hw1-code-337604821-326922390

July 11, 2024

[ ]:
```
!pip install mrjob
```

[11]:
```
%%file hw1_mrA_337604821_326922390.py

from mrjob.job import MRJob
from mrjob.step import MRStep
import re

def split_properley (line):
    item = line.split(',')
    date = item[-3] # taking third item from the righ to left on the row, which␣
↪represents the date
    air_time = int(item[-2]) if item[-2].isnumeric() else 0 #getting the␣
↪integer value fo the item second from the right side of the row

    if item[2] != item[-4]:# checking if there are multiple genres or only 1.
        genres = line.split('\"')[1].split(',') if item[0] != 'title' else␣
↪"Hello World"
    else:# There is only one genre for this tuple
      genres = [item[2]]
    title = item[0].strip()
    return title, genres, air_time, date

class MRWordFrequencyCount(MRJob):
    def steps(self):
        return [
            MRStep(mapper=self.mapper,
                   reducer=self.reducer),
            MRStep(reducer=self.reducer_count_genres),
            #MRStep(reducer=self.reducer_max)
        ]

    def mapper(self, _, line):
        title, genres, air_time, date = split_properley(line)# get relevant␣
↪data from the tuple
        try:
```

```
            if 70000 <= int(air_time) < 90000 and re.search('[jqz]', title.
↪lower()):# filter out by airtime and that title contains one of the letters␣
↪j,q,z
                for genre in genres:
                    if genre.strip() in ['Sitcom', 'Talk', 'Politics',␣
↪'Spanish', 'Community', 'Martial arts']:
                        ls = [title]
                        ls.extend(genres)
                        yield (tuple(ls), date)# if one of the genres is in the␣
↪list then we will move this tuple reformatted to the next step
        except Exception as e:
            pass

    def reducer(self, key, dates):
        yield key, len(set(dates))# reduce the value so it contains the number␣
↪of unique dates

    def reducer_count_genres(self, key, values):
        total_dates = 0
        for dates_count in values:
            total_dates += dates_count
        yield key, (total_dates, len(key)-1)# counting number of dates per key,␣
↪counting number of genres



if __name__ == '__main__':
    MRWordFrequencyCount.run()
```

```
Overwriting hw1_mrA_337604821_326922390.py
```

[12]:
```
!python hw1_mrA_337604821_326922390.py 420k_daily_prog_data.csv
```

```
No configs found; falling back on auto-configuration
No configs specified for inline runner
Creating temp directory
/tmp/hw1_mrA_337604821_326922390.root.20240710.125931.013212
Running step 1 of 2…
Running step 2 of 2…
job output is in
/tmp/hw1_mrA_337604821_326922390.root.20240710.125931.013212/output
Streaming final output from
/tmp/hw1_mrA_337604821_326922390.root.20240710.125931.013212/output…
["El Joven Ju\u00e1rez", "Spanish", "Biography"]        [1, 2]
["El Joven del Carrito", "Spanish", "Comedy"]   [1, 2]
["El Oreja Rajada", "Spanish", "Drama"] [1, 2]
["El Palenque", "Talk"] [1, 1]
```

