Hands-on I Big Data & Cloud Computing

Magister en Data Science, Universidad del Desarrollo

[redacted]
[redacted]

Objetivos

En esta actividad los alumnos deben realizar tareas relacionadas con la exploración de un dataset llamado GDELT (https://www.gdeltproject.org/), que contiene eventos extraídos de sitios web de noticias. Cada evento se normaliza y se cataloga de manera estandarizada, permitiendo su exploración y análisis mediante herramientas computacionales. Las tareas que se deben realizar son de dos tipos, entendimiento y exploración del dataset. En la primera se deben responder preguntas relacionadas a la información contenida, el número de archivos, columnas, filas y tamaños. En la segunda parte se deben responder preguntas de mayor complejidad, utilizando las herramientas Hive y Pig.

Tareas a realizar

Parte 1: Entender el dataset GDELT

1. ¿Qué información contiene el dataset GDELT? (1 punto)
2. ¿Cuál es el número total de archivos? (1 punto)
3. ¿Cuántas columnas tiene cada tipo de archivo (export, mentions o gkg)? (1 punto)
4. ¿Cuál es el tamaño promedio de cada tipo de archivo? (2 puntos)
5. ¿Cuántas filas en promedio tiene cada tipo de archivo? (2 puntos)
6. ¿Existe algún archivo cuyo peso en bytes esté fuera de rango? Debe definir qué significa fuera de rango. (3 puntos)

*Parte 2: **Exploración del dataset** con Hive y Pig*

1. ¿Cuál fue el sitio web con mayor cantidad de eventos registrados el 6 de Octubre del 2020?
    a. Implemente la consulta usando Pig (preguntar si los archivos ya fueron transformados a un formato legible por otro alumno) (3 puntos)
    b. Implemente la consulta usando Hive (preguntar si la tabla ya fue creada por otro alumno) (3 puntos)
2. ¿Cuál es la ciudad con mayor número de apariciones en los eventos registrados el 6 de Octubre del 2020?
    a. Implemente la consulta usando Pig (preguntar si los archivos ya fueron transformados a un formato legible por otro alumno) (3 puntos)
    b. Implemente la consulta usando Hive (preguntar si la tabla ya fue creada por otro alumno) (3 puntos)
3. ¿Cuál fue el evento generado durante Noviembre del 2020 con mayor cantidad de menciones en los medios?
    a. Implemente la consulta usando Pig (preguntar si los archivos ya fueron transformados a un formato legible por otro alumno) (6 puntos)
    b. Implemente la consulta usando Hive (preguntar si la tabla ya fue creada por otro alumno) (6 puntos)
4. ¿Cuáles son los 10 sitios web que generaron en promedio la mayor cantidad de menciones poco confiables (<40% de confianza) durante Noviembre del 2020?

    a. Implemente la consulta usando Pig (preguntar si los archivos ya fueron transformados a un formato legible por otro alumno) (6 puntos)

    b. Implemente la consulta usando Hive (preguntar si la tabla ya fue creada por otro alumno) (6 puntos)

5. ¿Cuáles son los 10 sitios web que generaron en promedio la mayor cantidad de menciones muy confiables (>80% de confianza) durante Septiembre, Octubre, Noviembre y Diciembre del 2020? (promedio de todas las apariciones durante los 4 meses)

    a. Implemente la consulta usando Pig (preguntar si los archivos ya fueron transformados a un formato legible por otro alumno) (12 puntos)

    b. Implemente la consulta usando Hive (preguntar si la tabla ya fue creada por otro alumno) (12 puntos)

6. Diseñe una consulta que le parezca relevante y que involucre cruzar los datasets export y mentions usando la llave GlobalEventID (primeros campos en ambos).

    a. Implemente la consulta en Pig o Hive (3 puntos)

Entregables

Se pueden realizar las tareas en grupos de 2 o 3 personas. Cada grupo debe entregar un documento detallando el proceso realizado para responder cada pregunta, incluyendo screenshots de los resultados dentro de la plataforma de cómputo (screenshot a la pantalla completa y un zoom a cada parte de interés). Los comandos o el código utilizados en cada pregunta deben ser explicados paso a paso (screenshots de apoyo también pueden ser incluídos de manera opcional).

