SCALABLE ANALYTICS – HW 2 REPORT

Chris Cirelli
04.08.2018

QUESTION 1

1) Explanation of the code design

Create Key-Value Pair RDD

Documentation:
i.) Create a top-level function called 'Transform' to
a.) Limit the return value to index 5 (text of review) and index 6 (rating)
b.) Split the review text on spaces and thereby tokenizing it.
c.) Filter out any punctuation from the tokenized review text
d.) take the length of the list of tokenized text
ii.) Return: (Review, length text)

Code:

```
def CreateKeyValuePair(Lists):
        punct = string.punctuation
        return Lists[6], len(list(filter(lambda x: x not in punct, Lists[5].split(' '))))
```

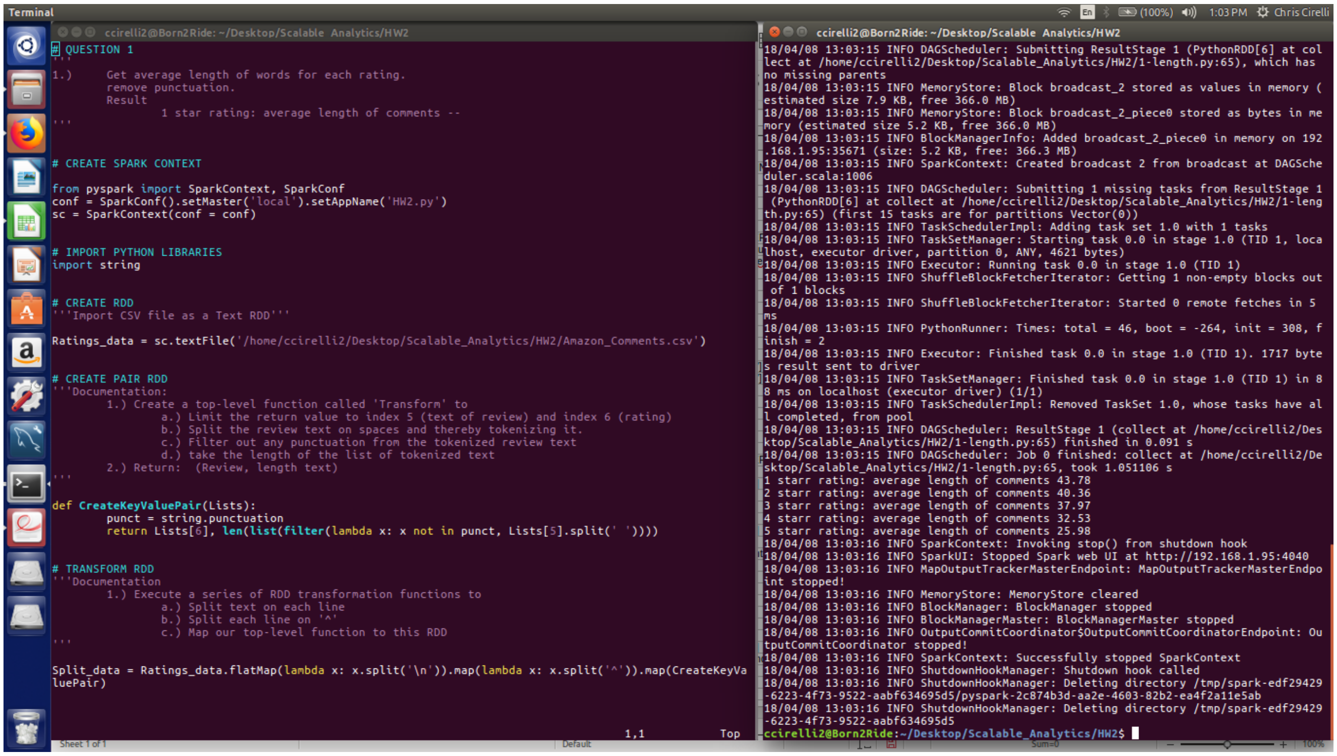
Transform RDD

Documentation:
i.) Execute a series of RDD transformation functions to
a.) Split text on each line
b.) Split each line on '^'
c.) Map our top-level function to this RDD

