Christian Frago

Christian.Frago@MyGeorgian.ca

BDAT1002

Data Systems Architecture

Final project

Apache Kafka for Twitter Ingestion

# Introduction

In the world of Big Data, one of the most daunting tasks is acquiring the data needed for analysis and predictions. In this context, this assignment focuses on setting up a pipeline for data ingestion from Twitter, followed by data cleaning, processing, and loading into a Hive table for further analysis. This pipeline will utilize various tools such as Apache Kafka, and Apache Hive to achieve the desired outcome. The goal of this assignment is to enable the use of data from Twitter for gaining insights through Hive queries, helping to solve real-world problems.

It's worth mentioning that this project serves as a practical application of the concepts and tools learned in the course. As such, this assignment represents an opportunity for students to apply their learning in a real-world setting. By completing this project successfully, students will gain hands-on experience in data ingestion, cleaning, processing, and analysis, using a variety of tools, including Apache Kafka, Apache Flume, and Apache Hive. This will not only help them to solidify their understanding of the course material but also equip them with valuable skills for tackling real-world Big Data challenges.

# Objective

The objectives of this project are as follows:

* Ingesting data from Twitter: The primary objective of this project is to set up a pipeline for ingesting data from Twitter using tools such as Apache Kafka and Apache Flume.
* Cleaning and processing data: The data collected from Twitter needs to be cleaned and processed to remove irrelevant data and transform it into a format that is suitable for analysis.
* Loading data into Hive: Once the data has been processed, it needs to be loaded into a Hive table for analysis using Apache Hive.
* Running queries on data: The final objective of this project is to run queries on the data using Apache Hive to gain insights and solve real-world problems.

# Challenges / Difficulties

Below are some challenges encountered in this project

* “kafka.errors.NoBrokersAvailable: NoBrokersAvailable” – This was encountered when running the producers and consumers. This shows up when the Kafka is not running. This is rectified by ensuring that zookeeper and kafka are all started prior to running the producers and consumers.
* Data Cleansing - as the tweets are a free text so it is hard to decide what cleaning will be done as each tweets are different. The common difficulty here is in dealing with icons, emojis and images included in the tweet. A lot of codes for those were identified and replaced since the goal is to do analysis on keywords. There should be continuous update of those characters in the code.
* Familiarity with shell scripting and linux as a whole – this challenged to learn more about different commands and strategies and how to apply those in the project.

# Configuration

In the context of this project, several configurations need to be made to ensure the proper functioning of the data ingestion, cleaning, processing, and analysis pipeline. The configuration process can be complex, requiring a deep understanding of the tools and their interdependencies.

## Kafka Producer Set Up

Kafka producer is responsible for collecting data from various sources, transforming it into a format that is compatible with Kafka, and publishing it to the Kafka cluster for further processing. Below is the set up of the kafka Producer.

Text

Description automatically generated

Text

Description automatically generated

## Kafka Consumer Set Up

Proper set up of Kafka consumer is made to ensure that the consumer is correctly configured to read and process data from the Kafka cluster. In the following sections, we will discuss the set up of a Kafka consumer:

Text

Description automatically generated

## Scheduled Job Set

A scheduled job is a critical component of any data pipeline that needs to be executed at predetermined intervals automatically. In the context of this project, a scheduled job is responsible for running the Kafka producer and cleaning commands Python files regularly. This ensures that the data ingestion and processing pipeline remains operational and up-to-date.

Text

Description automatically generated

## Data Cleaning

Data cleaning is a critical step in any data processing pipeline, especially in Big Data settings. In this project, data cleaning is necessary to remove irrelevant or erroneous data and transform the collected data into a format suitable for analysis using Apache Hive.

Text

Description automatically generated

# Hive Tables Set Up

HDFS tables are used to store and organize the data, making it accessible for querying and analysis. In this regard, the setup of HDFS tables involves configuring the tables to store the processed data in a way that enables efficient querying and analysis using Apache Hive.

## Hive Table

Text

Description automatically generated

## Analysis Query

Display the count of twitter mentions of the top players:

SELECT keywords, count(\*) as tweetcount

FROM nba\_playoffs

GROUP BY keywords

ORDER BY tweetcount desc;

Text

Description automatically generated

# Appendix

Sample consumer.log (before cleaning)

Text

Description automatically generated

Sample consumer\_clean.csv (data after cleaning)

Background pattern

Description automatically generated

Sample logfile.log

Text

Description automatically generated

Sample Hive Query

Graphical user interface, text

Description automatically generated