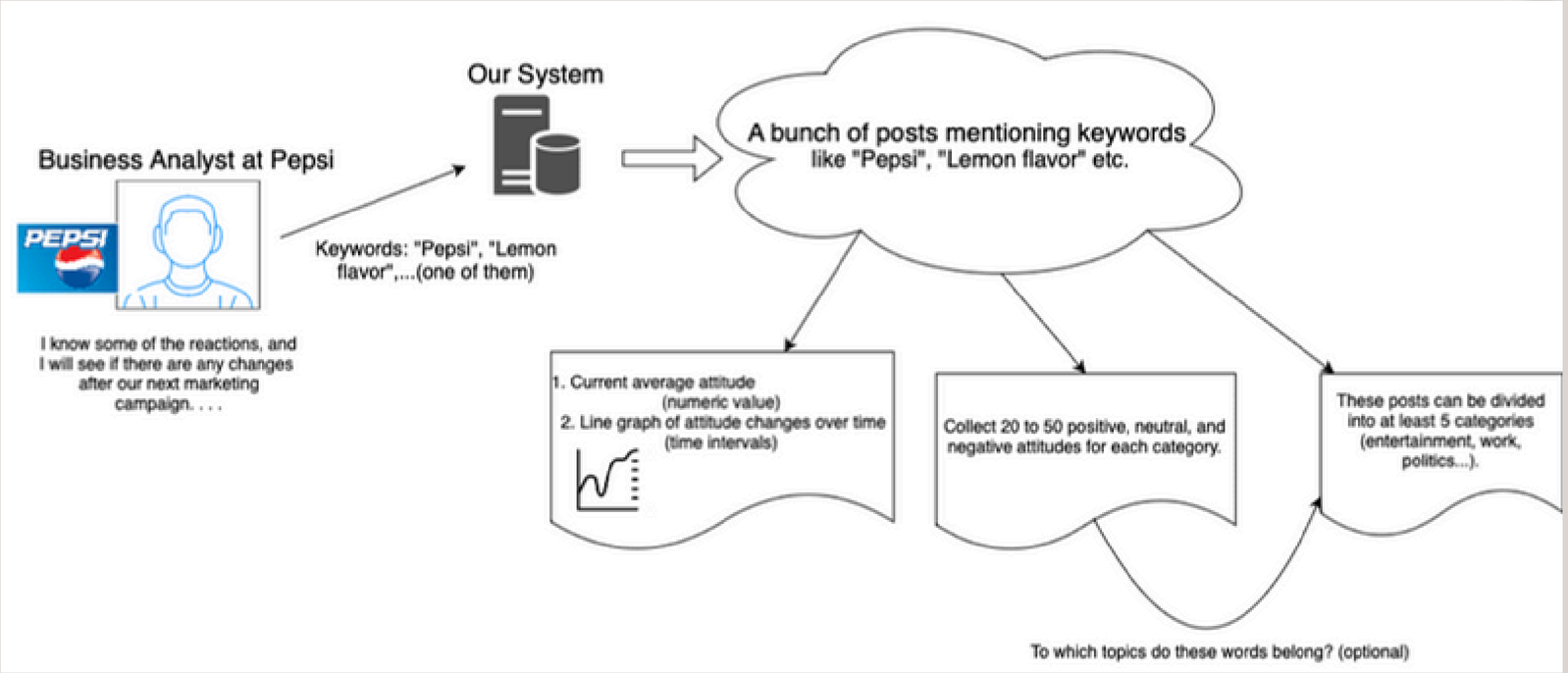
We built a real-time sentiment analysis system that collects and processes data from Youtube and news APIs.

Using Kafka, Spark, and NLP models, we classify sentiment and display results in a live dashboard.