Code:

```
Split_data = Ratings_data.flatMap(lambda x: x.split('\n')).map(lambda x:
x.split('^')).map(CreateKeyValuePair)
```

2) Experimental Results



```
# QUESTION 1
'''
1.)     Get average length of words for each rating.
        remove punctuation.
        Result
                1 star rating: average length of comments --
'''


# CREATE SPARK CONTEXT

from pyspark import SparkContext, SparkConf
conf = SparkConf().setMaster('local').setAppName('HW2.py')
sc = SparkContext(conf = conf)


# IMPORT PYTHON LIBRARIES
import string


# CREATE RDD
'''Import CSV file as a Text RDD'''

Ratings_data = sc.textFile('/home/ccirelli2/Desktop/Scalable_Analytics/HW2/Amazon_Comments.csv')


# CREATE PAIR RDD
'''Documentation:
        1.) Create a top-level function called 'Transform' to
                a.) Limit the return value to index 5 (text of review) and index 6 (rating)
                b.) Split the review text on spaces and thereby tokenizing it.
                c.) Filter out any punctuation from the tokenized review text
                d.) take the length of the list of tokenized text
        2.) Return:  (Review, length text)
'''

def CreateKeyValuePair(Lists):
        punct = string.punctuation
        return Lists[6], len(list(filter(lambda x: x not in punct, Lists[5].split(' '))))


# TRANSFORM RDD
'''Documentation
        1.) Execute a series of RDD transformation functions to
                a.) Split text on each line
                b.) Split each line on '^'
                c.) Map our top-level function to this RDD
'''

Split_data = Ratings_data.flatMap(lambda x: x.split('\n')).map(lambda x: x.split('^')).map(CreateKeyVa
luePair)



                                                                                  1,1           Top
```

```
18/04/08 13:03:15 INFO DAGScheduler: Submitting ResultStage 1 (PythonRDD[6] at col
lect at /home/ccirelli2/Desktop/Scalable_Analytics/HW2/1-length.py:65), which has
no missing parents
18/04/08 13:03:15 INFO MemoryStore: Block broadcast_2 stored as values in memory (
estimated size 7.9 KB, free 366.0 MB)
18/04/08 13:03:15 INFO MemoryStore: Block broadcast_2_piece0 stored as bytes in me
mory (estimated size 5.2 KB, free 366.0 MB)
18/04/08 13:03:15 INFO BlockManagerInfo: Added broadcast_2_piece0 in memory on 192
.168.1.95:35671 (size: 5.2 KB, free: 366.3 MB)
18/04/08 13:03:15 INFO SparkContext: Created broadcast 2 from broadcast at DAGSche
duler.scala:1006
18/04/08 13:03:15 INFO DAGScheduler: Submitting 1 missing tasks from ResultStage 1
 (PythonRDD[6] at collect at /home/ccirelli2/Desktop/Scalable_Analytics/HW2/1-leng
th.py:65) (first 15 tasks are for partitions Vector(0))
18/04/08 13:03:15 INFO TaskSchedulerImpl: Adding task set 1.0 with 1 tasks
18/04/08 13:03:15 INFO TaskSetManager: Starting task 0.0 in stage 1.0 (TID 1, loca
lhost, executor driver, partition 0, ANY, 4621 bytes)
18/04/08 13:03:15 INFO Executor: Running task 0.0 in stage 1.0 (TID 1)
18/04/08 13:03:15 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out
 of 1 blocks
18/04/08 13:03:15 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 5
ms
18/04/08 13:03:15 INFO PythonRunner: Times: total = 46, boot = -264, init = 308, f
inish = 2
18/04/08 13:03:15 INFO Executor: Finished task 0.0 in stage 1.0 (TID 1). 1717 byte
s result sent to driver
18/04/08 13:03:15 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 1) in 8
8 ms on localhost (executor driver) (1/1)
18/04/08 13:03:15 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have al
l completed, from pool
18/04/08 13:03:15 INFO DAGScheduler: ResultStage 1 (collect at /home/ccirelli2/Des
ktop/Scalable_Analytics/HW2/1-length.py:65) finished in 0.091 s
18/04/08 13:03:15 INFO DAGScheduler: Job 0 finished: collect at /home/ccirelli2/De
sktop/Scalable_Analytics/HW2/1-length.py:65, took 1.051106 s
1 starr rating: average length of comments 43.78
2 starr rating: average length of comments 40.36
3 starr rating: average length of comments 37.97
4 starr rating: average length of comments 32.53
5 starr rating: average length of comments 25.98
18/04/08 13:03:16 INFO SparkContext: Invoking stop() from shutdown hook
18/04/08 13:03:16 INFO SparkUI: Stopped Spark web UI at http://192.168.1.95:4040
18/04/08 13:03:16 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpo
int stopped!
18/04/08 13:03:16 INFO MemoryStore: MemoryStore cleared
18/04/08 13:03:16 INFO BlockManager: BlockManager stopped
18/04/08 13:03:16 INFO BlockManagerMaster: BlockManagerMaster stopped
18/04/08 13:03:16 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: Ou
tputCommitCoordinator stopped!
18/04/08 13:03:16 INFO SparkContext: Successfully stopped SparkContext
18/04/08 13:03:16 INFO ShutdownHookManager: Shutdown hook called
18/04/08 13:03:16 INFO ShutdownHookManager: Deleting directory /tmp/spark-edf29429
-6223-4f73-9522-aabf634695d5/pyspark-2c874b3d-aa2e-4603-82b2-ea4f2a11e5ab
18/04/08 13:03:16 INFO ShutdownHookManager: Deleting directory /tmp/spark-edf29429
-6223-4f73-9522-aabf634695d5
ccirelli2@Born2Ride:~/Desktop/Scalable_Analytics/HW2$ 
```

