Project 2

Group 17

Group members: Sai Aparanji Nemmani, Aniket Dineshkumar Hirpara

Output Link: s3://cloudcomputingvt/Aparanji/top_twenty_hashtags.txt/

Description:

The code snippet processes JSON data containing tweets to identify and count hashtags, leveraging PySpark's distributed processing capabilities. It utilizes PySpark, a Python API for Apache Spark, for distributed data processing. The SparkSession is initialized with the specified application name, "HashtagCount", using .builder and .getOrCreate() methods. This sets up the environment for Spark processing. The read_data function reads JSON data from the specified file path. This data contains tweets in JSON format. The hashtag_mapper function is defined to extract hashtags from the JSON data. It recursively traverses through the JSON structure to find strings containing hashtags using regular expressions. Each hashtag is mapped to a count of 1. The JSON data is loaded into a Resilient Distributed Dataset (RDD) using sparkSess.read.text(file_path).rdd.map(lambda row: row.value). This enables distributed processing of the data.

The hashtag_mapper function is applied to each element of the RDD using flatMap, resulting in a collection of (hashtag, count) pairs. The counts of hashtags are aggregated using reduceByKey to calculate the total count for each hashtag. The top 20 hashtags with the highest counts are extracted using takeOrdered and sorted by count in descending order. The top 20 hashtags along with their counts are formatted into a string (output_str). This string is written to a text file located at output_path using Spark's parallelized write operation. Finally, the Spark context is stopped to release resources.