viu
.es

2022-2023

ACTIVIDAD 1: Programación con Hadoop MapReduce.

Máster en Big Data y Data Science

01MBID – Fundamentos de la Tecnología de Big Data

Autores: Adrián Hernández Padrón y Andrea San Blas Hernández.

Fecha: 24/06/2022

Dado el siguiente programa Big Data que tiene defectos de diseño, se debe entender cuál es el defecto y hacer un informe que contenga: (1) Nombre y apellidos, (2) Tiempo empleado por el alumno en entender cuál es el defecto, y (3) Descripción del defecto. Todos los archivos del programa se encuentran en la carpeta: "PersonasQueCompranEnMuchasTiendas".

IMPORTANTE: los defectos del programa son defectos del diseño. La sintaxis del programa es correcta, pero la funcionalidad del programa no se ha programado correctamente siguiendo el modelo de procesamiento Big Data. Por ello, puede que los programas los ejecutemos en nuestro ordenador y funcionan correctamente, pero al moverlos al cluster Big Data empiecen a fallar porque están ejecutando varias Mapper, Combiner, Reducer, algunas de ellas puede que se re-ejecuten, acaben antes, etc. Es decir, si ejecutamos dos veces en un cluster de producción el mismo programa con los mismos datos, podría una vez emitir la salida correcta y otra vez una incorrecta. Esto es porque el programa no se diseñó adecuadamente siguiendo el modelo de procesamiento Big Data. Un programa bien diseñado, debería ejecutarse correctamente independientemente de cómo el cluster Big Data decida ejecutarlo. A continuación, se describe el programa y las preguntas que se tienen que responder:

Conjunto de datos: cada fila representa una compra que hizo una persona en una tienda. La fila tiene la siguiente estructura: "persona tienda".

Por ejemplo "Alice Nunc Corp." significa que Alice compró en "Nunc Corp.".

Descripción del programa: el programa tiene que obtener cuáles fueron las personas que compraron en 3 o más tiendas diferentes. Es decir, si se tiene como entrada:

Alice Nunc Corp.
Alice Arcu Aliquam Company
Alice Pharetra Quisque Ac Company
Alice Nunc Corp,
Bob Nunc Corp.

El programa debería emitir Alice porque compró en 3 o más tiendas distintas. Concretamente, en el ejemplo anterior, Alice compró en 3 tiendas: en "Nunc Corp." (dos compras), "Arcu Aliquam Company", y "Pharetra Quisque Ac Company". El programa no emite Bob porque sólo compró en una tienda.

Código: el equipo de desarrollo ha creado los *scripts:*
mapperPersonasQueCompranEnMuchasTiendas.py,
combinerPersonasQueCompranEnMuchasTiendas.py
reducerPersonasQueCompranEnMuchasTiendas.py.

Comando de ejecución:

```
hadoop jar $HADOOP_HOME/share/hadoop/tools/lib/hadoop-streaming-2.4.0.jar
-file ./mapperPersonasQueCompranEnMuchasTiendas.py
-mapper ./mapperPersonasQueCompranEnMuchasTiendas.py
-file ./combinerPersonasQueCompranEnMuchasTiendas.py
-combiner combinerPersonasQueCompranEnMuchasTiendas.py
```

-file ./reducerPersonasQueCompranEnMuchasTiendas.py
-reducer ./reducerPersonasQueCompranEnMuchasTiendas.py
-input casoDePrueba.txt -output ./misalida

(notar que dependiendo de la versión de Hadoop, habría que cambiar el .jar y también las entradas y salidas)

Problema: el equipo de analistas ha observado que el programa no funciona correctamente. Según reportan, han ejecutado el programa con los mismos datos y unas veces proporciona las personas que realmente compraron en 3 o más tiendas, pero en otras ocasiones el programa sólo emite alguna de esas personas. De todos los datos que hay en producción, han reportado que el defecto se puede reproducir con sólo 104 datos que están disponibles en casoDePrueba.txt, en la carpeta PersonasQueCompranEnMuchasTiendas. La salida esperara es Alice, Carol y Dave. Sin embargo, cuando el programa se ejecuta en producción hay ocasiones en las que emite correctamente a esas tres personas, pero en otras ocasiones sólo emite Alice y Dave.