2.2) Explain your results.

The results indicate that the higher the rating the fewer the average length of the comments.

QUESTION 2

CREATE KEY VALUE PAIR

Documentation

1.) Text_RDD.map(lambda x: x.split('^')): Split the text on '^'

2.) map(lambda x: (x[-1], x[5].lower())): Return the Rating + Text, convert text to lowercase

3.) map(lambda x: (x[0], x[1].split(' '))):Return Rating & Text tokenized

4.) flatMapValues(lambda x: (x[0], x[1])): Ensure KeyValue pair structure

5.) filter(lambda x: x[1] not in string.punctuation): Strip punctuation from text

6.) filter(lambda x: x[1] != 'i').persist(): Strip token 'i' that shows up in all ratings.

7.) persist(): Persist the result of this RDD

Code:

```
Create_KeyValuePair = Text_RDD.map(lambda x: x.split('^')).map(lambda x: (x[-1],
x[5].lower())).map(lambda x: (x[0], x[1].split(' '))).flatMapValues(lambda x: (x[0],
x[1])).filter(lambda x: x[1] not in string.punctuation).filter(lambda x: x[1] != 'i').persist()
```

GET TOP WORDS BY STAR

Documentation

1.) Create_KeyValuePair.filter(lambda x: '1.00' in x): Limit to a single rating

2.) map(lambda x: (x[1], 1)): Return a key value pair of (Token, 1)

3.) reduceByKey(lambda x,y: x+y): Reduce key value pair (add like keys together)

4.) map(lambda x: (x[1], x[0])): Change the shape of the keyvalue pair to (value, key)

5.) sortByKey(ascending = False): Sort from highest to lowest on the value

6.) map(lambda x: x[1]): Reshape to return just the key.

