
Project Overview: Real-Time Social Media Sentiment Analysis and Analytics Platform

Objective:

Participants will build a system that collects real-time data from social media platforms, performs sentiment analysis on the data, and generates real-time analytics. This project will involve creating data pipelines for ingesting social media data, processing it in real-time, performing sentiment analysis using machine learning models, and providing dashboards for real-time insights.

Week 1: Data Warehousing and SQL for Social Media Data

Topics Covered:

- Introduction to Data Warehousing for social media data
- SQL for creating tables, querying, and managing social media posts, user profiles, and sentiment scores

Capstone Project Milestone:

- **Objective**: Design a Data Warehouse schema to store social media posts, user data, and sentiment analysis results.

Tasks:

1. **Design Schema**: Create tables to store social media data (e.g., posts, user profiles) and sentiment scores (e.g., positive, negative, neutral).
2. **Querying Data**: Write SQL queries to retrieve posts, user profiles, and calculate metrics like the percentage of positive and negative sentiment over time.

Example Code:

```sql
-- Create schema for social media sentiment analysis
CREATE TABLE user_dim (
    user_id INT PRIMARY KEY,
    user_name VARCHAR(255),
    location VARCHAR(255)
);

CREATE TABLE post_fact (
    post_id INT PRIMARY KEY,
    user_id INT,
    post_text VARCHAR(500),
    post_date TIMESTAMP,
    sentiment_score DECIMAL(3, 2)  -- Sentiment score: -1 to 1
);

-- Query to calculate sentiment trends over time
SELECT DATE(post_date), AVG(sentiment_score) AS avg_sentiment
FROM post_fact
GROUP BY DATE(post_date)
ORDER BY DATE(post_date);
```

Outcome: By the end of Week 1, participants will have set up the Data Warehouse schema to store social media posts and sentiment scores, along with queries to analyze sentiment trends.

Week 2: Python for Data Processing and Sentiment Analysis

Topics Covered:

- Python for extracting social media data and performing sentiment analysis using machine learning models (e.g., NLP techniques like Vader, TextBlob, or Hugging Face transformers)

Capstone Project Milestone:

- **Objective**: Collect and preprocess social media data using Python, and perform sentiment analysis using machine learning models.

Tasks:

1. **Collect Data**: Use APIs (e.g., Twitter API or a simulated dataset) to collect social media posts in real-time.
2. **Sentiment Analysis**: Use machine learning libraries to analyze the sentiment of posts and assign sentiment scores.
3. **Feature Engineering**: Extract useful features from the posts (e.g., hashtags, keywords) for further analysis.

Example Code:

```
import pandas as pd
import tweepy
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

# Authenticate to Twitter (replace with real credentials)
auth = tweepy.OAuthHandler('API_KEY', 'API_SECRET_KEY')
auth.set_access_token('ACCESS_TOKEN', 'ACCESS_TOKEN_SECRET')
api = tweepy.API(auth)

# Collect real-time tweets (replace with your own query)
tweets = api.search(q='data engineering', lang='en', count=100)

# Perform sentiment analysis using VADER
analyzer = SentimentIntensityAnalyzer()
for tweet in tweets:
    sentiment_score = analyzer.polarity_scores(tweet.text)['compound']
    print(f"Tweet: {tweet.text}\nSentiment Score: {sentiment_score}")
```

Outcome: By the end of Week 2, participants will have collected social media data in real-time and performed sentiment analysis using Python.

Week 3: Real-Time Data Processing with Apache Spark and PySpark

Topics Covered:

- Using Apache Spark and PySpark for large-scale real-time data processing
- Processing and analyzing social media data in real-time to calculate sentiment scores and trends

Capstone Project Milestone:

- **Objective**: Implement real-time data processing using Apache Spark and PySpark to analyze social media data and compute sentiment scores in real-time.

Tasks:

1. **Set Up Spark Streaming**: Configure Apache Spark for real-time streaming of social media data.
2. **Real-Time Sentiment Analysis**: Use PySpark to process real-time posts and compute sentiment scores for each post.
3. **Real-Time Trend Analysis**: Calculate and update sentiment trends in real-time.

