

Problem 1: Warm-up exercises

1. Screenshots of exercises.

Initialising the cluster("example-cluster") using command line:

```
shivamojha@Shivams-MacBook-Air:~$ gcloud dataproc clusters create example-cluster --region=us-east1
Waiting on operation [projects/big-data-6893-326522/regions/us-east1/operations/bbc80142-4484-35b7-9402-37abbb8ee7ee].
Waiting for cluster creation operation...
WARNING: No image specified. Using the default image version. It is recommended to select a specific image version in production, as the default image version may change at any time.
Waiting for cluster creation operation...done.
Created [https://dataproc.googleapis.com/v1/projects/big-data-6893-326522/regions/us-east1/clusters/example-cluster] Cluster placed in zone [us-east1-c].
```

Example-cluster created in GUI:

Clusters								
		+ CREATE CLUSTER	REFRESH	START	STOP	DELETE	REGIONS ▾	+ 5 RECOMMENDED ALERTS
Filter Search clusters, press Enter								
<input type="checkbox"/>	Name ↑	Status	Region	Zone	Total worker nodes	Scheduled deletion	Cloud Storage staging bucket	Created
<input type="checkbox"/>	example-cluster	Running	us-east1	us-east1-c	2	Off	dataproc-staging-us-east1-663482674546-zehxpelw	Sep 21, 2021, 1:54:49 PM

(a) Pi Calculation:

Successful Job Execution for Pi Calculation on CLI:

```
shivamojha@Shivams-MacBook-Air:~$ gcloud dataproc jobs submit spark --cluster example-cluster \
>   --region=us-east1 \
>   --class org.apache.spark.examples.SparkPi \
>   --jar https://spark-packages.s3.amazonaws.com/jars/spark-examples.jar -- --numPartitions 1000
Job [5296776c4ed74978ad50ff65262c2b74] submitted.

21/09/21 18:40:17 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found. Using default resource-types.xml.
21/09/21 18:40:17 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found. Using default resource-types.xml.
21/09/21 18:40:17 INFO org.apache.hadoop.yarn.client.AMRProxy: Connecting to Application History server at example-cluster-m/10.142.0.8:10200
21/09/21 18:40:17 INFO org.apache.hadoop.yarn.conf.Configuration: resource-types.xml not found. Using default resource-types.xml.
21/09/21 18:40:17 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_1532246946872_0001
21/09/21 18:40:17 INFO org.apache.hadoop.yarn.client.AMRProxy: Connecting to ResourceManager at example-cluster-m/10.142.0.8:8088
21/09/21 18:40:17 INFO org.apache.hadoop.yarn.client.AMRProxy: ResourceManager registered gcs://con-pgce-cloud-hadoop-gcs.google.cloud.hadoop.yarn/amrProxyImpl: ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state