Depuración: el equipo de pruebas ha utilizado una herramienta de localización y de reducción de datos para depurar el programa. Han obtenido lo siguiente:

- El defecto ocurre cuando se ejecutan >1 Combiners.
- El defecto se manifiesta en la siguiente configuración con sólo 3 datos: ver imagen reduccion.jpg

viu Universidad Internacional de Valencia

Se tiene que analizar el programa para entender el defecto y posteriormente realizar un informe llamado PersonasQueCompranEnMuchasTiendas.pdf que contenga lo siguiente:

1) Nombre y apellidos del estudiante.
2) Tiempo empleado en entender el defecto del programa.
3) Descripción del defecto:
  a) Circunstancias bajo las que falla el programa: se tiene que indicar en qué ejecuciones podría fallar el programa.
  b) Motivos por los que falla el programa: se tiene que describir qué es lo que tiene erróneo el programa y que lo hace fallar.
  c) Directrices para corregir el defecto: se tiene que indicar a grandes rasgos lo que tendría que cambiar el equipo de desarrollo para eliminar el defecto del programa. No hace falta desarrollar el programa correcto, pero sí hay que indicar qué se tendría que cambiar.
4) ¿Te fue útil la información de depuración (la imagen reduccion.jpg y que el defecto se encontraba en >1 Combiners) para entender el defecto? Sí/No e indicar el por qué.

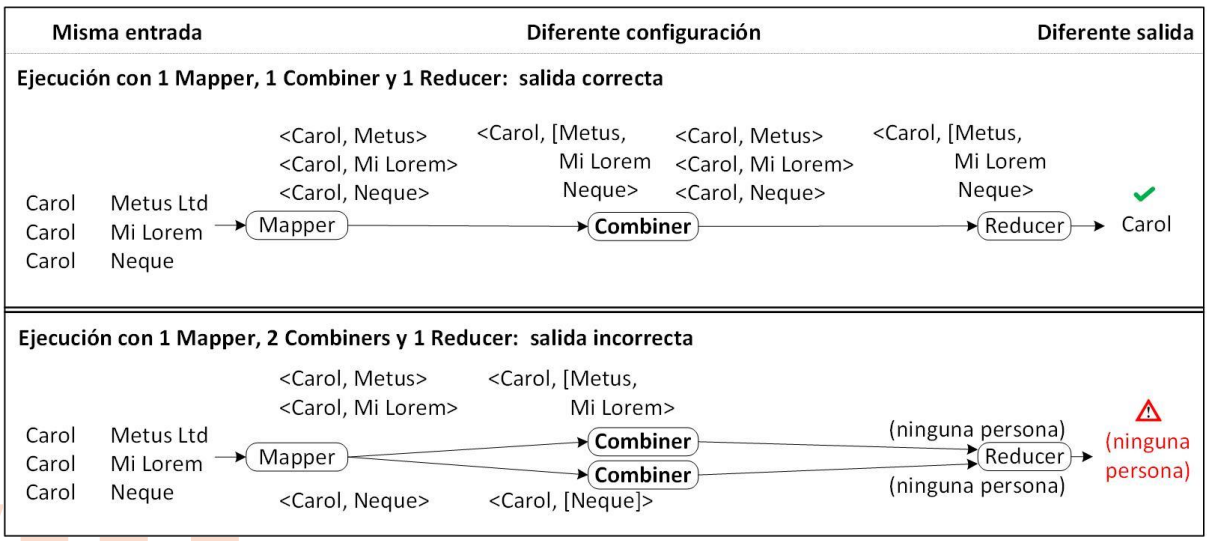
Actividades a elaborar:

Se tiene que analizar el programa para entender el defecto y posteriormente realizar un informe llamado PersonasQueCompranEnMuchasTiendas.pdf que contenga lo siguiente:

1) Nombre y apellidos del estudiante.
2) Tiempo empleado en entender el defecto del programa.
3) Descripción del defecto:
   a) Circunstancias bajo las que falla el programa: se tiene que indicar en qué ejecuciones podría fallar el programa.
   b) Motivos por los que falla el programa: se tiene que describir qué es lo que tiene erróneo el programa y que lo hace fallar.
   c) Directrices para corregir el defecto: se tiene que indicar a grandes rasgos lo que tendría que cambiar el equipo de desarrollo para eliminar el defecto del programa. No hace falta desarrollar el programa correcto, pero sí hay que indicar qué se tendría que cambiar.
4) ¿Te fue útil la información de depuración (la imagen reduccion.jpg y que el defecto se encontraba en >1 Combiners) para entender el defecto? Sí/No e indicar el por qué.