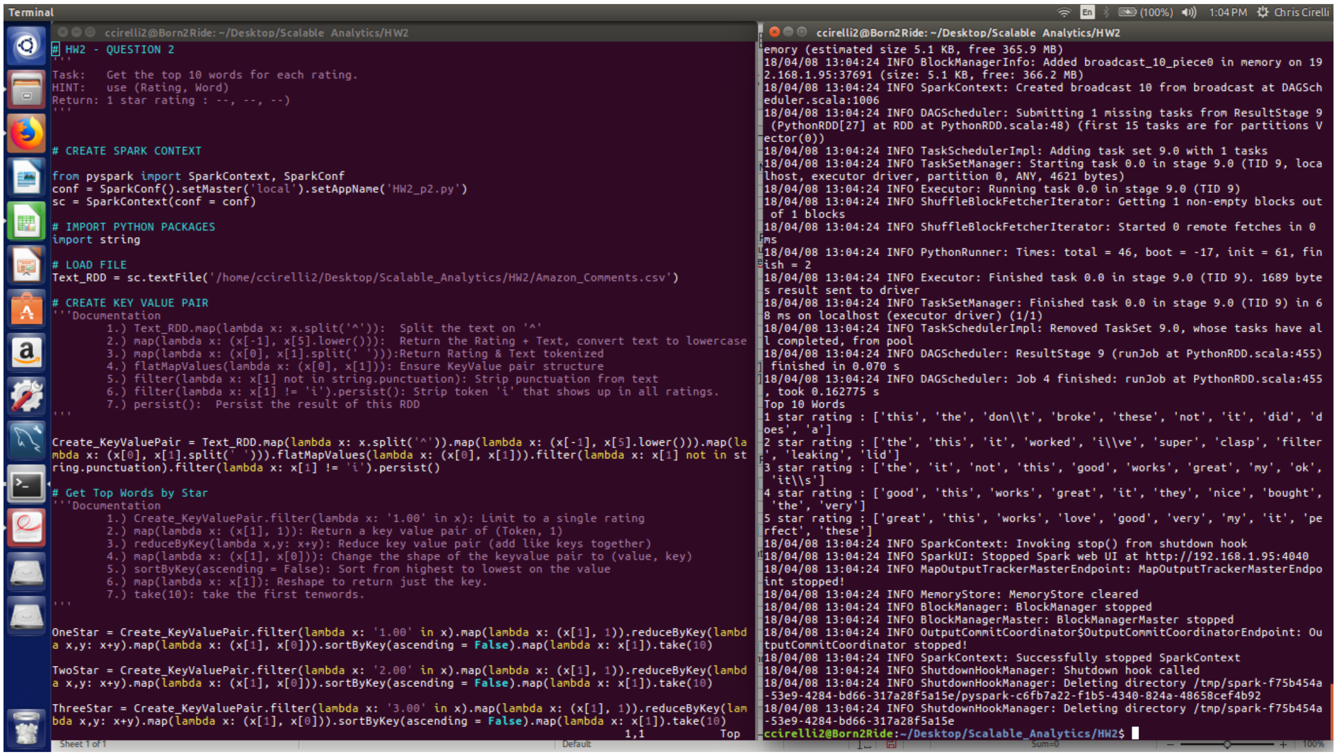
7.) take(10): take the first tenwords.

Code:

```
OneStar = Create_KeyValuePair.filter(lambda x: '1.00' in x).map(lambda x: (x[1],
1)).reduceByKey(lambda x,y: x+y).map(lambda x: (x[1], x[0])).sortByKey(ascending =
False).map(lambda x: x[1]).take(10)

TwoStar = Create_KeyValuePair.filter(lambda x: '2.00' in x).map(lambda x: (x[1],
1)).reduceByKey(lambda x,y: x+y).map(lambda x: (x[1], x[0])).sortByKey(ascending =
False).map(lambda x: x[1]).take(10)

ThreeStar = Create_KeyValuePair.filter(lambda x: '3.00' in x).map(lambda x: (x[1],
1)).reduceByKey(lambda x,y: x+y).map(lambda x: (x[1], x[0])).sortByKey(ascending =
False).map(lambda x: x[1]).take(10)

FourStar = Create_KeyValuePair.filter(lambda x: '4.00' in x).map(lambda x: (x[1],
1)).reduceByKey(lambda x,y: x+y).map(lambda x: (x[1], x[0])).sortByKey(ascending =
False).map(lambda x: x[1]).take(10)

FiveStar = Create_KeyValuePair.filter(lambda x: '5.00' in x).map(lambda x: (x[1],
1)).reduceByKey(lambda x,y: x+y).map(lambda x: (x[1], x[0])).sortByKey(ascending =
False).map(lambda x: x[1]).take(10)

print('Top 10 Words')
print('1 star rating :', OneStar)
print('2 star rating :', TwoStar)
print('3 star rating :', ThreeStar)
print('4 star rating :', FourStar)
print('5 star rating :', FiveStar)
```

Results:

- ‘Great’ and ‘Good’ are the two words most frequency used to describe the 4 and 5 star ratings.


