Project 2 – X Hashtag Counter using Amazon EMR Report

Group Number : 8
Group Members : Haritha Injam, Divakara Rao Annepu

Output Solution location : s3://cloudcomputingvt/Haritha/Top_20_HashTag_Output

Short Description :

In this project, the objective was to analyze social media data to identify the top 20 hashtags from a dataset of tweets assisting a political campaign team in understanding the concerns and interests of city constituents. The dataset for this analysis was stored in an S3 bucket on AWS, ready to be processed using big data tools.
We decided to utilize Amazon Elastic MapReduce (EMR) for this task due to its ability to process large amounts of data efficiently. Amazon EMR was chosen because it provides a managed cluster platform that simplifies running big data frameworks like Apache Hadoop and Apache Spark, which are perfect for processing vast datasets quickly.The chosen big data processing tool is Apache Spark, renowned for its in-memory data processing capabilities and compatibility with the Hadoop ecosystem, which is pivotal given the dataset's voluminous nature. Once the local testing confirms the application's functionality, the next phase involved deploying it on AWS EMR, an orchestrated big data platform that simplifies running big data frameworks for processing and analyzing large datasets. The process begins with launching an EMR cluster adhering to the project guidelines. This includes setting up a transient cluster with specific hardware and software configurations such as selecting emr-7.0.0, ensuring only necessary applications are chosen and configuring the cluster to terminate after a period of inactivity.

In the PySpark application, the map function is employed to extract hashtags from each tweet and normalize them to lowercase ensuring consistent counting regardless of text case. Each hashtag is then mapped to a key-value pair where the key is the hashtag itself, and the value is 1. The reduceByKey function aggregates these pairs by summing the values for each unique hashtag effectively counting occurrences. This aggregation process culminates in identifying the top 20 hashtags by applying a sorting algorithm on the counts. The results are then formatted and written to an S3 bucket providing accessible insights into the most prevalent topics within the dataset.

Ensuring secure access to the EMR cluster particularly for maintenance or troubleshooting necessitates configuring the security group for the main node to allow SSH connections. Once the cluster status indicates

readiness the Spark job is executed remotely and its outputs are monitored for accuracy and completeness. Post-processing, the cluster's automatic termination feature aids in curtailing unnecessary costs aligning with the project's budgetary constraints and adherence to AWS usage guidelines.