

(RABDA.2 / CEBDA.2b / 2p)

Vuelve a contar las palabras que tiene El Quijote, pero haciendo usos de los scripts Python, teniendo en cuenta que el proceso de mapeo va a limpiar las palabras de signos ortográficos (quitar puntos, comas, paréntesis) y en el reducer vamos a considerar que las palabras en mayúsculas y minúsculas son la misma palabra.







Tip: para la limpieza, puedes utilizar el método de string translate de manera que elimine las string.punctuation.

Debes ejecutar ambos script como procesos MapReduce mediante Hadoop Streaming y comprobar en HDFS el archivo que se ha creado.

Este ejercicio lo tengo hecho en clase, voy a utilizar el código para que lo puedas ver y si tienes alguna sugerencia estaré encantado de conocerla (imparto el módulo de Big Data Aplicado).

Utilizo la librería nltk así elimino signos de puntuación y artículos, preposiciones....

Estoy con un Mac M1, utilizo como sistema de virtualización un software llamado UTM, gracias al cual puedo emular una debian 64bits pero me va muy mal.  
Lanzar el mapreduce a través de hadoop son unos 10-15 minutos...

|                                                                                     |                              |                                          |
|-------------------------------------------------------------------------------------|------------------------------|------------------------------------------|
|    | <b>Estado</b>                | Iniciado                                 |
|  | <b>Arquitectura</b>          | x86_64                                   |
|  | <b>Máquina</b>               | Standard PC (Q35 + ICH9, 2009) (alias... |
|  | <b>Memoria</b>               | 4 GB                                     |
|  | <b>Tamaño</b>                | 13,25 GB                                 |
|  | <b>Directorio compartido</b> |                                          |

## Mapper

```
hadoop@debianh:~/quijote$ cat mapper.py
#!/usr/bin/env python3
"""mapper.py"""

import sys
import string
import nltk
nltk.download('stopwords',quiet=True)
from nltk.corpus import stopwords
stop_words = set(stopwords.words('spanish'))
punctuations = '''!()-[]{};:'"\, <> ./?@#$$%^&*~'''

# input comes from STDIN (standard input)
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip()
    # split the line into words
    words = line.split()
    # increase counters
    for word in words:
        word = word.translate(str.maketrans('', '', string.punctuation)).lower()

        if word not in stop_words:
            print('%s\t%s' % (word, 1))
#
#
##### Instalar nltk
##### python3.11 -m pip install nltk
##### apt install python3-nltk
```

## Reducer

En mi caso para el reducer no utilizo diccionario ya que Hadoop después del mapper hace un sort y ya no es necesario, por lo que cuento las palabras que van saliendo línea a línea (ya que están ordenadas).

Para ejecutarlo en bash emulo el sort de Hadoop:

```
$ cat quijote.txt | python3 mapper.py | sort | python3 reducer.py > result.txt
```

```
hadoop@debian:~/quijote$ tail result.txt
zoroástrica 1
zorra 1
zorras 1
zorrana 1
zuecos 1
zulema 1
zumban 1
zurdo 2
zurrón 1
zuzaban 1
```

```
#!/usr/bin/env python3
"""reducer.py"""

import sys

word = None
current_word = None
current_count = 0

# input comes from STDIN (standard input)
for line in sys.stdin:
    data = line.strip().split("\t")

    if len(data) != 2:
        continue
    word, count = data

    try:
        count = int(count)
    except ValueError:
        continue

    if current_word == word:
        current_count += count
    else:
        if current_word: # Primera iteración es None
            print('%s\t%s' % (current_word, current_count))
            current_word = word
            current_count = 1

if current_word == word:
    print('%s\t%s' % (current_word, current_count))
```

