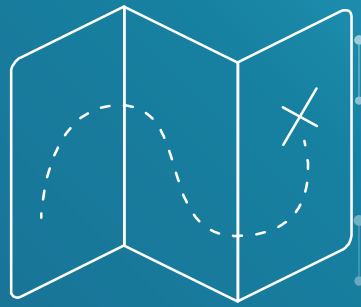
PageRank Implementation using PySpark and Scala on GCP

Belsabel Woldemichael

Contents

- Introduction
- Design
- Implementation
- Test
- Enhancement Ideas
- Conclusion



Introduction

Objective: Implement the PageRank algorithm using both PySpark and Scala on a Google Cloud Dataproc cluster to analyze link structures and compute page importance scores.

Technologies Used:

- Apache Spark: Distributed computing framework for processing large datasets.
- PySpark: Python API for Spark, used for data manipulation and algorithm implementation.
- Scala: Programming language for high-performance computing in Spark.
- Google Cloud Platform (GCP): Cloud environment for data storage (GCS), computation (Dataproc), and job orchestration (Cloud Shell, gcloud SDK).

Design

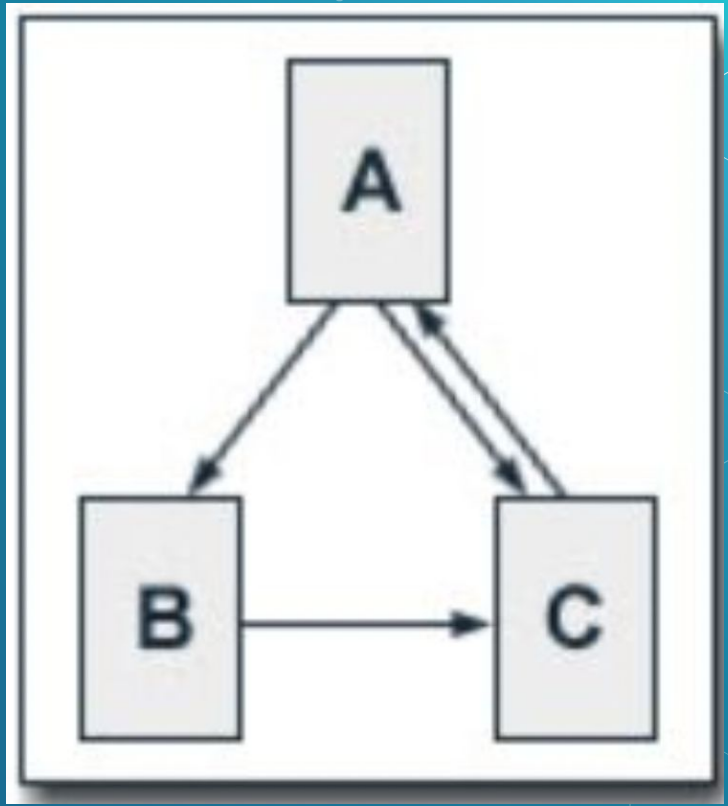
The following is the manual calculation of the diagram below.

Webpage A links to B and C.
Webpage B links to C.
Webpage C links back to A.

Initial Setup:
Each webpage starts with a PageRank value of 1.
Damping factor (d) = 0.85.

First Iteration:
PR(A) = 1−d+d×(PR(C)/1) = 1−0.85+0.85×1 = 1
PR(B) = 1−d+d×(PR(A)/2) = 1−0.85+0.85×1/2 = 0.575
PR(C) = 1−d+d×((PR(A)/2)+PR(B)/1) = 1−0.85+0.85×(0.5+1) = 1.425

Second Iteration:
PageRank (A) = 1 – 0.85 + 0.85 * 1.425 = 1.36125
PageRank (B) = 1 – 0.85 + 0.85 * 0.5 = 0.575
PageRank (C) = 1 – 0.85 + 0.85 * 1.075 = 1.0637



