Sistemas de Gestión de Datos y de la Información
Máster en Ingeniería Informática, 2022-23
Ejercicio MapReduce

Uso de mrjob en el laboratorio (Linux)

Las tareas MapReduce de *mrjob* se definen implementando una clase que herede de `MRJob` y definiendo sus métodos `mapper`, `reducer` y opcionalmente `combiner` (ver el fichero `word_count_mr.py` y la documentación `https://mrjob.readthedocs.io/en/latest/guides/writing-mrjobs.html` para más detalles). Una vez se tiene implementada esta clase la tarea se lanza desde el **terminal** con el comando[1]:

```
$ /usr/local/python/anaconda3/bin/python word_count_mr.py texto.txt
```

Esta tarea tomará como entrada el fichero `texto.txt` y su salida se mostrará por la salida estándar. Podéis redirigir la salida estándar a un fichero, e incluso redirigir la salida de error por la que el sistema muestra distinta información de depuración:

```
$ /usr/local/python/anaconda3/bin/python python word_count_mr.py
   fichero1 fichero2 ... > salida.txt 2> error.txt
```

A) Felicidad

Tomaremos como partida el fichero `happiness.txt`, que contiene una valoración de *felicidad* para más de 10.000 palabras inglesas. Este fichero ha sido obtenido del artículo *Temporal patterns of happiness and information in a global social network: Hedonometrics and Twitter*, `http://arxiv.org/abs/1101.5120`. El archivo está separado por **tabuladores**, donde cada línea sigue el siguiente formato:

```
word happiness_rank happiness_average happiness_standard_deviation twitter_rank
google_rank nyt_rank lyrics_rank
```

Queremos encontrar las 5 palabras más *alegres* (su `happiness_average` es mayor) dentro del subconjunto de palabras *extremadamente tristes*, que son aquellas cuyo `happiness_average` está por debajo de 2. No nos interesan todas ellas, sino únicamente las que tienen ranking Twitter, es decir, `twitter_rank` es diferente de `--`. En nuestro pequeño listado de palabras queremos que aparezca también la *felicidad media* asociada a cada palabra.

[1] Si el intérprete `python` adecuado no está en el *path* deberéis proporcionar la ruta completa.

La salida debe tener un formato así *(los valores concretos no son reales)*:

```
susto       1.99
muerte      1.98
coronavirus 1.98
confinamiento 1.97
teorema     1.83
```

B) Datos meteorológicos

Utilizaremos los ficheros `JCMB_2014.csv` y `JCMB_2015.csv` que registran diversos parámetros recogidos por una estación meteorológica en la Universidad de Edimburgo durante los años 2014 y 2015, obtenidos de `https://www.geos.ed.ac.uk/~weather/jcmb_ws/`. El formato del fichero es de valores separados por comas (CSV) con una cabecera que indica su contenido:

```
date-time,atmospheric pressure (mBar),rainfall (mm),wind speed (m/s),
wind direction (degrees),surface temperature (C),relative humidity ( %),
solar flux (Kw/m2),battery (V)
```

Queremos procesar los ficheros para conocer en cada mes el nivel **mínimo**, **máximo** y **promedio** de la **batería**, obteniendo una salida como se muestra a continuación *(los valores no son reales)*:

```
...
10/2013  {'max': 15.83, 'avg': 11.734, 'min': 9.29}
02/2014  {'max': 13.56, 'avg': 12.234, 'min': 10.69}
...
```

C) Log de servidor web

Consideremos el fichero `access_log_Jul95` obtenido de `ftp://ita.ee.lbl.gov/traces/NASA_access_log_Jul95.gz`, que registra las peticiones HTTP recibidas por el *Centro Espacial John F. Kennedy* (CEK) de la NASA. Cada línea del *log* representa una petición HTTP, y está compuesta por varios campos. Nos interesan los siguientes campos:

- *Host* que realiza la petición (nombre o dirección IP). Es el primer campo de la línea.
- Código HTTP de respuesta: 200, 404, 302, etc. Es el penúltimo campo.
- Número de bytes de la contestación. Es el último campo.

Queremos construir un listado con información resumida por cada *host*: el número total de peticiones realizadas, el tamaño total de las contestaciones enviadas a ese *host*, y el número de peticiones que han recibido un error 4xx o 5xx. Supondremos que si el tamaño de una petición no está establecido en el *log* (es `-`) su tamaño es 0. La salida debe tener un formato similar al siguiente *(los valores son ficticios)*:

```
...
"bubu.com"          (1, 4888, 0)
"robot.1234.epa.gov" (1, 5287, 4)
...
```