CODING DOJO

الأكاديمية السعودية الرقمية
SAUDI DIGITAL ACADEMY

MapReduce Programming Model

Report

Team: Bigger than data(BTD)

Abdullah Alhuwaishel

Amjad Almusallam Afnan Alzahrani

Mahmoud Alhassan Jumana Almussa

Dataset:

The **Hotel Reviews** dataset contains two columns and 20491 row.

Columns are:
Review: description of the experiment.
Rating: integers from 1 to 5, where 5 is the best and 1 is the worst.

Problem statement:

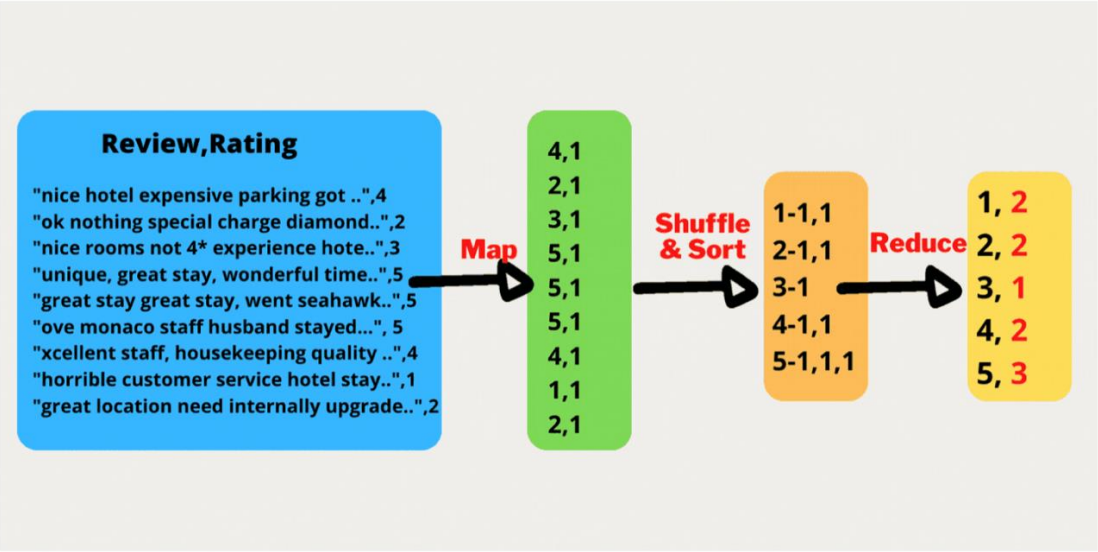
A hotel intends to do maintenance based on the rating; If the most of customers rated less than three stars, the maintenance will be done by the end of the year; otherwise, it will be done later.
So, we used MapReduce to count the number of reviews for each rating(1,2,3,4,5) in the dataset.

Implementation Steps:

Step 1: Transform raw data into key/value pairs in parallel.
The mapper will get the data file and make the Rating the key and the values will be the reviews. We will add number 1 for reviews.

Step 2: Shuffle and short by the MapReduce model.
The process of transferring mappers' intermediate output to the reducer is known as shuffling. It will collect all the reviews(number 1s) together with the individual key and it will sort them. it will get sorted by key.

Step3: Process the data using Reduce.
Reduce will count each value(number 1) for each key.

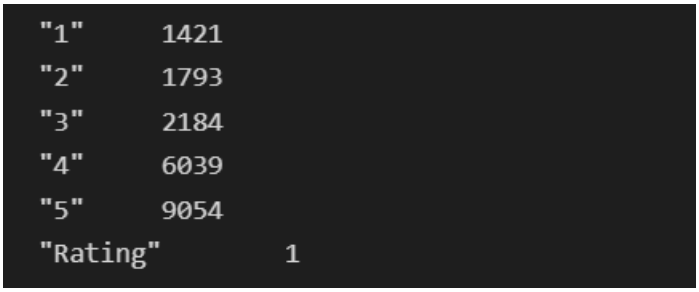


Code:

```
%%file Reviews.py
#Magic function that saves the code cell as a file

from mrjob.job import MRJob #import the mrjob library
from mrjob.step import MRStep #import the MRStep library
import csv #import the csv library

#store columns names
columns = ["Review","Rating"]

class NoRatings(MRJob): # Create a class named NoRatings (number of Ratings )
    def steps(self): # Create method named steps and pass  the mapper and the reducer for
MRStep
        return[
            MRStep(mapper=self.mapping,
                   reducer=self.reducing)
        ]

#Create Mapper function
    def mapping(self, _, line): #(_, line)ignore the key and take each line of the document
as the value.
       reader = csv.reader([line]) #reader from csv file line by line
       for row in reader: # for loop to read rows
           zipped=zip(columns,row) # creates a tuple  ,,, the columns is the key , row is the
values
           diction=dict(zipped) # creates dic
           ratings=diction['Rating'] #store Rating

           yield ratings, 1  #outputing as key value pairs

#Reducer function
    def reducing(self, key, values):  #sum all the values per key
       yield key, sum(values)

if __name__ == "__main__":
    NoRatings.run()
```

```
# run the code as a terminal command
!python Reviews.py Hotel_Reviews.csv
```

Result:



Since most of the ratings are in the 5-star and the 4-star, so the hotel maintenance plans can be postponed until the next year.