21/09/21 18:40:17 INFO org.apache.hadoop.yarn.client.AMRProxy: ResourceManager registered gcs://con-pgce-cloud-hadoop-gcs.google.cloud.hadoop.yarn/amrProxyImpl: ignoring exception of type GoogleJsonResponseException; verified object already exists with desired state
21/09/21 18:40:17 INFO org.apache.hadoop.yarn.client.AMRProxy: Stopped SparkId7B2be(HTTP/1.1, (http://1.1)){0.0.0.0:8088}
Job [5296776c4ed74978ad50ff65262c2b74] finished successfully.

Job [5296776c4ed74978ad50ff65262c2b74] is roughly 1.147171331473232.

21/09/21 18:40:17 INFO org.apache.hadoop.yarn.server.AbstractConnector: Stopped SparkId7B2be(HTTP/1.1, (http://1.1)){0.0.0.0:8088}
done: true
driverControlFileUri: gs://dataproc-staging-us-east1-663482674546-zehxpelw/google-cloud-dataproc-metainfo/19835f9c-f1c6-48a1-a4d4-1ccc57d29f6/jobs/5296776c4ed74978ad50ff65262c2b74
driverOutputResourceUri: gs://dataproc-staging-us-east1-663482674546-zehxpelw/google-cloud-dataproc-metainfo/19835f9c-f1c6-48a1-a4d4-1ccc57d29f6/jobs/5296776c4ed74978ad50ff65262c2b74
driverYarnRootURI: gs://dataproc-staging-us-east1-663482674546-zehxpelw/google-cloud-dataproc-metainfo/19835f9c-f1c6-48a1-a4d4-1ccc57d29f6/jobs/5296776c4ed74978ad50ff65262c2b74
placement:
  - placementType: example-cluster
    clusterUuid: 19835f9c-f1c6-48a1-a4d4-1ccc57d29f6
    referenced:
      - clusterUuid: 19835f9c-f1c6-48a1-a4d4-1ccc57d29f6
        projectID: big-data-6893-326522
    sparkJob:
      - id: 1
        name: "Pi"
        progress: 1000
        jarFileUri: gs://spark/examples/jars/spark-examples.jar
        mainClass: org.apache.spark.examples.SparkPi
        state: DONE
        stateStartTime: "2021-09-21T18:01:19.689962Z"
        stateEndTime: "2021-09-21T18:01:51.787654Z"
      - id: 2
        name: "Pi"
        progress: 1000
        jarFileUri: gs://spark/examples/jars/spark-examples.jar
        mainClass: org.apache.spark.examples.SparkPi
        state: PENDING
        stateStartTime: "2021-09-21T18:00:51.822642Z"
        stateEndTime: "2021-09-21T18:00:51.822642Z"
      - id: 3
        name: "Pi"
        progress: 1000
        jarFileUri: gs://spark/examples/jars/spark-examples.jar
        mainClass: org.apache.spark.examples.SparkPi
        state: RUNNING
        stateStartTime: "2021-09-21T18:00:52.888156Z"
        stateEndTime: "2021-09-21T18:00:52.888156Z"
      - id: 4
        name: "Pi"
        progress: 1000
        jarFileUri: gs://spark/examples/jars/spark-examples.jar
        mainClass: org.apache.spark.examples.SparkPi
        state: FINISHED
        stateStartTime: "2021-09-21T18:00:52.888156Z"
        stateEndTime: "2021-09-21T18:00:52.888156Z"
        url: "http://example-cluster-m:8088/rmws/application_1632246946872_0001"
```