Implementation and Test

Implementation and Testing of The PageRank calculation

A. Using PySpark

1. Set up the GCP Environment

- Ensure Google Cloud SDK is Installed
  gcloud version
- Update Google Cloud SDK

```
app-engine-go 1.9.76
app-engine-java 2.0.28
app-engine-python 1.9.113
app-engine-python-extras 1.9.106
beta 2024.06.14
bigtable
bq 2.1.6
bundled-python3-unix 3.11.8
cbt 1.20.0
cloud-datastore-emulator 2.3.1
cloud-run-proxy 0.5.0
core 2024.06.14
gcloud-crc32c 1.0.0
gke-gcloud-auth-plugin 0.5.8
gsutil 5.30
kpt 1.0.0-beta.50
kubectl 1.27.14
local-extract 1.5.9
minikube 1.33.1
nomos 1.18.1-rc.1
package-go-module 0.4.0
pubsub-emulator 0.8.14
skaffold 2.11.1
belsabelteklemariam@cloudshell:~ (cs570-427815)$
```

```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ sudo apt-get update
********************************************************************************
You are running apt-get inside of Cloud Shell. Note that your Cloud Shell
machine is ephemeral and no system-wide change will persist beyond session end.

To suppress this warning, create an empty ~/.cloudshell/no-apt-get-warning file.
The command will automatically proceed in 5 seconds or on any key.

Visit https://cloud.google.com/shell/help for more information.
********************************************************************************
Get:1 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Get:2 https://cli.github.com/packages stable InRelease [3,917 B]
Get:3 https://packages.microsoft.com/ubuntu/22.04/prod jammy InRelease [3,632 B]
Get:4 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [41.5 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:6 https://cli.github.com/packages stable/main amd64 Packages [346 B]
Get:7 https://packages.microsoft.com/ubuntu/22.04/prod jammy/main amd64 Packages [160 kB]
Get:8 https://packages.microsoft.com/ubuntu/22.04/prod jammy/main armhf Packages [13.9 kB]
Get:9 https://packages.microsoft.com/ubuntu/22.04/prod jammy/main arm64 Packages [39.2 kB]
Get:10 http://security.ubuntu.com/ubuntu jammy-security/multiverse Sources [12.1 kB]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/main Sources [351 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/universe Sources [250 kB]
Get:13 https://apt.postgresql.org/pub/repos/apt jammy-pgdg InRelease [123 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/restricted Sources [75.9 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1,974 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [1,114 kB]
Get:17 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2,566 kB]
Get:18 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [44.7 kB]
Get:19 https://apt.postgresql.org/pub/repos/apt jammy-pgdg/main amd64 Packages [546 kB]
Hit:20 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:21 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:22 https://packages.cloud.google.com/apt gcsfuse-jammy InRelease [1,225 B]
Get:23 https://packages.cloud.google.com/apt cloud-sdk InRelease [1,616 B]
```

Install PySpark

- sudo apt-get update
- sudo apt-get install -y python3-pip
- sudo pip3 install pyspark

```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ sudo apt-get install -y python3-pip
********************************************************************************
You are running apt-get inside of Cloud Shell. Note that your Cloud Shell
machine is ephemeral and no system-wide change will persist beyond session end.

To suppress this warning, create an empty ~/.cloudshell/no-apt-get-warning file.
The command will automatically proceed in 5 seconds or on any key.

Visit https://cloud.google.com/shell/help for more information.
********************************************************************************
E: Could not get lock /var/lib/dpkg/lock-frontend. It is held by process 2512 (apt-get)
N: Be aware that removing the lock file is not a solution and may break your system.
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontend), is another process using it?
belsabelteklemariam@cloudshell:~ (cs570-427815)$ sudo pip3 install pyspark
Collecting pyspark
  Downloading pyspark-3.5.1.tar.gz (317.0 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 317.0/317.0 MB 3.0 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Collecting py4j==0.10.9.7
  Downloading py4j-0.10.9.7-py2.py3-none-any.whl (200 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 200.5/200.5 KB 29.5 MB/s eta 0:00:00
Building wheels for collected packages: pyspark
  Building wheel for pyspark (setup.py) ... done
  Created wheel for pyspark: filename=pyspark-3.5.1-py2.py3-none-any.whl size=317488491 sha256=822b86cd3df8e19cad02a774e7087f253f7dd08667c47a09f0444bd1e9e85547
  Stored in directory: /root/.cache/pip/wheels/80/1d/60/2c256ed38dddce2fdd93be545214a63e02fbd8d74fb0b7f3a6
Successfully built pyspark
Installing collected packages: py4j, pyspark
Successfully installed py4j-0.10.9.7 pyspark-3.5.1
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment in
stead: https://pip.pypa.io/warnings/venv
belsabelteklemariam@cloudshell:~ (cs570-427815)$
```