Para realizar las tareas, debe utilizar un cluster Hadoop con algún proveedor de servicios cloud. En particular, se recomienda el servicio Dataproc de Google Cloud, Azure HDInsight, o EMR de AWS. Si por alguna razón no puede crear una cuenta gratis con algunos de estos proveedores, contáctese con el profesor. Advertencia: no utilice su tarjeta de crédito normal para activar una cuenta, se recomienda usar tarjetas de crédito virtuales con un mínimo de USD habilitado (por ejemplo 10 USD).

Se recomienda que en cada grupo de alumnos al menos un integrante tenga activado un servicio en alguno de los proveedores cloud, y entre todos gestionen el uso de los recursos de manera eficiente (eliminar/detener el cluster si no lo están utilizando).

Fecha de entrega

Lunes 26 de Abril antes de las 23:59 hrs, a través de la sección Tareas de Canvas.

Consultas

A través del foro de Canvas o directamente al profesor al correo o.peredo@udd.cl.

Ejemplo

Como ejemplo, se mostrará el paso a paso para responder una pregunta parecida a las planteadas en las tareas, utilizando datos del 2018 y una cuenta en Microsoft Azure (cluster HDInsight).

¿Cuál fue el evento con mayor cantidad de menciones muy confiables (>60% de confianza) durante Junio del 2018?

Solución:

El primer problema que se debe resolver es la transformación de formato de los archivos en el storage en la nube. Los archivos de Junio 2018 están en la siguiente carpeta:

```
sshuser@hn0-cluste:~$ hadoop fs -ls wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/ | head

Found 7617 items

-rw-r--r--   1 sshuser supergroup     309238 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601000000.export.CSV.zip

-rw-r--r--   1 sshuser supergroup   10782365 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601000000.gkg.csv.zip

-rw-r--r--   1 sshuser supergroup     267582 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601000000.mentions.CSV.zip

-rw-r--r--   1 sshuser supergroup     255695 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601001500.export.CSV.zip

-rw-r--r--   1 sshuser supergroup   11760180 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601001500.gkg.csv.zip

-rw-r--r--   1 sshuser supergroup     289261 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601001500.mentions.CSV.zip

-rw-r--r--   1 sshuser supergroup     220110 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601003000.export.CSV.zip

-rw-r--r--   1 sshuser supergroup   12115188 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601003000.gkg.csv.zip

-rw-r--r--   1 sshuser supergroup     281508 2019-04-06 16:42 wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/20180601003000.mentions.CSV.zip
```

El script bash realiza la conversión a texto plano y almacenamiento en la ruta similar gdelt/2018/06 es como sigue:

```
hadoop fs -copyToLocal wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/*mentions*zip .

for file in `ls *mentions*zip | sed 's/.zip//g'`

do

        unzip ${file}.zip

done

hadoop fs -copyFromLocal -f *mentions*CSV  wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/
```

Pig

Ya teniendo los archivos en formato texto plano en el storage en la nube, primero vamos a implementar el script Pig llamado `script.pig` que genera la respuesta a la pregunta:

```
set tez.runtime.io.sort.mb 1091;

DATA = LOAD 'wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/201806*mentions*' using
PigStorage('\t');

ONLYCONFID = foreach DATA generate $0,$11;

HIGHCONFID = filter ONLYCONFID by $1>60;

GG = group HIGHCONFID by $0;

CC = foreach GG generate group, COUNT(HIGHCONFID);

ORDERBYCOUNT = order CC by $1 desc;

TOPCOUNT = limit ORDERBYCOUNT 1;

dump TOPCOUNT;
```

La ejecución del script Pig es de la siguiente manera en la línea de comandos:

```
sshuser@hn0-cluste:~$ pig script.pig
```

Y el resultado, con GlobalEventID **764935245** y **1285** menciones con confianza >60%, aparece en el dump desplegado en la salida standard:

```
…

Input(s):

Successfully read 15958341 records from: "wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06/201806*mentions*"


Output(s):

Successfully stored 10 records (180 bytes) in: "wasb://hands-on@magisterudd.blob.core.windows.net/tmp/temp1208712681/tmp1346498370"


2019-04-09 04:27:11,129 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1

2019-04-09 04:27:11,129 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1

(764935245,1285)

2019-04-09 04:27:51,726 [main] INFO  org.apache.pig.Main - Pig script completed in 3 minutes, 58 seconds and 285 milliseconds (238285 ms)

…
```