["El Rediezcubrimiento de M\u00e9xico", "Spanish", "Comedy-drama"] [1, 2]
["El Santos vs la Tetona Mendoza", "Spanish", "Comedy", "Animated"] [1, 3]
["El Santos vs. la T…a Mendoza", "Spanish", "Comedy", "Animated"] [1, 3]
["El Tejedor de Milagros", "Spanish", "Drama"] [1, 2]
["El Tigre de Guanajuato", "Spanish", "Adventure"] [1, 2]
["El Vizconde de Montecristo", "Spanish", "Comedy"] [1, 2]
["El mejor", "Spanish", "Drama"] [1, 2]
["Esos de P\u00e9njamo", "Spanish", "Drama"] [1, 2]
["Estrella Sin Luz", "Spanish", "Drama"] [1, 2]
["Fallaste Coraz\u00f3n", "Spanish", "Drama"] [1, 2]
["Fashionably Late With Rachel Zoe", "Talk", "Fashion"] [1, 2]
["George Lopez", "Sitcom"] [65, 1]
["Hagit - Designer Jewelry", "Shopping", "Talk"] [1, 2]
["I Dream of Jeannie", "Sitcom"] [21, 1]
["Israel Tour June 2015", "Community"] [5, 1]
["Izrail' Plyus Predstavlyaet", "Community"] [1, 1]
["Jack Holt At The River", "Religious", "Community"] [1, 2]
["Jade Warrior", "Spanish", "Action", "Adventure", "Martial arts"] [1, 4]
["Jerry Springer", "Talk"] [109, 1]
["Jimmy Kimmel Live", "Talk", "Comedy"] [9, 2]
["Jonathan Last on The Dadly Virtues", "Special", "Talk"] [1, 2]
["Juan sin Miedo", "Spanish", "Drama"] [2, 2]
["Judo Budapest Grand Prix 2014 Highlights", "Special", "Sports non-event",
"Martial arts"] [1, 3]
["Just Shoot Me", "Sitcom"] [13, 1]
["Justice With Judge Jeanine", "Talk", "News"] [4, 2]
["La Maldici\u00f3n de la Momia Azteca", "Spanish", "Horror"] [2, 2]
["La Masacre de los P\u00e9rez", "Spanish", "Drama"] [1, 2]
["La Monja Alf\u00e9rez", "Spanish", "Drama"] [1, 2]
["La Oveja Negra", "Spanish", "Drama"] [1, 2]
["La Vida Dif\u00edcil de una Mujer F\u00e1cil", "Spanish", "Drama"] [1, 2]
["La visita que no toc\u00f3 el timbre", "Spanish", "Comedy"] [1, 2]
["Lamberto Quintero", "Spanish", "Drama"] [1, 2]
["Late Night Joy", "Talk"] [1, 1]
["Lo Azul del Cielo", "Spanish", "Drama", "Romance", "Suspense"] [1, 4]
["Lo Mejor de Caso Cerrado", "Law", "Reality", "Talk"] [1, 3]
["Lo Mejor de la Madre Ang\u00e9lica", "Talk", "Religious"] [15, 2]
["Los Campeones Justicieros", "Spanish", "Action"] [1, 2]
["Los Fern\u00e1ndez de Peralvillo", "Spanish", "Drama"] [1, 2]
["Los Hijos de Peralvillo", "Spanish", "Drama"] [1, 2]
["Los Maistros: El D\u00eda de la Santa Cruz", "Spanish", "Comedy"] [1, 2]
["M\u00e1s Vale P\u00e1jaro en Mano", "Spanish", "Comedy"] [1, 2]
["MediaBuzz", "News", "Talk", "Public affairs", "Politics"] [2, 4]
["Mejor Estar Solo", "Spanish", "Comedy"] [1, 2]
["Mojados", "Spanish", "Drama"] [1, 2]
["Mojoe", "Entertainment", "Talk", "Newsmagazine"] [1, 3]
["Music for Change: The Global Citizen", "Special", "Music", "Community"]
[1, 3]