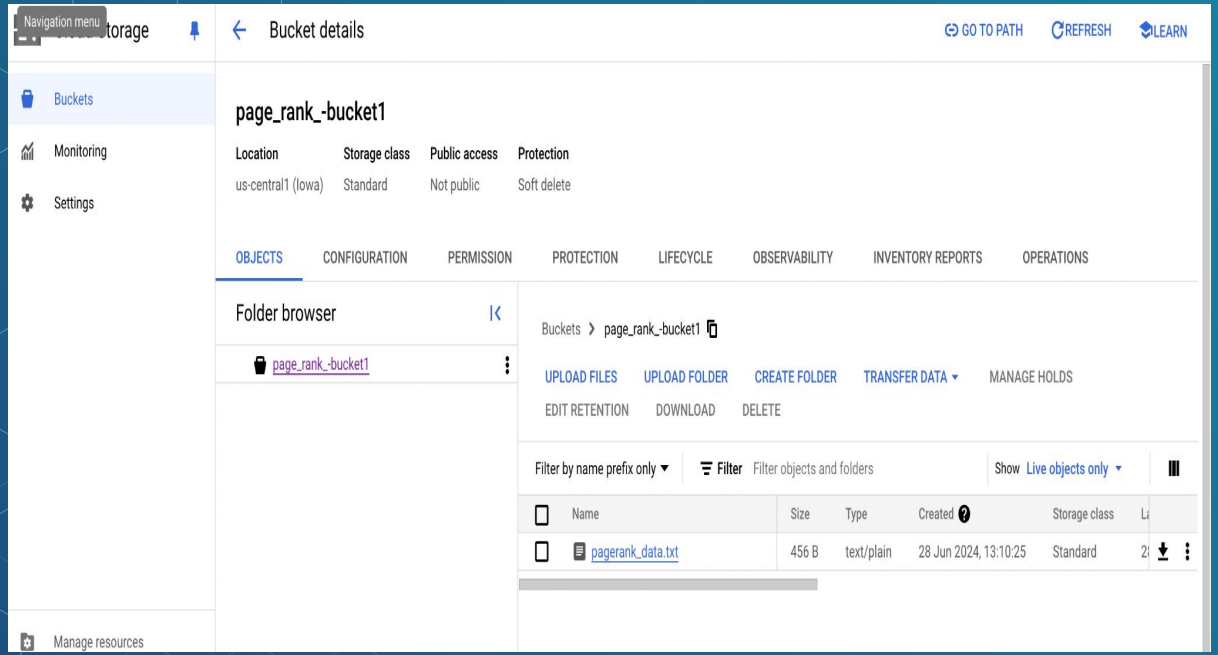
Verify PySpark Installation:

pyspark –version

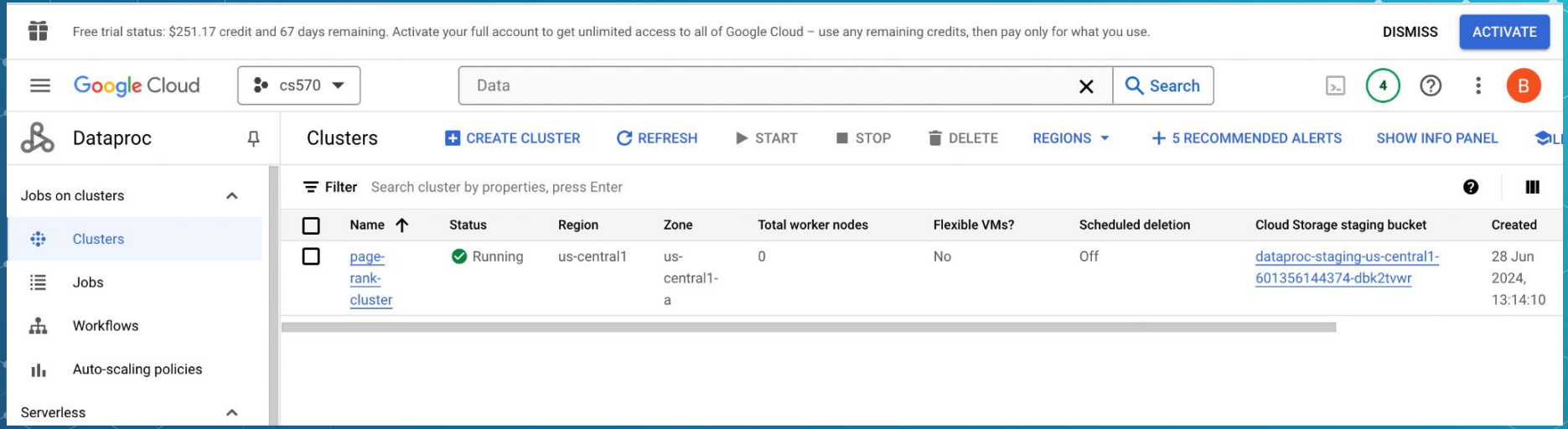
```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ pyspark --version
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 3.5.1
      /_/

Using Scala version 2.12.18, OpenJDK 64-Bit Server VM, 17.0.11
Branch HEAD
Compiled by user heartsavior on 2024-02-15T11:24:58Z
Revision fd86f85e181fc2dc0f50a096855acf83a6cc5d9c
Url https://github.com/apache/spark
Type --help for more information.
```