## Ejecución a través de Hadoop

hdfs dfs -put quijote.txt /

\$ hadoop jar /opt/hadoop/share/hadoop/tools/lib/hadoop-streaming-3.3.6.jar -file mapper.py  
-mapper mapper.py -file reducer.py -reducer reducer.py -input /quijote.txt -output /quijote\_salidaX

```
2024-04-03 22:43:33,229 INFO mapred.FileInputFormat: Total input files to process : 1
2024-04-03 22:43:33,971 INFO mapreduce.JobSubmitter: number of splits:2
2024-04-03 22:43:37,656 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1712175529914_0002
2024-04-03 22:43:37,658 INFO mapreduce.JobSubmitter: Executing with tokens: []
2024-04-03 22:43:39,648 INFO conf.Configuration: resource-types.xml not found
2024-04-03 22:43:39,652 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2024-04-03 22:43:41,710 INFO impl.YarnClientImpl: Submitted application application_1712175529914_0002
2024-04-03 22:43:42,377 INFO mapreduce.Job: The url to track the job: http://debianh:8088/proxy/application_1712175529914_0002/
2024-04-03 22:43:42,432 INFO mapreduce.Job: Running job: job_1712175529914_0002
2024-04-03 22:45:20,521 INFO mapreduce.Job: Job job_1712175529914_0002 running in uber mode : false
2024-04-03 22:45:20,563 INFO mapreduce.Job: map 0% reduce 0%
2024-04-03 22:47:18,168 INFO mapreduce.Job: map 8% reduce 0%
2024-04-03 22:47:25,717 INFO mapreduce.Job: Task Id : attempt_1712175529914_0002_m_000001_0, Status : FAILED
Error: java.lang.RuntimeException: PipeMapRed.waitOutputThreads(): subprocess failed with code 1
    at org.apache.hadoop.streaming.PipeMapRed.waitOutputThreads(PipeMapRed.java:326)
    at org.apache.hadoop.streaming.PipeMapRed.mapRedFinished(PipeMapRed.java:539)
    at org.apache.hadoop.streaming.PipeMapper.close(PipeMapper.java:130)
    at org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:61)
    at org.apache.hadoop.streaming.PipeMapRunner.run(PipeMapRunner.java:34)
    at org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:466)
    at org.apache.hadoop.mapred.MapTask.run(MapTask.java:350)
    at org.apache.hadoop.mapred.YarnChild$2.run(YarnChild.java:178)
```

(RABDA.4 / CEBDA.4a / 1p) Entra en Hadoop UI y en YARN, y visualiza los procesos que se han ejecutado en las actividades 1 y 2, comprobando la configuración tanto en Hadoop UI como en YARN así como la ejecución de los jobs arrancando el Job History Server.



Logged in as: dr:who

## Application Attempt appattempt\_1712175529914\_0002\_000001

- Cluster
- about
- nodes
- node Labels
- applications
- NEW
- NEW SAVING
- SUBMITTED
- ACCEPTED
- RUNNING
- FINISHED
- FAILED
- KILLED
- Scheduler
- tools

| Application Attempt Overview          |                                                        |
|---------------------------------------|--------------------------------------------------------|
| Application Attempt State:            | RUNNING                                                |
| Started:                              | mié. abr. 03 22:43:41 +0200 2024                       |
| Elapsed:                              | 2mins, 19sec                                           |
| AM Container:                         | <a href="#">container_1712175529914_0002_01_000001</a> |
| Node:                                 | debianh:42025                                          |
| Tracking URL:                         | <a href="#">ApplicationMaster</a>                      |
| Diagnostics Info:                     |                                                        |
| Nodes blacklisted by the application: | -                                                      |
| Nodes blacklisted by the system:      | -                                                      |

| Application Attempt Metrics                            |  |
|--------------------------------------------------------|--|
| Application Attempt Headroom : <memory:4096, vCores:5> |  |

|                                                                                                                                                 |                     |                       |                    |
|-------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|-----------------------|--------------------|
| Total Allocated Containers: 3                                                                                                                   |                     |                       |                    |
| Each table cell represents the number of NodeLocal/RackLocal/OffSwitch containers satisfied by NodeLocal/RackLocal/OffSwitch resource requests. |                     |                       |                    |
|                                                                                                                                                 | Node Local Request  | Rack Local Request    | Off Switch Request |
| Num Node Local Containers (satisfied by)                                                                                                        | 2                   |                       |                    |
| Num Rack Local Containers (satisfied by)                                                                                                        | 0                   | 0                     |                    |
| Num Off Switch Containers (satisfied by)                                                                                                        | 0                   | 0                     | 1                  |
| Show 20 entries Search:                                                                                                                         |                     |                       |                    |
| Container ID                                                                                                                                    | Node                | Container Exit Status | Logs               |
| container_1712175529914_0002_01_000003                                                                                                          | http://debianh:8042 | 0                     | Logs               |
| container_1712175529914_0002_01_000002                                                                                                          | http://debianh:8042 | 0                     | Logs               |
| container_1712175529914_0002_01_000001                                                                                                          | http://debianh:8042 | 0                     | Logs               |
| Showing 1 to 3 of 3 entries First Previous 1 Next Last                                                                                          |                     |                       |                    |