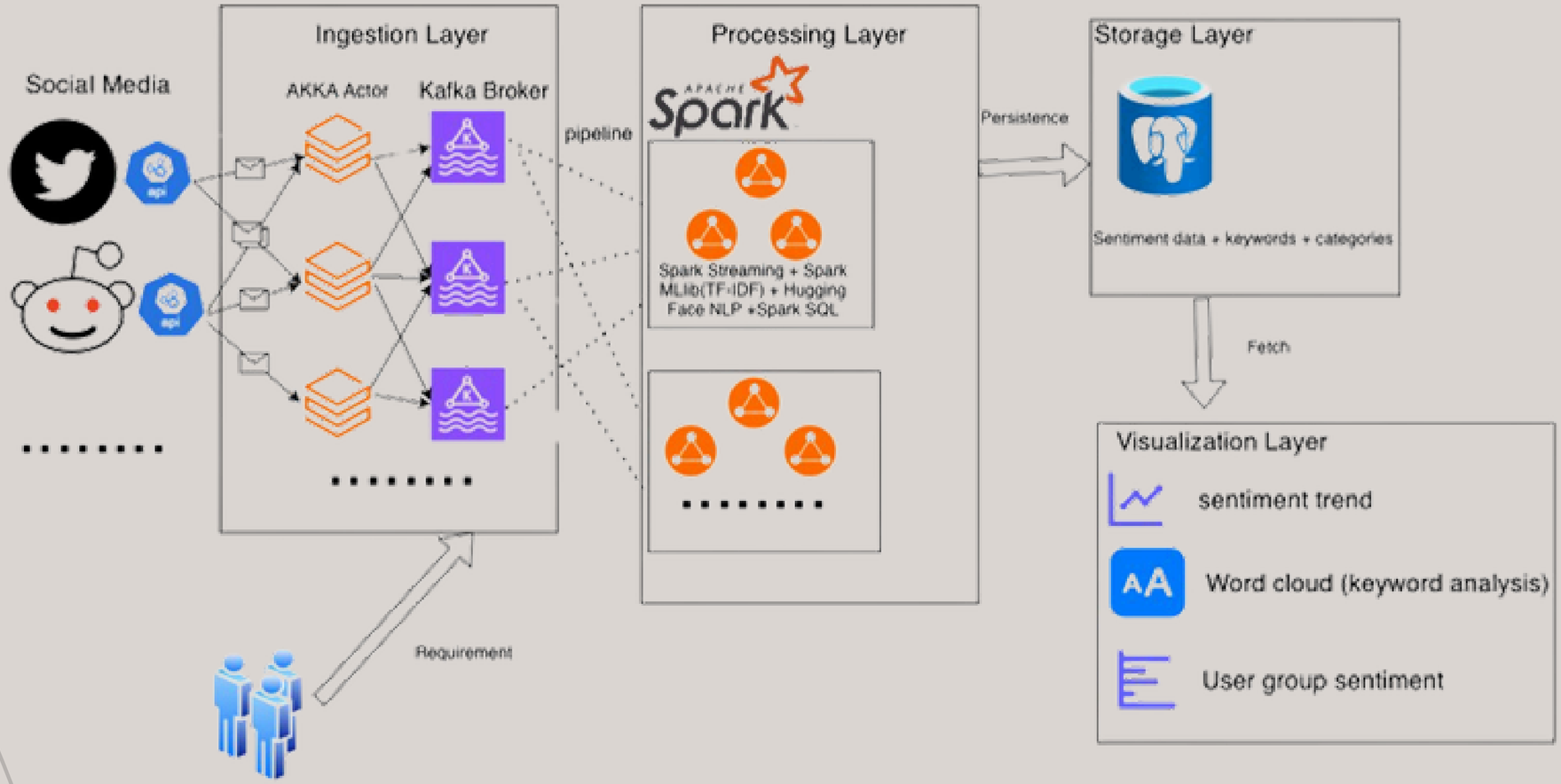
The system is scalable, concurrent, and optimized for streaming performance.