["Operaci\u00f3n Jaque", "Spanish", "Drama"] [1, 2]
["Programa do J\u00f4", "Talk", "Interview"] [12, 2]
["Q & A", "News", "Talk", "Interview"] [1, 3]
["Q", "Talk", "Entertainment", "Variety"] [2, 3]
["Quadriga - The International Talk Show", "Talk", "Public affairs",
"Newsmagazine"] [1, 3]
["Rosario Tijeras", "Spanish", "Crime drama", "Romance"] [1, 3]
["Santo y Mantequilla N\u00e1poles", "Spanish", "Action"] [1, 2]
["Serpiente Azteca", "Spanish", "Drama"] [1, 2]
["Soy el Hijo del Tah\u00far", "Spanish", "Action", "Drama"] [1, 3]
["St. Joe Live Presents", "Community"] [1, 1]
["State of Mine: Jim Hunt Story", "Special", "Community"] [1, 2]
["The Daily Show With Jon Stewart", "Talk", "Interview", "Comedy"] [11, 3]
["The Dr. Oz Show", "Talk", "Health"] [94, 2]
["The Gossip Queens", "Talk", "Entertainment"] [1, 2]
["The Jamie Foxx Show", "Sitcom"] [17, 1]
["The Jeffersons", "Sitcom"] [9, 1]
["The Jim Gaffigan Show", "Sitcom"] [3, 1]
["The Josh Wolf Show", "Talk", "Comedy"] [3, 2]
["The King of Queens", "Sitcom"] [146, 1]
["The Late Late Show With James Corden", "Talk", "Comedy"] [52, 2]
["The Queen Latifah Show", "Talk", "Variety"] [48, 2]
["The Suite Life of Zack & Cody", "Children", "Sitcom"] [11, 2]
["The Tonight Show Starring Jimmy Fallon", "Talk", "Comedy"] [9, 2]
["Todo Lo Que T\u00fa Quieras", "Spanish", "Drama"] [1, 2]
["Town Square", "Community"] [1, 1]
["Un Balazo para Quintana", "Spanish", "Action"] [1, 2]
["Una Mujer Para los S\u00e1bados", "Spanish", "Drama"] [1, 2]
["Una Mujer Sin Amor", "Spanish", "Drama"] [1, 2]
["Viaje Redondo", "Spanish", "Drama"] [1, 2]
["WLJC Spring Telethon", "Special", "Community"] [1, 2]
["What Would Julieanna Do?", "Talk", "Cooking"] [1, 2]
["Wizards of Waverly Place", "Children", "Sitcom", "Fantasy"] [7, 3]
["\u00bfQui\u00e9n Paga la Cuenta?", "Spanish", "Comedy"] [1, 2]
["q", "Talk", "Entertainment", "Variety"] [8, 3]
["2014 LBJ Civil Rights Summit", "Community"] [2, 1]
["7 Cajas", "Spanish", "Action", "Suspense"] [1, 3]
["Adventures of Johnny Tao: Rock", "Action", "Adventure", "Martial arts"]
[1, 3]
["Al Rojo Vivo", "Talk", "Newsmagazine"] [4, 2]
["Alicia Menendez Tonight", "Talk", "Politics"] [1, 2]
["Amor y Frijoles", "Spanish", "Comedy-drama"] [2, 2]
["Antiques Roadshow: In Bismarck", "Collectibles", "Community"] [1, 2]
["Aqu\u00ed Nos Toc\u00f3 Vivir", "Community", "Travel"] [1, 2]
["Around the Corner With John McGivern", "Community"] [2, 1]
["Arquitectos de lo Imposible", "Community"] [1, 1]
["Art Basel Design District Magazine", "Community", "Public affairs"] [5, 2]
["Bajo el Mismo Techo", "Sitcom"] [1, 1]

```
["Big Morning Buzz Live", "Talk", "Entertainment", "News"]      [2, 3]
["Check Please! Arizona", "Community"]  [2, 1]
["Choque de Opiniones", "Talk", "News", "Debate"]       [3, 3]
["Cilantro y Perejil", "Spanish", "Romance-comedy"]     [1, 2]
["Cool Jobs", "Community", "Educational"]       [8, 2]
["Coruj\u00e3o do Esporte", "Sports non-event", "Talk"] [1, 2]
["Crazy Talk", "Comedy", "Talk"]        [30, 2]
["Crimenes De Lujuria", "Spanish", "Drama", "Suspense"] [1, 3]
["Di\u00e1logos en Confianza", "Talk"]  [1, 1]
["Dos Mojados En Apuros", "Spanish", "Comedy"]  [1, 2]
["Dulces Navajas", "Spanish", "Drama"]  [1, 2]
["Duro y Parejo en la Casita del Pecado", "Spanish", "Comedy"]  [1, 2]
["Ek The Raja Ek Thi Rani", "Community"]        [4, 1]
["El Baile de San Juan", "Spanish", "Historical drama"] [1, 2]
["El Cuerpazo del Delito", "Spanish", "Comedy-drama"]   [1, 2]
["El Efecto Tequila", "Spanish", "Comedy-drama"]        [1, 2]
["El Esqueleto de la Se\u00f1ora Morales", "Spanish", "Comedy-drama"]   [1, 2]
Removing temp directory
/tmp/hw1_mrA_337604821_326922390.root.20240710.125931.013212…
```