Successful Job Execution for Pi Calculation on GUI:

Cluster details

SUBMIT JOB REFRESH START STOP DELETE VIEW LOGS

No image specified. Using the default image version. It is recommended to select a specific image version in production, as the default image version may change at any time.

Name	example-cluster
Cluster UUID	1983539c-fc6-48a1-a4d4-1cccc57d29f6
Type	Dataproc Cluster
Status	Running

MONITORING JOBS VM INSTANCES CONFIGURATION WEB INTERFACES

Filter Filter jobs

Job ID	Status	Region	Type	Start time	Elapsed time	Labels
5296776c4ed74970ad5d0ff652dc2b74	Succeeded	us-east1	Spark	Sep 21, 2021, 20:05:11 PM	28 sec	None

Job details

CLOSE DELETE STOP REFRESH

Job ID	5296776c4ed74970ad5d0ff652dc2b74
Job UUID	a6e7d784-ac2f-3942-be2e-07b06c329428
Type	Dataproc Job
Status	Succeeded

MONITORING CONFIGURATION

The charts below represent the metrics from the cluster this job ran on, scoped to the time that this job was running. It is possible for more than one job to run on a cluster at a time, so these metrics may not reflect this job's resource usage accurately. Metrics for a job may lag behind the job run by several minutes.