Use Case

- Corporate brand manager inputs keywords like “Pepsi”, “Lemon flavor”
- → System continuously collects related posts, analyzes sentiment, and visualizes trends on a dashboard..
- News analyst monitors real-time sentiment related to political keywords
- → System provides emotional distribution, trending terms, and time-series sentiment graphs.



Methodology

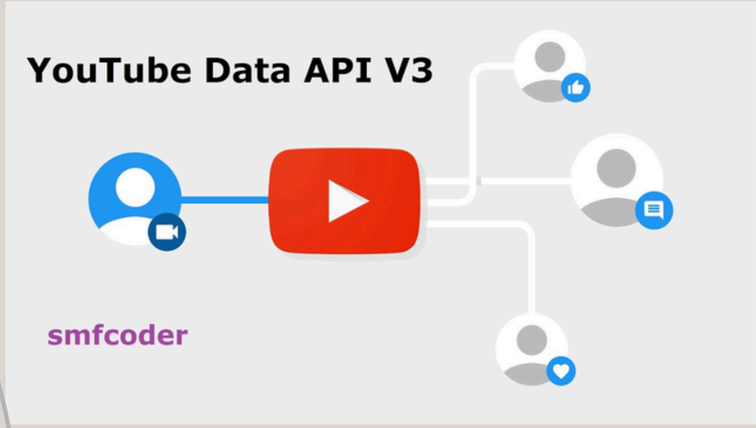


Methodology

```
val youtubeFetcher1 = context.spawn(YoutubeFetcherActor(kafka, ytbCfg1), "YoutubeFetcher_zijie")
val youtubeFetcher2 = context.spawn(YoutubeFetcherActor(kafka, ytbCfg2), "YoutubeFetcher_weifan")
val youtubeFetcher3 = context.spawn(YoutubeFetcherActor(kafka, ytbCfg3), "YoutubeFetcher_xuanli")

val newsFetcher1 = context.spawn(NewsFetcherActor(kafka, newsCfg1), "NewsFetcher_zijie")
val newsFetcher2 = context.spawn(NewsFetcherActor(kafka, newsCfg2), "NewsFetcher_weifan")
val newsFetcher3 = context.spawn(NewsFetcherActor(kafka, newsCfg3), "NewsFetcher_xuanli")
```

- Due to rate limits on Twitter and Reddit APIs, we integrated Google Cloud YouTube API v3 and news API. This offer higher throughput and enhance our real-time sentiment and keyword analysis.



News API

Methodology

$$\text{positive:} \quad t = 5 \begin{cases} P > 0.5: & t = 2 \\ \text{neutral:} & P < 0.5: \quad t = -2 \end{cases}$$