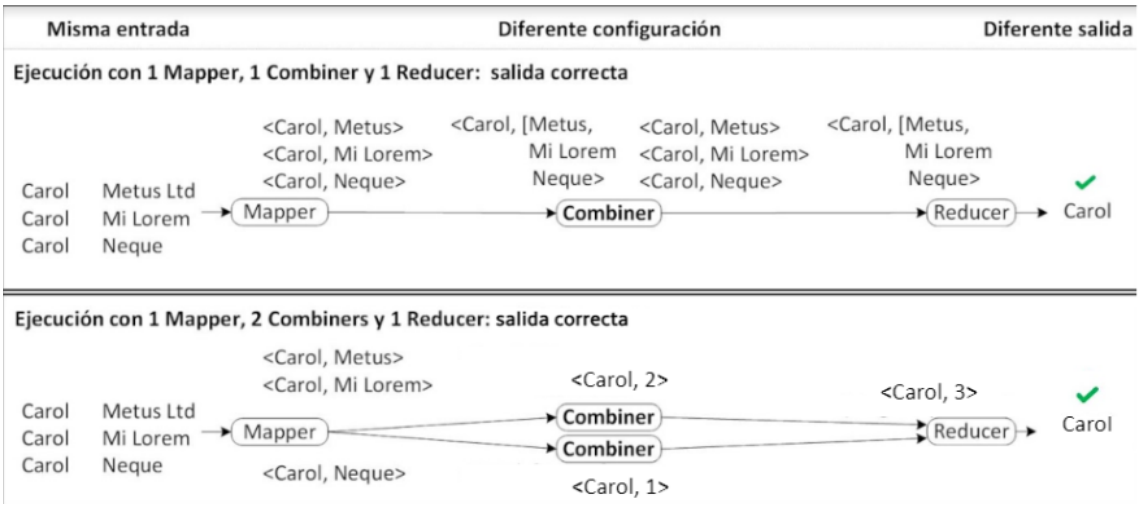
Solución:

1) Adrián Hernández Padrón y Andrea San Blas Hernández.
2) Tiempo de entendimiento del defecto: 10 minutos.
3) Analizando los 3 programas llegamos a la conclusión de que el defecto se encuentra dentro del programa combinerPersonasQueCompranEnMuchasTiendas.py. Cómo bien se menciona en el enunciado la falla no se encuentra en un fallo de código, sino en el entendimiento de cómo opera el framework mapreduce.
   a) Cuando la respuesta sale del mapper, esta se reparte en los diferentes combiners que el framework mapreduce determina. Tal como se muestra en la imagen en el caso de que solamente haya un combiner vamos a obtener la respuesta correcta, sin embargo cuando la información de Carol se reparta en diferentes combiners vamos a obtener una respuesta errónea.



b) Los motivos por los que el programa falla es que estamos realizando un filtrado de la información en el combiner, este combiner está programado para que solo pase información al reducer de aquellas personas que hayan comprado en 3 o

más tiendas. Como vemos en la imagen anterior Carol ha comprado en tres tiendas y en el primer caso lo identifica bien, porque toda la información de Carol cae sobre el mismo combiner. Sin embargo, en el segundo caso la información de Carol se divide en dos combiners, aplicando el mismo filtro, la información no va a pasar al reducer puesto que para estos dos combiner Carol no cumple la condición de que haya comprado en 3 o más tiendas, debido a que operan por separados el uno del otro.



c) La solución a este problema sería establecer un combiner que actúe como contador, es decir que cuente en cuantas tiendas ha comprado, esta información será pasada al reducer que será el que aplique el filtro y muestre el resultado final.

4) Si, gracias a la imagen se pudo entender perfectamente que el error estaba en el combiner y después de analizar el código de este llegamos al error del programa.