——————————MarkDown ——————————— Part A Q1.B

[13]:
```
%%file hw1_mrB_337604821_326922390.py

from mrjob.job import MRJob
from mrjob.step import MRStep
import re

def split_properley (line):
    item = line.split(',')
    date = item[-3]
    air_time = int(item[-2]) if item[-2].isnumeric() else 0

    if item[2] != item[-4]:
        genres = line.split('\"')[1].split(',') if item[0] != 'title' else␣
 ↪"Hello World"
    else:
      genres = [item[2]]
    title = item[0].strip()
    return title, genres, air_time, date

class MRWordFrequencyCount(MRJob):
    def steps(self):
        return [
            MRStep(mapper=self.mapper,
                   reducer=self.reducer),
            MRStep(reducer=self.reducer_count_genres),
            MRStep(reducer=self.reducer_max)
```

```
        ]

    def mapper(self, _, line):
        title, genres, air_time, date = split_properley(line)
        try:
            if 70000 <= int(air_time) < 90000 and re.search('[jqz]', title.
↪lower()):
                for genre in genres:
                    if genre.strip() in ['Sitcom', 'Talk', 'Politics',␣
↪'Spanish', 'Community', 'Martial arts']:
                        ls = [title]
                        ls.extend(genres)
                        yield (tuple(ls), date)
        except Exception as e:
            pass

    def reducer(self, key, dates):
        unique_dates = set(dates)
        yield key, len(unique_dates)

    def reducer_count_genres(self, key, values):
        total_dates = 0
        for dates_count in values:
            total_dates += dates_count
        yield None, (key, total_dates + len(key)-1)# yield key with the sum of␣
↪genres and total unique dates

    def reducer_max(self, _, values):
        yield max(values, key=lambda x: x[1])



if __name__ == '__main__':
    MRWordFrequencyCount.run()
```

```
Overwriting hw1_mrB_337604821_326922390.py
```

[14]:
```
!python hw1_mrB_337604821_326922390.py 420k_daily_prog_data.csv
```

```
No configs found; falling back on auto-configuration
No configs specified for inline runner
Creating temp directory
/tmp/hw1_mrB_337604821_326922390.root.20240710.130008.257936
Running step 1 of 3…
Running step 2 of 3…
Running step 3 of 3…
job output is in
```

```
/tmp/hw1_mrB_337604821_326922390.root.20240710.130008.257936/output
Streaming final output from
/tmp/hw1_mrB_337604821_326922390.root.20240710.130008.257936/output…
["The King of Queens", "Sitcom"]        147
Removing temp directory
/tmp/hw1_mrB_337604821_326922390.root.20240710.130008.257936…
```

——————————MarkDown ——————————

Part **B**

[ ]:
```
# Install PySpark on the Colab machine - code in   "    "
# Cell 1
!apt-get install openjdk-8-jdk-headless -qq > /dev/null
import os
os.environ["JAVA_HOME"] = "/usr/lib/jvm/java-8-openjdk-amd64"
!update-alternatives --set java /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java
!java -version
# Cell 2
!pip install --force-reinstall pyspark==3.4
!pip install findspark
```

[ ]:
```
# Install PySpark on the Colab machine - code in   "    "
# Cell 1
!apt-get install openjdk-8-jdk-headless -qq > /dev/null
import os
os.environ["JAVA_HOME"] = "/usr/lib/jvm/java-8-openjdk-amd64"
!update-alternatives --set java /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java
!java -version
# Cell 2
!pip install --force-reinstall pyspark==3.4
!pip install findspark
```

[32]:
```
import pyspark
from pyspark.sql import SparkSession
from pyspark.mllib.random import RandomRDDs
from pyspark.sql.types import*
from pyspark.sql.functions import to_date, dayofweek, date_format
import logging
import re
```