$$\text{negative:} \quad t = -5$$

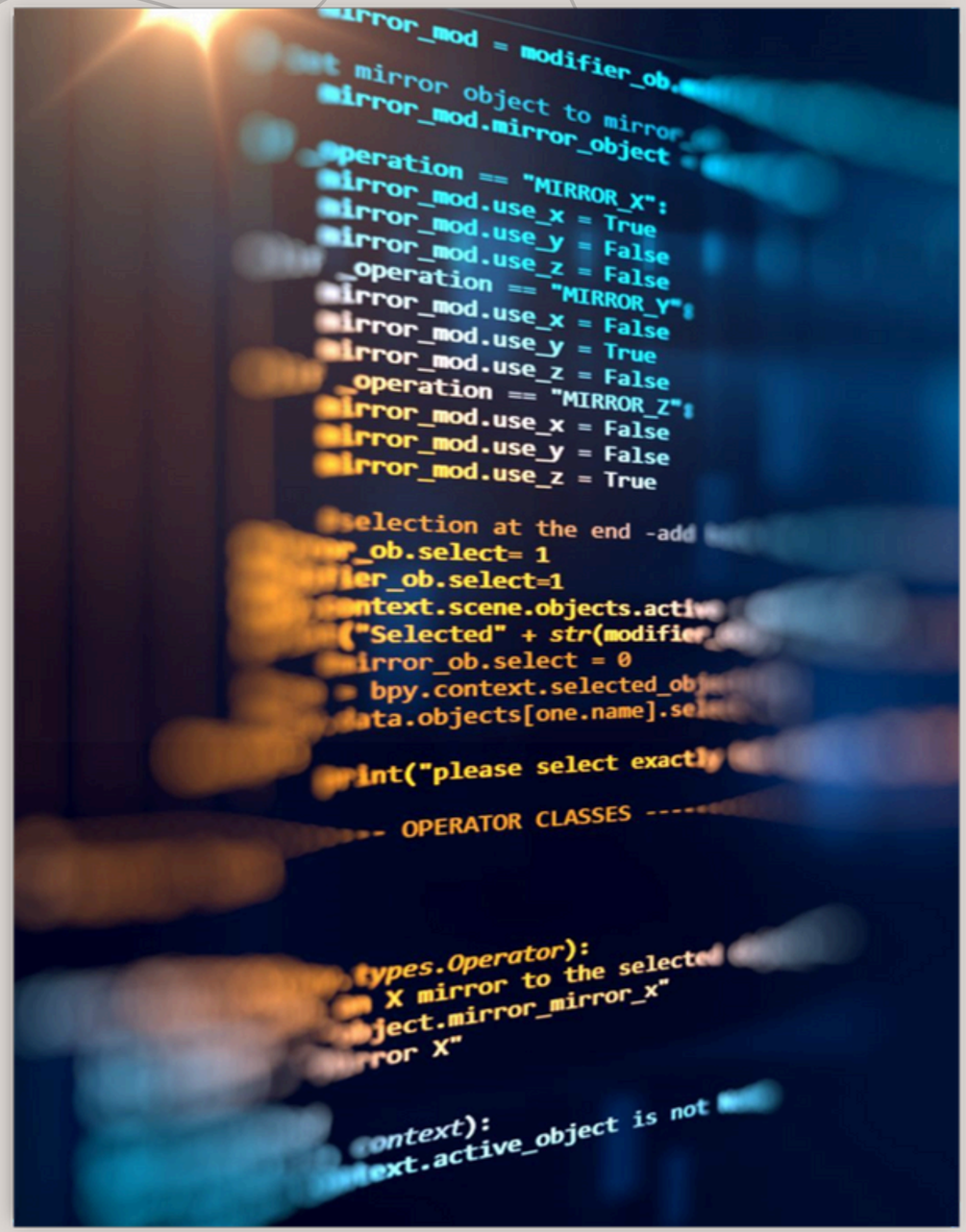
$$\text{score} = 5 + P \cdot t$$

$$\{\text{ negative, } 0.4) \quad \text{score} = 5 - (.4) = 5 - 2 = 3$$

$$\{\text{ neutral, } 0.8) \quad \text{score} = 5 + (2.8) = 6.6$$

- **Feature Engineering**

We assign scores based on sentiment type and confidence:
score=5+P · t
This helps translate sentiment and probability into a unified numeric value for analysis.



Data sources

Data Volume & Scalability

- Google Cloud YouTube V3 API and News API are used to collect real-time text, timestamps, user info.
- Due to rate limits (1000 requests per user/day for Google API, 100 requests per user/day for News API), we distribute the load across multiple accounts.
- Kafka Streaming: Ingest data into Kafka topics; Spark Streaming (or equivalent) consumes data in real time.
- Ingestion Rate: ~3 new tweets/comments per second
- Data Size: 100m
- Github:https://github.com/Zijie000/social-media-keywords-sentiment-analysis-system

About Scala

✓ **Real-Time Data Streaming**
- Implemented Kafka producers/consumers for continuous data ingestion
- Efficiently processed Google Cloud API and News API streams

✓ **Big Data Processing (Apache Spark)**
- Utilized Spark Streaming for real-time, large-scale data processing
- Performed distributed sentiment analysis using parallel computation

✓ **Natural Language Processing (NLP)**
- Integrated Spark NLP and Hugging Face Transformers for sentiment classification (positive, neutral, negative)
- Optimized model inference for real-time performance

✓ **Concurrent Processing (Akka Actors)**
- Enabled concurrent execution of sentiment analysis tasks
- Enhanced system scalability, responsiveness, and fault tolerance
  -

Project Milestones

Functional Sentiment Analysis System

Milestone #1

Mar 18 - Mar 27

- Project setup, define architecture,
- API authentication (Twitter & Reddit),
- set up Kafka pipeline.