RESET ZOOM 1 hour 6 hours 12 hours 1 day 2 days 4 days 7 days 14 days 30 days 1:58 PM - 2:03 PM ▾

YARN memory

YARN pending memory

Output LINE WRAP OFF

```
21/09/21 18:00:56 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@1d7f02abe([HTTP/1.1, (http://1.1)])(0.0.0.0:44411)
21/09/21 18:00:57 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to ResourceManager at example-cluster-m@19.142.0.8:8083
21/09/21 18:00:57 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at example-cluster-m@19.142.0.8:10208
21/09/21 18:00:57 INFO org.apache.hadoop.hadoop.conf.Configuration: resource-types.xml not found
21/09/21 18:00:57 INFO org.apache.hadoop.yarn.util.YarnClientImpl: Unable to find 'resource-type.xml'
21/09/21 18:00:57 INFO org.apache.hadoop.yarn.util.YarnClientImpl: Submitted application application_162246946872_8801
21/09/21 18:00:58 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_162246946872_8800
21/09/21 18:00:59 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to ResourceManager at example-cluster-m@19.142.0.8:8080
21/09/21 18:01:00 INFO google.cloud.hadoop.repackaged.gcs.google.cloud.hadoop.gcs.GoogleCloudStorageImpl: Ignoring exception of type GoogleJsonResponseException
Pi is roughly 3.141572314717323
21/09/21 18:01:17 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped ServerConnector@1d7f02abe([HTTP/1.1, (http://1.1)])(0.0.0.0:44411)
```

Output is complete

(b) Word-Count:

Successful Job Execution for word-count on CLI:

Successful Job Execution for word-count on GUI:

The screenshot shows the Google Cloud DataProc interface. On the left, there's a sidebar with 'Clusters' selected. In the main area, a cluster named 'word-count-cluster' is shown with its UUID and status as 'Running'. Below it, a job with ID '1063963923834abda06cb0505146cc' has a status of 'Succeeded'. The 'Job details' tab is active, displaying monitoring metrics like YARN memory usage over time.

Word Count Result in CLI:

```
(big-data) shivamojha@Shivams-MacBook-Air:~/Files/Columbia/Courses/Fall 21/Big Data Analytics/Hw0/Code$ gsutil cat gs://${BUCKET_NAME}/output/*
("what", 1)
("in", 1)
("name?", 1)
("that", 1)
("the", 1)
("call", 1)
("rose", 1)
("other", 1)
("one", 1)
("would", 1)
("smell", 1)
("is", 1)
("sweet.", 1)
("a", 2)
("which", 1)
("by", 1)
("any", 1)
```

Delete example-cluster after executing the jobs:

```
shivamojha@Shivams-MacBook-Air:~$ gcloud dataproc clusters delete example-cluster \
-> --region=us-east1
The cluster 'example-cluster' and all attached disks will be deleted.

Do you want to continue (Y/n)? Y
Waiting on operation [projects/big-data-6893-326522/regions/us-east1/operations/2775f37a-9adc-3f80-a35c-56d22b96e739].
Waiting for cluster deletion operation...done.
Deleted [https://dataproc.googleapis.com/v1/projects/big-data-6893-326522/regions/us-east1/clusters/example-cluster].
```

2. Spark transformations and actions:

- First, the randomly generated data is parallelized to create a RDD. Then the transformation "filter" is used to filter out the points inside the circle. Action "count" is used to count these points. In the end, we can simulate the the "Pi" value by performing the necessary mathematical calculation.
- A RDD is created initially using the "textFile" function. Then three transformations are applied: "FlatMap" is used to split word from lines, "map" is used to generate key-value pairs, and "reduceByKey" is used to count the frequency of each word. The action "saveAsTextFile" is used afterwards to save the "word-count" result as a text file.

Problem 2: NYC Bike expert

1. How many stations with longitude between -73.94 and -74.04?

The screenshot shows a query editor window titled "HW0_NY...". The query is:1 #(1) How many stations with longitude between -73.94 and -74.04?
2 select count(*) from `big-data-6893-326522.dataset1.bike_data`
3 where longitude between -74.04 and -73.94;Below the query, it says "Query complete (0 sec elapsed, cached)". The results tab is selected, showing a single row:

Row	f0_
1	698

2. What's the total number of bikes available in region_id 71?

The screenshot shows a query editor window titled "HW0_NY...". The query is:1 #(2) What's the total number of bikes available in region_id 71?
2 select sum(num_bikes_available) from `big-data-6893-326522.dataset1.bike_data`
3 where region_id=71;Below the query, it says "Query complete (0.2 sec elapsed, 13.2 KB processed)". The results tab is selected, showing a single row:

Row	f0_
1	11885

3. What's the largest capacity for a station?

The screenshot shows a query editor window titled "HW0_NY...". The query is:1 #(3) What's the largest capacity for a station?
2 select max(capacity) from `big-data-6893-326522.dataset1.bike_data`;Below the query, it says "Query complete (0.1 sec elapsed, 6.6 KB processed)". The results tab is selected, showing a single row:

Row	f0_
1	79

List all the station_id of the stations that have the largest capacity?

The screenshot shows a Jupyter Notebook interface with a query results table. The table has two columns: Row and station_id. The data is as follows:

Row	station_id
1	445
2	422
3	501

Problem 3: Understanding William Shakespeare

- Without any text pre-processing, the result for top 10 frequent words:

```
[('the', 620), ('and', 427), ('of', 396), ('to', 367), ('I', 326), ('a', 256), ('you', 193), ('in', 190), ('is', 185), ('my', 170)]
```

Create cluster and submit job through Cloud shell:

```
shivam@hae:~/Big Data Analytics/HW0$ gcloud dataproc clusters create notebook-example \
--optional-components=ANACONDA,JUPYTER \
--region=us-east1 \
--image-version=1.3 \
--enable-component-gateway \
--bucket big-data-eecs6893 \
--project big-data-6893-326522 \
--single-node \
--metadata 'PIP_PACKAGES=google-cloud-storage' \
--initialization-actions gs://goog-dataproc-initialization-actions-us-east1/python/pip-install.sh
Waiting on operation [projects/big-data-6893-326522/regions/us-east1/operations/8e5f78ad-c339-3d48-a81c-74cc6f3fd84c].
Waiting for cluster creation operation...
WARNING: When using a single node cluster with less than 1TB of SSD, we strongly recommend provisioning 1TB or larger to ensure consistently high I/O performance. See https://cloud.google.com/compute/docs/risks/performance for more information on disk I/O performance.
Waiting for cluster creation operation...done.
Created [https://dataproc.googleapis.com/v1/projects/big-data-6893-326522/regions/us-east1/clusters/notebook-example] Cluster placed in zone [us-east1-c].
shivam@hae:~/Big Data Analytics/HW0$ cd ~/notebook-example
shivam@hae:~/notebook-example$ gcloud dataproc jobs submit pyspark count_txt_words.py --cluster notebook-example --region=us-east1
Job [ef5416f603f64558bea0d91ae9c81147] submitted.
Waiting for job output...
21/09/23 05:42:58 INFO org.spark_project.jetty.util.log: Logging initialized @2601ms
21/09/23 05:42:58 INFO org.spark_project.jetty.server.Server: jetty-9.3.2-SNAPSHOT, build timestamp: unknown, git hash: unknown
21/09/23 05:42:58 INFO org.spark_project.jetty.server.Server: Started @268ms
21/09/23 05:42:58 INFO org.spark_project.jetty.server.AbstractConnector: Started ServerConnector@694034e2{HTTP/1.1,[http/1.1]}{0.0.0.0:4040}
21/09/23 05:42:58 INFO org.apache.spark.scheduler.FairScheduler: FairScheduler configuration file not found so jobs will be scheduled in FIFO order. To use fair scheduling, configure spark.scheduler.configurationFile to point to a file located in the classpath.
21/09/23 05:42:59 INFO org.apache.hadoop.yarn.Client: RMProxy: Connecting to ResourceManager at notebook-example-m/10.142.0.14:8082
21/09/23 05:43:00 INFO org.apache.hadoop.yarn.Client: AHSProxy: Connecting to Application History server at notebook-example-m/10.142.0.14:10200
21/09/23 05:43:02 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_1632375676436_0001
21/09/23 05:43:11 INFO org.apache.hadoop.mapred.FileInputFormat: Total input files to process : 1
[('the', 620), ('and', 427), ('of', 396), ('to', 367), ('I', 326), ('a', 256), ('you', 193), ('in', 190), ('is', 185), ('my', 170)]
21/09/23 05:43:15 INFO org.spark_project.jetty.server.AbstractConnector: Stopped Spark@694034e2{HTTP/1.1,[http/1.1]}{0.0.0.0:4040}
Job [ef5416f603f64558bea0d91ae9c81147] finished successfully.
done. true
driverControlUrl: gs://big-data-eecs6893/google-cloud-dataproc-metainfo/32c704ee-4e7b-4285-9ce6-fb8316eb0b16/jobs/ef5416f603f64558bea0d91ae9c81147/
driverOutputResourceUrl: gs://big-data-eecs6893/google-cloud-dataproc-metainfo/32c704ee-4e7b-4285-9ce6-fb8316eb0b16/jobs/ef5416f603f64558bea0d91ae9c81147/driveoutput
jobId: ef5416f603f64558bea0d91ae9c81147
placement:
  clusterName: notebook-example
  clusterUuid: 32c704ee-4e7b-4285-9ce6-fb8316eb0b16
pysparkJob:
  mainPythonFileUri: gs://big-data-eecs6893/google-cloud-dataproc-metainfo/32c704ee-4e7b-4285-9ce6-fb8316eb0b16/jobs/ef5416f603f64558bea0d91ae9c81147/staging/count_txt_
words.py
  refId: efs1
  jobId: ef5416f603f64558bea0d91ae9c81147
  projectId: big-data-6893-326522
status:
  state: DONE
  stateStartTime: '2021-09-23T05:43:15.911654Z'
  statusHistory:
    - state: PENDING
      stateStartTime: '2021-09-23T05:42:54.216672Z'
    - state: SETUP_DONE
      stateStartTime: '2021-09-23T05:42:54.238964Z'
    - done: true reported job success
      state: RUNNING
      stateStartTime: '2021-09-23T05:42:54.484821Z'
  yarnApplications:
    - name: count_txt_words.py
      progress: 1.0
      state: FINISHED
      trackingUrl: http://notebook-example-m:8088/proxy/application_1632375676436_0001/
```