[4]:
```
# Before getting/creating the Session, we can try to modify parameters.
spark = SparkSession.builder.appName('Spark - HW01 Q2')\
    .getOrCreate()
sc = spark.sparkContext
# keep only important logs
spark.sparkContext.setLogLevel("ERROR")
```

[52]:
```
bl = ["Big", "the", "bang", "theory", "almanac", "met", "mother", "your",␣
↪"city", "anatomy", "game", "throne", "guy", "family", "friend", "senate",␣
↪"two"]

g = lambda genre, typ, pnts: pnts if (typ in genre and len(genre) == 1) else 0␣
↪# give pnts based on matching genre if it's the only genre in tuple
time_len = lambda time: time / 15
def white_list(ls):
  for word in bl:# check if given ls has a word in the black list, false if it␣
↪does otherwise True.
    for wrd in re.split('[,:]?\s', ls):
      if word.lower() == wrd.lower():
        return False
  return True
# score calculation of tuple
score_func = lambda title, genres, date, duration: (date + sum(g(genres, gg, p)␣
↪for gg, p in zip(["Action", "Documentary", "Sitcom"],[90,90,7])) +␣
↪time_len(duration)) if white_list(title) else 0
```

[61]:
```
def split_properly(item):
  date = 3 if item[-3] and item[-3] in [1,3,5, 7] else 0 #looking for odd date
  genres_tuple = tuple(item[2].split(',')) if item[2] else ()# find the␣
↪different genres
  title = item[0]
  duration = float(str(item[-1])) if re.match(r"^-?[0-9]+(\.[0-9]+)?$",␣
↪str(item[-1])) else 0 # check if format matches and convert to float
  return title, genres_tuple, date, duration
```

[62]:
```
df = spark.read.csv('420k_daily_prog_data.csv', header=True, inferSchema=True)
df = df.withColumn("air_date", dayofweek(to_date(df.air_date, "yyyyMMdd")))#␣
↪convert date to corresponding weekday value
```

[63]:
```
sol = df.rdd.map(split_properly) \
          .map(lambda row: ((row[0], row[1]), score_func(row[0], row[1],␣
↪row[2], row[3]))) \
          .reduceByKey(lambda x, y: x + y) \
          .sortBy(lambda x: x[1], ascending=False)
# rdd that applies split_properly on each row, then maps the title and genres␣
↪to a score on each tuple
# we then reduce to count duplicate keys to get total score and sort by to find␣
↪the top ones
```

[64]:
```
solution = sol.collect()[:20]
# run the action and take the top 20
```

[65]:
```python
for (title, genres), score in solution:
    genres_str = ', '.join(genres)
    print(f'{{"{title}", {genres_str}}} | {score}')
```

```
{"SIGN OFF", Special} | 22648.13333333333
{"Everybody Loves Raymond", Sitcom} | 14961.533333333326
{"Documentary", Documentary} | 13933.06666666666
{"Mike & Molly", Sitcom} | 13269.600000000002
{"Hot in Cleveland", Sitcom} | 11855.666666666668
{"Seinfeld", Sitcom} | 11084.133333333335
{"Community", Sitcom} | 9085.533333333333
{"ABC World News Now", News} | 8337.999999999989
{"Strange Inheritance", Documentary} | 8113.0
{"Un Mundo Maravilloso", Documentary} | 8078.0
{"Drug Wars", Documentary} | 7890.0
{"Weather Radar", Weather} | 6987.0
{"True Life", Documentary} | 5923.466666666664
{"Larry King Special Report", Consumer} | 5796.4000000000015
{"Classic Arts Showcase", Art} | 5731.666666666666
{"Programa Pagado", Shopping} | 5447.799999999999
{"Live Cameras and Forecast Maps", Weather} | 5391.733333333235
{"Forensic Files", Reality, Crime} | 5277.0
{"Local Weather", Weather} | 5257.0
{"Dateline on ID", Documentary} | 5254.0
```