Milestone #2

Mar 28 - Apr 6

- Implement real-time data ingestion
- Spark Streaming integration
- Store raw data

Milestone #3

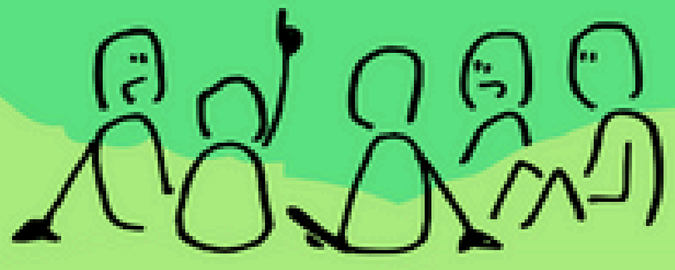
Apr 7 - Apr 16

- Develop sentiment analysis model
- Optimize classification accuracy

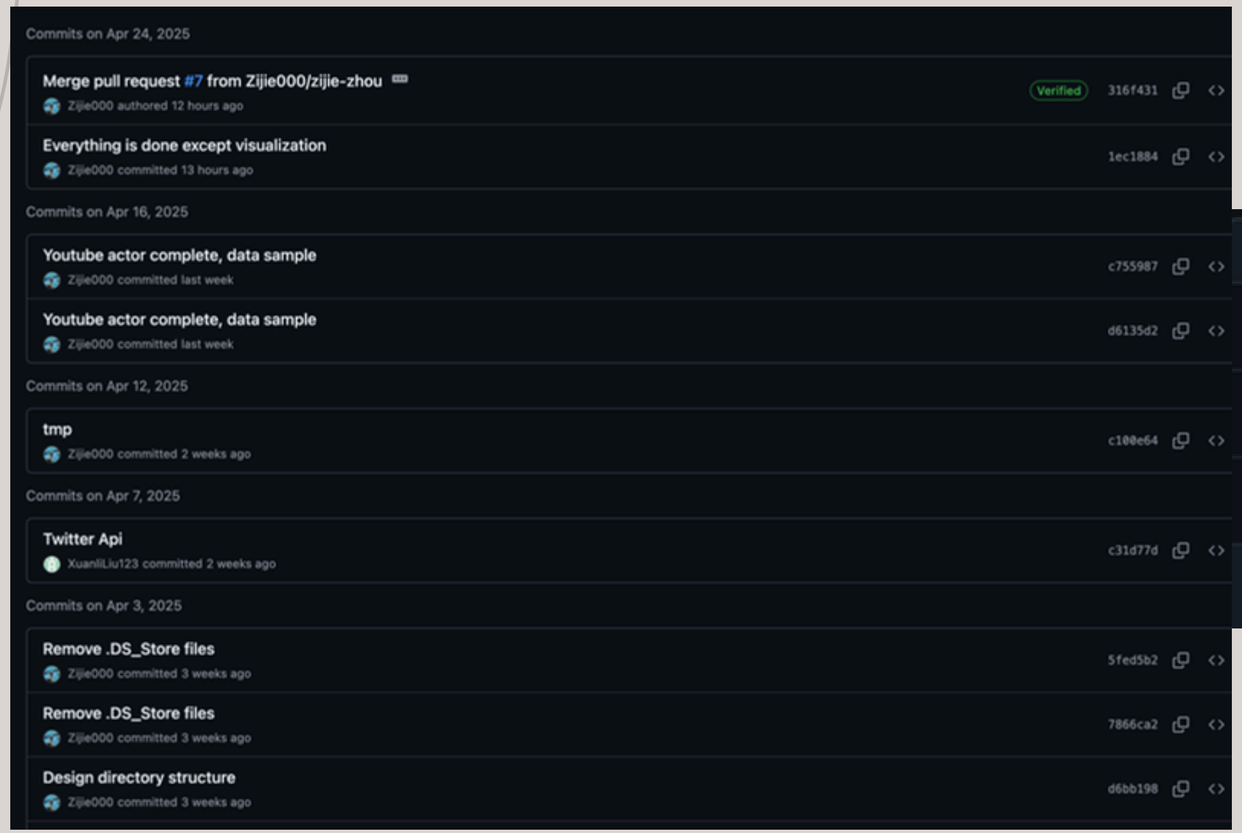
Milestone #4

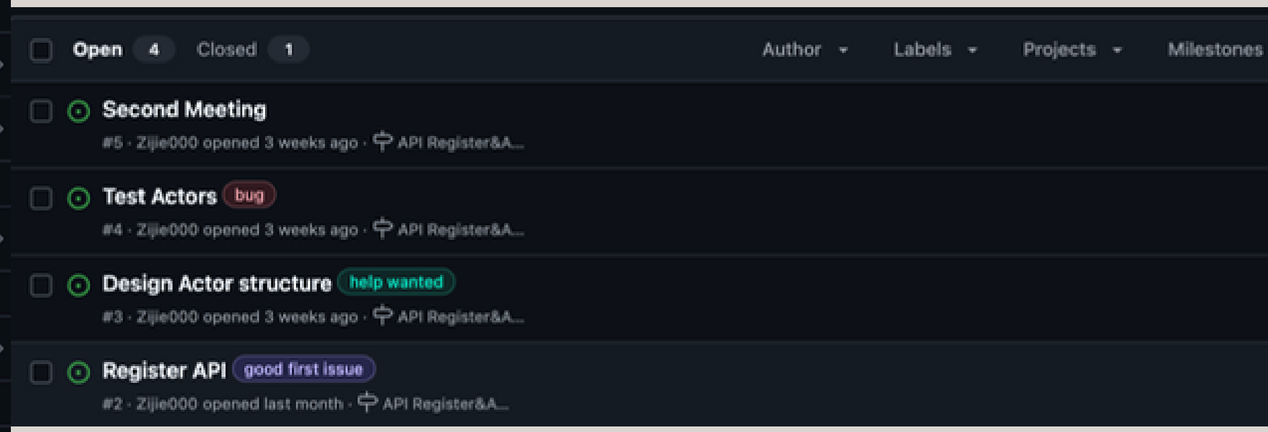
Apr 17 - Apr 24

- Build visualization dashboard
- Finalize API endpoints
- Conduct system testing & optimizations.



Project MileStone





- Our project followed the structured sprint plan, with each milestone spanning approximately 7–10 days

Acceptance criteria

- The system can run at least 1 day without break ✅
- At least ⩾ 3 comments can be collected per second (to ensure the real-time data flow) ✅ over 36 comments
- Support 90% asynchronous API requests without blocking the main thread ✅
- Data deduplication rate ⩾ 85% (the same content is not stored repeatedly) ❓ not yet measured
- It can display the visualization interface, numbers, line chart, word cloud mentioned in the above Hypothetical customers. ✅

Goals of the project

- ✅ Real-Time Data Streaming & Processing → Continuous ingestion and transformation of social media data.
- ✅ Sentiment Analysis & Keyword Insights → Detect emotion shifts, trending words, and sentiment distribution.
- ✅ Data Storage & Visualization → Store insights in PostgreSQL, visualize trends in Grafana.
- ❌ Scalable & Extensible System → Capable of handling millions of records, adaptable for different industries.

- We build the sentiment analysis system that delivers real-time insights and meets core project goals.

Thank You

Presented by Burning Crusade