Example Code:

```
from pyspark.sql import SparkSession
from pyspark.sql.functions import udf
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

# Create Spark session for streaming tweets
spark = SparkSession.builder.appName("SentimentAnalysis").getOrCreate()

# Define UDF to calculate sentiment scores
analyzer = SentimentIntensityAnalyzer()
@udf("float")
def sentiment_udf(text):
    return analyzer.polarity_scores(text)['compound']

# Read streaming data (e.g., from Kafka)
tweet_stream = spark.readStream.format("kafka").option("subscribe", "tweets").load()

# Apply sentiment analysis
tweet_stream = tweet_stream.withColumn("sentiment_score", sentiment_udf("text"))

# Write sentiment analysis results to console or a database
query = tweet_stream.writeStream.outputMode("append").format("console").start()
query.awaitTermination()
```

Outcome: By the end of Week 3, participants will have implemented real-time sentiment analysis using Apache Spark and PySpark to process and analyze social media data in real-time.

Week 4: Real-Time ETL Pipelines with Azure Databricks

Topics Covered:

- Azure Databricks for building ETL pipelines for social media sentiment analysis
- Delta Live Tables for real-time processing and incremental data updates

Capstone Project Milestone:

- **Objective**: Build real-time ETL pipelines to process social media data and perform sentiment analysis using Azure Databricks.

Tasks:

1. **Set Up Databricks**: Configure Azure Databricks for real-time processing of social media data.
2. **Build ETL Pipeline**: Create a pipeline that processes social media data, computes sentiment scores, and updates the Data Warehouse in real-time using Delta Live Tables.

Example Code:

```
# Create Delta Live Table to process social media posts
CREATE OR REPLACE LIVE TABLE social_media_posts
AS SELECT * FROM streaming.`/mnt/streaming/social_media_posts`;

# Build ETL pipeline to compute sentiment scores in real-time
CREATE OR REPLACE STREAMING LIVE TABLE sentiment_analysis
AS SELECT post_id, user_id, post_text, sentiment_udf(post_text) AS sentiment_score
FROM social_media_posts;
```

Outcome: By the end of Week 4, participants will have built real-time ETL pipelines using Azure Databricks to process social media data and compute sentiment scores.

Week 5: Deploying the Real-Time Sentiment Analysis Platform with Azure DevOps

Topics Covered:

- Automating deployment of the sentiment analysis pipeline with Azure DevOps
- Implementing CI/CD for deploying and monitoring the sentiment analysis system

Capstone Project Milestone:

- **Objective**: Deploy and automate the sentiment analysis pipeline using Azure DevOps for continuous integration and monitoring.

Tasks:

1. **Azure DevOps Pipelines**: Set up Azure DevOps pipelines to automate the deployment of the sentiment analysis system.
2. **Monitoring and Alerts**: Set up monitoring and alerting for real-time sentiment trends (e.g., detect sentiment spikes or negative trends).

Example Code:

```
# Azure DevOps pipeline YAML for deploying the sentiment analysis system
trigger:
  - main

pool:
  vmImage: 'ubuntu-latest'

steps:
  - task: UsePythonVersion@0
    inputs:
      versionSpec: '3.x'

  - script: |
      pip install -r requirements.txt
```

```
      python deploy_sentiment_analysis.py
    displayName: 'Deploy Sentiment Analysis System'
```

Outcome: By the end of Week 5, participants will have deployed and automated the real-time sentiment analysis pipeline using Azure DevOps.

Summary of Outcomes:

1. **Week 1**: Design a Data Warehouse schema for social media posts and sentiment analysis, and write SQL queries to analyze sentiment trends.
2. **Week 2**: Collect social media data and perform sentiment analysis using machine learning models in Python.
3. **Week 3**: Implement real-time data processing and sentiment analysis using Apache Spark and PySpark.
4. **Week 4**: Build real-time ETL pipelines with Azure Databricks to process social media data and compute sentiment scores.
5. **Week 5**: Automate the deployment and monitoring of the real-time sentiment analysis platform using Azure DevOps.