Terminal
ccirelli2@Born2Ride: ~/Desktop/Scalable Analytics/HW2
En
(100%)
1:04 PM
Chris Cirelli
ccirelli2@Born2Ride: ~/Desktop/Scalable Analytics/HW2
Sheet 1 of 1
Default
Sum=0
100%

```
# HW2 - QUESTION 2
'''
Task:   Get the top 10 words for each rating.
HINT:   use (Rating, Word)
Return: 1 star rating : --, --, --)
'''


# CREATE SPARK CONTEXT

from pyspark import SparkContext, SparkConf
conf = SparkConf().setMaster('local').setAppName('HW2_p2.py')
sc = SparkContext(conf = conf)

# IMPORT PYTHON PACKAGES
import string

# LOAD FILE
Text_RDD = sc.textFile('/home/ccirelli2/Desktop/Scalable_Analytics/HW2/Amazon_Comments.csv')

# CREATE KEY VALUE PAIR
'''Documentation
        1.) Text_RDD.map(lambda x: x.split('^')):  Split the text on '^'
        2.) map(lambda x: (x[-1], x[5].lower())):  Return the Rating + Text, convert text to lowercase
        3.) map(lambda x: (x[0], x[1].split(' '))):Return Rating & Text tokenized
        4.) flatMapValues(lambda x: (x[0], x[1])): Ensure KeyValue pair structure
        5.) filter(lambda x: x[1] not in string.punctuation): Strip punctuation from text
        6.) filter(lambda x: x[1] != 'i').persist(): Strip token 'i' that shows up in all ratings.
        7.) persist():  Persist the result of this RDD
'''

Create_KeyValuePair = Text_RDD.map(lambda x: x.split('^')).map(lambda x: (x[-1], x[5].lower())).map(la
mbda x: (x[0], x[1].split(' '))).flatMapValues(lambda x: (x[0], x[1])).filter(lambda x: x[1] not in st
ring.punctuation).filter(lambda x: x[1] != 'i').persist()

# Get Top Words by Star
'''Documentation
        1.) Create_KeyValuePair.filter(lambda x: '1.00' in x): Limit to a single rating
        2.) map(lambda x: (x[1], 1)): Return a key value pair of (Token, 1)
        3.) reduceByKey(lambda x,y: x+y): Reduce key value pair (add like keys together)
        4.) map(lambda x: (x[1], x[0])): Change the shape of the keyvalue pair to (value, key)
        5.) sortByKey(ascending = False): Sort from highest to lowest on the value
        6.) map(lambda x: x[1]): Reshape to return just the key.
        7.) take(10): take the first tenwords.
'''

OneStar = Create_KeyValuePair.filter(lambda x: '1.00' in x).map(lambda x: (x[1], 1)).reduceByKey(lambd
a x,y: x+y).map(lambda x: (x[1], x[0])).sortByKey(ascending = False).map(lambda x: x[1]).take(10)

TwoStar = Create_KeyValuePair.filter(lambda x: '2.00' in x).map(lambda x: (x[1], 1)).reduceByKey(lambd
a x,y: x+y).map(lambda x: (x[1], x[0])).sortByKey(ascending = False).map(lambda x: x[1]).take(10)

ThreeStar = Create_KeyValuePair.filter(lambda x: '3.00' in x).map(lambda x: (x[1], 1)).reduceByKey(lam
bda x,y: x+y).map(lambda x: (x[1], x[0])).sortByKey(ascending = False).map(lambda x: x[1]).take(10)
                                                                              1,1           Top
```

```
emory (estimated size 5.1 KB, free 365.9 MB)
18/04/08 13:04:24 INFO BlockManagerInfo: Added broadcast_10_piece0 in memory on 19