Code:

```
In [2]: import pyspark
import sys

# Cloud storage URI where the text file is stored
input_uri = "gs://big-data-eecs6893/input/shakes.txt"

sc = pyspark.SparkContext.getOrCreate()
txt_file = sc.textFile(input_uri)

# Split lines to words, and create key value pairs
words = txt_file.flatMap(lambda line:line.encode("ascii", "ignore").split())
words_kv = words.map(lambda word: (word, 1))

# Add values of RDD by key and sort
counts = words_kv.reduceByKey(lambda v1, v2: v1 + v2).sortBy(lambda x: -x[1])
print(counts.take(10))

[('the', 620), ('and', 427), ('of', 396), ('to', 367), ('I', 326), ('a', 256), ('you', 193), ('in', 190), ('is', 185),
 ('my', 170)]
```

2. With NLTK text pre-processing, the result for top 10 frequent words:

[('macb', 137), ('haue', 122), ('thou', 90), ('enter', 81), ('shall', 68), ('macbeth', 67),
('vpon', 62), ('thee', 61), ('macd', 58), ('th', 57)]

Submit job through Cloud shell:

```
shivamojha@Shivams-MacBook-Air:~/Files/Columbia/Courses/Fall 21/Big Data Analytics/HW0/Code$ gcloud dataproc jobs submit pyspark count_txt_words_nltk.py --cluster notebook-cluster --region us-east1
Job [8e02be63c2b14fdc8494ead25426284] submitted.
Notebook job [notebook-job-1] has been submitted.
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data] Package stopwords is already up-to-date!
21/09/24 02:54:19 INFO org.spark_project.jetty.util.log: Logging initialized @2678ms
21/09/24 02:54:19 INFO org.spark_project.jetty.server.Server: jetty-9.3.z-SNAPSHOT, build timestamp: unknown, git hash: unknown
21/09/24 02:54:19 INFO org.spark_project.jetty.server.Server: Started @2750ms
21/09/24 02:54:19 INFO org.spark_project.jetty.server.AbstractConnector: Started ServerConnector@34cc80@6[HTTP/1.1,[http/1.1]{0.0.0.0:4848}]
21/09/24 02:54:19 WARN org.spark_project.jetty.server.AbstractConnector: fair Scheduler configuration file not found so jobs will be scheduled in FIFO order. To use fair scheduling, configure pools in FairScheduler.xml or set spark.scheduler.allocation.file to a file that contains the configuration.
21/09/24 02:54:20 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at notebook-example-m/10.142.0.15:8032
21/09/24 02:54:21 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at notebook-example-m/10.142.0.15:10200
21/09/24 02:54:23 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_1632442049500_0014
21/09/24 02:54:24 INFO org.apache.hadoop.mapred.FileInputFormat: Total input files to process : 1
[('macb', 137), ('haue', 122), ('thou', 90), ('enter', 81), ('shall', 68), ('macbeth', 67), ('vpon', 62), ('thee', 61), ('macd', 58), ('th', 57)]
21/09/24 02:54:34 INFO org.spark_project.jetty.server.AbstractConnector: Stopped Spark@34cc80@6[HTTP/1.1,[http/1.1]{0.0.0.0:4848}]
Job [8e02be63c2b14fdc8494ead25426284] finished successfully.
done: true
driverControlFilesUri: gs://big-data-eecs6893/google-cloud-dataproc-metainfo/406dc3e8-bb50-4aaa-9444-c6c60570a7d5/jobs/8e02be63c2b14fdc8494ead25426284/
driverOutputResourceUri: gs://big-data-eecs6893/google-cloud-dataproc-metainfo/406dc3e8-bb50-4aaa-9444-c6c60570a7d5/jobs/8e02be63c2b14fdc8494ead25426284/driveroutput
jobId: [REDACTED]
jobName: [REDACTED]
clusterName: notebook-example
clusterUuid: 406dc3e8-bb50-4aaa-9444-c6c60570a7d5
sparkJobId:
mainPythonFileUri: gs://big-data-eecs6893/google-cloud-dataproc-metainfo/406dc3e8-0b50-4aaa-9444-c6c60570a7d5/jobs/8e02be63c2b14fdc8494ead25426284/staging/count_txt_words_nltk.py
referenceId:
jobId: 8e02be63c2b14fdc8494ead25426284
projectId: big-data-6893-326522
status:
state: DONE
stateStartTime: '2021-09-24T02:54:39.099256Z'
statusEndTime: null
- state: PENDING
  stateStartTime: '2021-09-24T02:54:16.219798Z'
- state: SETUP DONE
  stateStartTime: '2021-09-24T02:54:16.245252Z'
- details: Agent reported job success
  state: RUNNING
  stateStartTime: '2021-09-24T02:54:16.499979Z'
yarnApplications:
- name: count_txt_words_nltk.py
  progress: 1.0
  state: FINISHED
  trackingUrl: http://notebook-example-m:8088/proxy/application_1632442049500_0014
```

Code:

```
1 import pyspark
2 import sys
3 import nltk
4
5 nltk.download('stopwords')
6 stop_words = set(nltk.corpus.stopwords.words('english'))
7
8
9 def remove_stop_words(line):
10     """
11     Function to filter out stop words provided by NLTK package
12     """
13     words_list = []
14     text_list = nltk.tokenize.RegexpTokenizer(r'\w+').tokenize(line)
15     for word in text_list:
16         word = word.encode("ascii", "ignore").lower()
17         if word not in stop_words:
18             words_list.append(word)
19     return words_list
20
21
22 sc = pyspark.SparkContext()
23 lines = sc.textFile("gs://big-data-eecs6893/input/shakes.txt")
24 words_filter = lines.flatMap(remove_stop_words)
25 words_kv = words_filter.map(lambda x: (x,1))
26
27 # Add values of RDD by key and sort
28 counts = words_kv.reduceByKey(lambda v1, v2: v1 + v2).sortBy(lambda x: -x[1])
29 print(counts.take(10))
```