Hive

Ahora vamos a crear la tabla Hive para poder realizar la consulta en esta herramienta. Como paso previo, tenemos que crear una carpeta especial con los archivos “mentions” y copiar en ella todos los archivos de Junio 2018 de ese tipo en formato texto plano. Esto se requiere para poder crear una tabla externa que se cargue con los datos directamente de una carpeta. Creamos un archivo llamado `create_table.hql` que contiene lo siguiente:

```
create external table 201806mentions (
        GlobalEventID int,
        EventTimeDate int,
        MentionTimeDate int,
        MentionType int,
        MentionSourceName int,
        MentionIdentifier int,
        SentenceID int,
        Actor1CharOffset int,
        Actor2CharOffset int,
        ActionCharOffset int,
        InRawText int,
        Confidence int,
        MentionDocLen int,
        MentionDocTone int,
        MentionDocTranslationInfo string,
        Extras string
)
row format delimited
fields terminated by '\t'
lines teminated by '\n'
stored as textfile
location 'wasb://hands-on@magisterudd.blob.core.windows.net/gdelt/2018/06_mentions/';
```

Para ejecutar este script, tenemos que conectarnos al cliente Hive, llamado Beeline, de la siguiente manera:

```
sshuser@hn0-cluste:~$ beeline -u 'jdbc:hive2://hn0-cluste:10001/;transportMode=http'
Connecting to jdbc:hive2://hn0-cluste:10001/;transportMode=http
Connected to: Apache Hive (version 1.2.1000.2.6.5.3006-29)
Driver: Hive JDBC (version 1.2.1000.2.6.5.3006-29)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.2.1000.2.6.5.3006-29 by Apache Hive
0: jdbc:hive2://hn0-cluste:10001/>
```

Dentro del cliente podemos ingresar cada línea por separado (el prompt en este caso es el símbolo '>'), o podemos ejecutar el script completo de la siguiente manera:

```
sshuser@hn0-cluste:~$ beeline -u 'jdbc:hive2://hn0-cluste:10001/;transportMode=http' -f create_table.hql
Connecting to jdbc:hive2://hn0-cluste:10001/;transportMode=http
Connected to: Apache Hive (version 1.2.1000.2.6.5.3006-29)
Driver: Hive JDBC (version 1.2.1000.2.6.5.3006-29)
Transaction isolation: TRANSACTION_REPEATABLE_READ
0: jdbc:hive2://hn0-cluste:10001/> create external table 201806mentions (
0: jdbc:hive2://hn0-cluste:10001/>     GlobalEventID int,
0: jdbc:hive2://hn0-cluste:10001/>     EventTimeDate int,
0: jdbc:hive2://hn0-cluste:10001/>     MentionTimeDate int,
0: jdbc:hive2://hn0-cluste:10001/>     MentionType int,
0: jdbc:hive2://hn0-cluste:10001/>     MentionSourceName int,
0: jdbc:hive2://hn0-cluste:10001/>     MentionIdentifier int,
0: jdbc:hive2://hn0-cluste:10001/>     SentenceID int,
0: jdbc:hive2://hn0-cluste:10001/>     Actor1CharOffset int,
0: jdbc:hive2://hn0-cluste:10001/>     Actor2CharOffset int,
0: jdbc:hive2://hn0-cluste:10001/>     ActionCharOffset int,
0: jdbc:hive2://hn0-cluste:10001/>     InRawText int,
0: jdbc:hive2://hn0-cluste:10001/>     Confidence int,
0: jdbc:hive2://hn0-cluste:10001/>     MentionDocLen int,
0: jdbc:hive2://hn0-cluste:10001/>     MentionDocTone int,
0: jdbc:hive2://hn0-cluste:10001/>     MentionDocTranslationInfo string,
0: jdbc:hive2://hn0-cluste:10001/>     Extras string
0: jdbc:hive2://hn0-cluste:10001/> )
0: jdbc:hive2://hn0-cluste:10001/> row format delimited
0: jdbc:hive2://hn0-cluste:10001/> fields terminated by '\t'
0: jdbc:hive2://hn0-cluste:10001/> stored as textfile
0: jdbc:hive2://hn0-cluste:10001/> location 'wasb://hands-
on@magisterudd.blob.core.windows.net/gdelt/2018/06_mentions/';
No rows affected (2.239 seconds)
0: jdbc:hive2://hn0-cluste:10001/>
0: jdbc:hive2://hn0-cluste:10001/>
Closing: 0: jdbc:hive2://hn0-cluste:10001/;transportMode=http
```

