DSP Project: Weather Data Analysis and Kafka Integration

Alpha Oumar Barry, Vianney Grimaud

Table of Contents

Introduction

Comprehensive solution for real-time weather data processing:

- Streamlit dashboard for visualization.
- Kafka integration for seamless data processing.
- GeoPandas for geographical data plotting.

Figure: Streamlit Dashboard.

Problematic and Methodology

Objective: Construct a user-friendly dashboard for exploring real-time weather data.

- Utilizes WeatherAPI.com for current weather information.
- Kafka for efficient data streaming.
- Various analysis techniques for temperature trends.

Dashboard Construction

Streamlit Dashboard:

- Developed using Streamlit in the `dashboard.py` script.
- Interactive platform for users to explore weather data.
- Features include location-based weather search and display of current conditions.
- Prioritizes user experience for easily understandable weather data presentation.

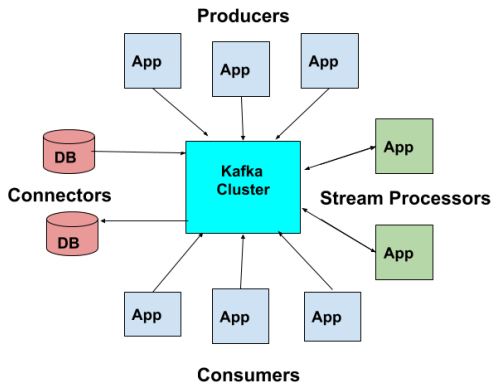
Kafka Integration

Kafka Producer (`kafka_producer.py`):

- Fetches real-time weather data for a specified location.
- Publishes data to a Kafka topic using KafkaProducer.
- Utilizes the `publish_messages_to_kafka` function.

Kafka Consumer (`kafka_consumer.py`):

- Consumes weather data from the Kafka topic.
- Updates a Streamlit dashboard in real-time.
- Uses KafkaConsumer and the `consume_messages_from_kafka` function.



Online Machine Learning with Kafka

Online Learning Model (`analysis3.py`):

- Implements online machine learning using the `river` library.
- Performs online linear regression on weather data.
- Predictions published back to Kafka for real-time updates.

Conclusion & Future Directions

Conclusion: Successful integration of Streamlit, Kafka, and temperature trend exploration with `SGDRegressor`.

Future Directions:

- Integration of additional WeatherAPI.com APIs for more comprehensive analysis.
- Exploration of machine learning-based weather prediction.