BigData

EVALUACIÓN

PROGRAMACIÓN CON MAPREDUICE Y HDFS

21 AL 24 DE FEBRERO

CRITERIOS DE EVALUACIÓN:

- Comprensión de los conceptos básicos de MapReduce y HDFS
- Habilidad para diseñar algoritmos MapReduce eficientes
- Habilidad para implementar algoritmos MapReduce en Hadoop

Los criterios para autoevaluación, coevaluación y heteroevaluación, se define a partir de la rúbrica anexa.

Construcción del DataSet (individual): 10 pts

1. Seleccionar, cada estudiante, 15 libros del proyecto web https://www.gutenberg.org/
2. General un listado con el link de descarga del libro en el formato Plain Text UTF-8
3. Realizar un script (Shell o Python) para descarga los 15 libros (con ***wget*** *o* ***curl***) al interior de la carpeta llamada **input**. Se puntúa si el script es genérico y hace la descarga de cualquier listado.
    - *Entrega 1: Script individual con el nombre* ***1_download_book.sh.***

Análisis de palabras comunes de 15 libros (individual) 20 pts

4. Para los 15 libros, y por medio de MapReduce, cada estudiante debe identificar las 20 palabras comunes y más recurrentes en los libros que tengan una longitud mayor a 5 caracteres.
    - *Entrega 2: Script individual de MapReduce, con el nombre* ***2_Wordcount.py****. se debe explicar dentro del script las líneas fundamentales de código. Al final del código colocar comentado el comando que se ha utilizados para ejecutar este proceso. Ejemplo:*

        #python3 2_Wordcount.py -r hadoop --output-dir out --no-cat-output hdfs://user/root/input
    - *Entrega 3: Archivo de salida con los resultados, con el nombre* ***3_out.txt***

Análisis de popularidad de Autores (grupal) 40 pts

5. En el texto de cada libro se encuentra el nombre del autor. Utilizar mrjob para extraer el nombre del autor de cada libro. General un archivo de salida con el listado de los autores y el número de libros de dicho autor del listado total de libros de los estudiantes del grupo.
    - *Entrega 4: Script grupal de MapReduce, con el nombre* ***4_Autorcount.py*** *se debe explicar dentro del script si se usa o no combiner. Este problema se puede abordar con varios enfoques, por lo tanto, el explicar la metodología de trabajo para resolvieron el problema de análisis de popularidad.*
    - *Entrega 5: Archivo de salida con los resultados, con el nombre* ***5_out.txt***

- *Entrega 6: Entregar el script de descarga para el input de los datos **6_books.sh***

Análisis de tweets (grupal) 30pts

6. Con el DataSet tweets2016_olympic_rio.test, colección de mensajes de Twitter recopilados durante los Juegos Olímpicos de Río 2016 de la API responder:
   - ¿Cuál es el día más popular respecto a la actividad de Twitter y cuantos tweets hay de ese día? Se recomienda usar la librería **time** con los métodos **gmtime** y **strftime**
   - ¿Cuál es el promedio de la longitud de texto de cada tweet?

   Entregables:

   - Archivos de texto plano con el resultado para cada pregunta en los archivos **7a_out.txt y 7b_out.txt**
   - Script de mapreduce utilizado para cada pregunta en los archivos **8a_tweetcount.py y 8b_tweet average.**

   Recuerdes Los datos se almacenan en formato CSV, y cada línea contiene los siguientes campos:
   a. **epoch_time**: Marca de tiempo UNIX del tweet en milisegundos desde el 01-01-1970.
   b. **tweetId**: ID único del tweet
   c. **tweet**: contenido del tweet que incluye el #hashtags
   d. **device**: información adicional meta-datos, donde se incluye el dispositivo usado