← → ↻ No es seguro 192.168.66.4:8042/node/allApplications



## Applications running on this node

- ResourceManager
- RM Home
- NodeManager
- Tools

|                                                        |                  |
|--------------------------------------------------------|------------------|
| Show 20 entries Search:                                |                  |
| ApplicationId                                          | ApplicationState |
| application_1712175529914_0002                         | RUNNING          |
| Showing 1 to 1 of 1 entries First Previous 1 Next Last |                  |

|                          |            |        |            |     |              |   |     |                 |  |
|--------------------------|------------|--------|------------|-----|--------------|---|-----|-----------------|--|
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Nov 15 17:13 | 0 | 0 B | quijote_salida2 |  |
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Nov 15 18:21 | 0 | 0 B | quijote_salida3 |  |
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Nov 15 18:22 | 0 | 0 B | quijote_salida4 |  |
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Nov 15 18:46 | 0 | 0 B | quijote_salida5 |  |
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Nov 15 18:53 | 0 | 0 B | quijote_salida6 |  |
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Nov 15 19:02 | 0 | 0 B | quijote_salida7 |  |
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Nov 15 19:23 | 0 | 0 B | quijote_salida8 |  |
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Nov 15 19:31 | 0 | 0 B | quijote_salida9 |  |
| <input type="checkbox"/> | drwxr-xr-x | hadoop | supergroup | 0 B | Apr 03 22:45 | 0 | 0 B | quijote_salidaX |  |

(RABDA.3 / CEBDA.3a y CEBDA.3c / 0.5p)

En este ejercicio vamos a practicar los comandos básicos de HDFS. Una vez arrancado Hadoop:

Ejecuto alguno con time para que puedas ver la velocidad de la MV...

**Crea la carpeta /user/iabd/ejercicios.**

```
time hdfs dfs -mkdir -p /user/iabd/ejercicios
```

```
hadoop@debianh:~/quijote$ time hdfs dfs -mkdir -p /user/iabd/ejercicios
real    0m23,189s
user    0m20,083s
sys     0m1,547s
```

**Sube el archivo el\_quijote.txt a la carpeta creada.**

```
hadoop@debianh:~/quijote$ time hdfs dfs -put el_quijote.txt /user/iabd/ejercicios/
real    0m26,007s
user    0m21,298s
sys     0m1,630s
```

**Crea una copia en HDFS y llámala el\_quijote2.txt.**

```
hdfs dfs -cp /user/iabd/ejercicios/el_quijote.txt /user/iabd/ejercicios/el_quijote2.txt
```

**Recupera el principio del fichero el\_quijote2.txt.**

```
hdfs dfs -head /user/iabd/ejercicios/el_quijote2.txt
```

**Renombra el\_quijote2.txt a el\_quijote\_copia.txt.**

```
hdfs dfs -mv /user/iabd/ejercicios/el_quijote2.txt /user/iabd/ejercicios/el_quijote_copia.txt
```

**Descarga en local el\_quijote\_copia.txt con su código CRC.**

```
hdfs dfs -get -crc /user/iabd/ejercicios/el_quijote_copia.txt
```

**Adjunta una captura desde el interfaz web donde se vean ambos archivos.**

## Browse Directory

/user/iabd/ejercicios

Go!

Show

25

entries

Search:

| <input type="checkbox"/> | Permission | Owner  | Group      | Size    | Last Modified | Replication | Block Size | Name                 |  |
|--------------------------|------------|--------|------------|---------|---------------|-------------|------------|----------------------|--|
| <input type="checkbox"/> | -rw-r--r-- | hadoop | supergroup | 2.04 MB | Apr 03 23:11  | 1           | 128 MB     | el_quijote.txt       |  |
| <input type="checkbox"/> | -rw-r--r-- | hadoop | supergroup | 2.04 MB | Apr 03 23:12  | 1           | 128 MB     | el_quijote_copia.txt |  |

Showing 1 to 2 of 2 entries

Previous

1

Next

**Vuelve al terminal y elimina la carpeta con los archivos contenidos mediante un único comando.**

```
hdfs dfs -rm -rf /user/iabd/ejercicios
```