Con esto podemos observar la nueva tabla con la siguiente consulta dentro de Beeline:

```
sshuser@hn0-cluste:~$ beeline -u 'jdbc:hive2://hn0-cluste:10001/;transportMode=http'
Connecting to jdbc:hive2://hn0-cluste:10001/;transportMode=http
Connected to: Apache Hive (version 1.2.1000.2.6.5.3006-29)
Driver: Hive JDBC (version 1.2.1000.2.6.5.3006-29)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 1.2.1000.2.6.5.3006-29 by Apache Hive
0: jdbc:hive2://hn0-cluste:10001/> show tables;
+------------------+--+
|     tab_name     |
+------------------+--+
| 201806mentions   |
| hivesampletable  |
+------------------+--+
3 rows selected (0.813 seconds)
0: jdbc:hive2://hn0-cluste:10001/>
```

Y finalmente ejecutamos la consulta, guardada en el archivo query.hql, como sigue:

```
sshuser@hn0-cluste:~$ beeline -u 'jdbc:hive2://hn0-cluste:10001/;transportMode=http' -f query.hql
Connecting to jdbc:hive2://hn0-cluste:10001/;transportMode=http
Connected to: Apache Hive (version 1.2.1000.2.6.5.3006-29)
Driver: Hive JDBC (version 1.2.1000.2.6.5.3006-29)
Transaction isolation: TRANSACTION_REPEATABLE_READ
0: jdbc:hive2://hn0-cluste:10001/> select * from
0: jdbc:hive2://hn0-cluste:10001/> (
0: jdbc:hive2://hn0-cluste:10001/>     select GlobalEventID, COUNT(Confidence) as cnt
0: jdbc:hive2://hn0-cluste:10001/>     from 201806mentions
0: jdbc:hive2://hn0-cluste:10001/>     where Confidence>60
0: jdbc:hive2://hn0-cluste:10001/>     group by GlobalEventID
0: jdbc:hive2://hn0-cluste:10001/> ) a
0: jdbc:hive2://hn0-cluste:10001/> order by cnt desc
0: jdbc:hive2://hn0-cluste:10001/> limit 1;
INFO  : Tez session hasn't been created yet. Opening session
…
+------------------+--------+--+
| a.globaleventid  | a.cnt  |
+------------------+--------+--+
| 764935245        | 1285   |
+------------------+--------+--+
1 row selected (59.935 seconds)
0: jdbc:hive2://hn0-cluste:10001/>
```

```
0: jdbc:hive2://hn0-cluste:10001/>

Closing: 0: jdbc:hive2://hn0-cluste:10001/;transportMode=http
```

Búsqueda de GlobalEventID en archivos “export”

Teniendo el ID del evento, procedemos a buscarlo en los archivos “export”, también convertidos a formato texto plano de la misma manera que los archivos “mentions”:

```
sshuser@hn0-cluste:~$ hadoop fs -cat wasb://hands-
on@magisterudd.blob.core.windows.net/gdelt/2018/06/201806*export* | grep 764935245

764935245       20180617        201806  2018    2018.4575       AFGINSTAL       TALIBAN AFG     TAL
INS                     1190    190     19      4       -10.0   24      4       24      -6.88433758066071
4       Jalalabad, Nangarhar, Afghanistan       AF      AF18    9992834.4265 70.4515 -3377673        0
4       Jalalabad, Nangarhar, Afghanistan       AF      AF1899928    34.4265 70.4515 -3377673
20180617113000  https://www.journal-news.com/news/world/the-latest-suicide-bombing-afghanistan-
kills/yVnwhBtVPHGxtzEucP31SJ/
```

Herramientas

Línea de comandos Linux

`cd`: cambiar de directorio.