2.168.1.95:37691 (size: 5.1 KB, free: 366.2 MB)
18/04/08 13:04:24 INFO SparkContext: Created broadcast 10 from broadcast at DAGSch
eduler.scala:1006
18/04/08 13:04:24 INFO DAGScheduler: Submitting 1 missing tasks from ResultStage 9
 (PythonRDD[27] at RDD at PythonRDD.scala:48) (first 15 tasks are for partitions V
ector(0))
18/04/08 13:04:24 INFO TaskSchedulerImpl: Adding task set 9.0 with 1 tasks
18/04/08 13:04:24 INFO TaskSetManager: Starting task 0.0 in stage 9.0 (TID 9, loca
lhost, executor driver, partition 0, ANY, 4621 bytes)
18/04/08 13:04:24 INFO Executor: Running task 0.0 in stage 9.0 (TID 9)
18/04/08 13:04:24 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out
 of 1 blocks
18/04/08 13:04:24 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 0
ms
18/04/08 13:04:24 INFO PythonRunner: Times: total = 46, boot = -17, init = 61, fin
ish = 2
18/04/08 13:04:24 INFO Executor: Finished task 0.0 in stage 9.0 (TID 9). 1689 byte
s result sent to driver
18/04/08 13:04:24 INFO TaskSetManager: Finished task 0.0 in stage 9.0 (TID 9) in 6
8 ms on localhost (executor driver) (1/1)
18/04/08 13:04:24 INFO TaskSchedulerImpl: Removed TaskSet 9.0, whose tasks have al
l completed, from pool
18/04/08 13:04:24 INFO DAGScheduler: ResultStage 9 (runJob at PythonRDD.scala:455)
 finished in 0.070 s
18/04/08 13:04:24 INFO DAGScheduler: Job 4 finished: runJob at PythonRDD.scala:455
, took 0.162775 s
Top 10 Words
1 star rating : ['this', 'the', 'don\\t', 'broke', 'these', 'not', 'it', 'did', 'd
oes', 'a']
2 star rating : ['the', 'this', 'it', 'worked', 'i\\ve', 'super', 'clasp', 'filter
', 'leaking', 'lid']
3 star rating : ['the', 'it', 'not', 'this', 'good', 'works', 'great', 'my', 'ok',
 'it\\s']
4 star rating : ['good', 'this', 'works', 'great', 'it', 'they', 'nice', 'bought',
 'the', 'very']
5 star rating : ['great', 'this', 'works', 'love', 'good', 'very', 'my', 'it', 'pe
rfect', 'these']
18/04/08 13:04:24 INFO SparkContext: Invoking stop() from shutdown hook
18/04/08 13:04:24 INFO SparkUI: Stopped Spark web UI at http://192.168.1.95:4040
18/04/08 13:04:24 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpo
int stopped!
18/04/08 13:04:24 INFO MemoryStore: MemoryStore cleared
18/04/08 13:04:24 INFO BlockManager: BlockManager stopped
18/04/08 13:04:24 INFO BlockManagerMaster: BlockManagerMaster stopped
18/04/08 13:04:24 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: Ou
tputCommitCoordinator stopped!
18/04/08 13:04:24 INFO SparkContext: Successfully stopped SparkContext
18/04/08 13:04:24 INFO ShutdownHookManager: Shutdown hook called
18/04/08 13:04:24 INFO ShutdownHookManager: Deleting directory /tmp/spark-f75b454a
-53e9-4284-bd66-317a28f5a15e/pyspark-c6fb7a22-f1b5-4340-824a-48658cef4b92
18/04/08 13:04:24 INFO ShutdownHookManager: Deleting directory /tmp/spark-f75b454a
-53e9-4284-bd66-317a28f5a15e
ccirelli2@Born2Ride:~/Desktop/Scalable_Analytics/HW2$
```