`ls`: listar archivos en un directorio.

`cat`: imprimir contenido de un archivo en la salida standard (pantalla).

`head`: imprimir las primeras líneas de un archivo en la salida standard.

`tail`: imprimir las últimas líneas de un archivo en la salida standard.

`more`: imprimir contenido de un archivo por partes de manera incremental.

`wc`: contar bytes, caracteres, palabras o filas de un archivo.

`sort`: ordenar un archivo según orden lexicográfico o numérico.

`uniq`: eliminar o mostrar las líneas repetidas de un archivo.

`awk`: lenguaje de programación para procesamiento y detección de patrones en texto.

`grep`: herramienta para detección y transformación de patrones usando expresiones regulares.

`sed`: herramienta para detección y transformación de patrones usando expresiones regulares.

Ejemplos de comandos “one-liner” en Linux

Conteo de filas:

```
cat file.txt | wc -l
```

Conteo de columnas (con separador ‘,’):

```
cat file.txt | awk –F ‘,’ ‘{print NF}’ | sort | uniq –c
```

Conteo de columnas (con separador espacio):

```
cat file.txt | awk ‘{print NF}’ | sort | uniq –c
```

Imprimir la segunda columna de todas las filas (con separador ‘,’):

```
cat file.txt | awk –F ‘,’ ‘{print $2}’
```

Conteo de filas con la misma llave (ejemplo de fila con llave en la primera columna: id1,val1,val2,val3):

```
cat file.txt | awk -F ',' '{map[$1]+=1}END{for(key in map){print key,map[key]}}'
```

Suma de los valores en la primera columna (con separador ','):

```
cat file.txt | awk -F ',' '{sum+=$1}END{print sum}'
```

Filtrar filas con la palabra error (sólo minúsculas):

```
cat file.txt | grep 'error'
```

Seleccionar filas con la palabra error (minúsculas o mayúsculas):

```
cat file.txt | grep '[Ee][Rr][Rr][Oo][Rr]'
```

Remover filas con la palabra error (sólo minúsculas):

```
cat file.txt | grep -v 'error'
```

Remover filas con la palabra error (minúsculas o mayúsculas), simplificado:

```
cat file.txt | grep -iv 'error'
```

Calcular el promedio de los valores en la segunda columna (con separador ','):

```
cat file.txt | awk -F ',' '{sum+=$2; counter+=1}END{print sum/counter}'
```

Calcular la desviación estándar de los valores en la segunda columna (con separador ','):

```
cat file.txt | awk -F ',' '{sum+=$2; counter+=1; squaresum+=($2*$2)}END{print
sqrt( (squaresum/counter) - (sum/counter)^2 )}'
```

Comandos en el filesystem Hadoop

`hadoop fs -ls ruta`: listar archivos en un directorio en el filesystem distribuído.

`hadoop fs -cat ruta/archivo`: imprimir contenido de un archivo (texto plano) en el filesystem distribuído en la salida standard (pantalla).

`hadoop fs -text ruta/archivo`: imprimir contenido de un archivo (comprimido o serializado) en el filesystem distribuído en la salida standard (pantalla).

`hadoop fs -copyFromLocal archivo ruta/` : copiar un archivo en el filesystem local (linux) hacia el filesystem distribuído.

`hadoop fs -copyToLocal ruta/archivo .` : copiar un archivo desde el filesystem distribuído hacia el filesystem local (Linux), en la carpeta actual.

Referencias

https://es.hortonworks.com/tutorial/beginners-guide-to-apache-pig/

https://es.hortonworks.com/tutorial/how-to-process-data-with-apache-hive/

http://yahoohadoop.tumblr.com/post/98256601751/pig-and-hive-at-yahoo

"Programming Pig", 2nd Edition, Alan Gates and Daniel Dai (2017)

"Programming Hive", 1st Edition, Dean Wampler, Jason Rutherglen and Edward Capriolo (2012)

"Processing Big Data with Azure HDInsight", 1st Edition, Vinit Yadav (2017)