2. Solve the question using pyspark

- Go to the APIs & Services Dashboard and enable the Dataproc API.
- Create a Google Cloud Storage (GCS) Bucket and upload page-rank-data.txt



Create a Dataproc Cluster



Prepare PySpark Script:

Create a file pagerank.py with the following content:

```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ cat pagerank.py
from pyspark import SparkConf, SparkContext

def computeContribs(urls, rank):
    num_urls = len(urls)
    for url in urls:
        yield (url, rank / num_urls)

def parseNeighbors(urls):
    parts = urls.split()
    if len(parts) >= 2:
        return parts[0], parts[1]
    else:
        return None

if __name__ == "__main__":
    # Spark configuration
    conf = SparkConf().setAppName("PythonPageRank")
    sc = SparkContext(conf=conf)

    # Load input file
    lines = sc.textFile("gs://page_rank_bucket1/pagerank_data.txt")

    # Parse neighbors
    links = lines.map(parseNeighbors).filter(lambda x: x is not None).distinct().groupByKey().cache()

    # Initialize ranks
    ranks = links.map(lambda url_neighbors: (url_neighbors[0], 1.0))

    # Number of iterations
    iterations = 10

    for iteration in range(iterations):
        # Calculate contributions
        contribs = links.join(ranks).flatMap(
```

Submit PySpark Job:

Use the following command to submit the PySpark job to Dataproc:

```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ gcloud dataproc jobs submit pyspark pagerank.py \
  --cluster=page-rank-cluster \
  --region=us-central1 \
  -- gs://page_rank_bucket1/pagerank_data.txt 10
Job [7364d5e707f64f2790d737943b89bd5e] submitted.
Waiting for job output...
24/06/27 14:46:29 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
24/06/27 14:46:29 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
24/06/27 14:46:29 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
24/06/27 14:46:29 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
24/06/27 14:46:29 INFO org.sparkproject.jetty.util.log: Logging initialized @4231ms to org.sparkproject.jetty.util.log.Slf4jLog
24/06/27 14:46:29 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.668Z; git: b881a572662e1943a14ae12e7e1207989f218b74; jvm 1.8.0_412-
b08
24/06/27 14:46:29 INFO org.sparkproject.jetty.server.Server: Started @4385ms
24/06/27 14:46:29 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@2e2a2171{HTTP/1.1, (http/1.1)}{0.0.0.0:44063}
24/06/27 14:46:30 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at page-rank-cluster-m/10.128.0.11:8032
24/06/27 14:46:30 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at page-rank-cluster-m/10.128.0.11:10200
24/06/27 14:46:32 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
24/06/27 14:46:32 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
24/06/27 14:46:33 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_1719478903994_0006
24/06/27 14:46:34 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at page-rank-cluster-m/10.128.0.11:8030
24/06/27 14:46:36 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_get_file_status. latencyMs=285; previousMaxLatencyMs=
0; operationCount=1; context=gs://dataproc-temp-us-central1-323098012664-kn6qemw4/cf192013-0618-49a1-aa4c-61519e77ea61/spark-job-history
24/06/27 14:46:36 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageImpl: Ignoring exception of type GoogleJsonResponseException; verifie
d object already exists with desired state.
24/06/27 14:46:36 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_mkdirs. latencyMs=164; previousMaxLatencyMs=0; operat
ionCount=1; context=gs://dataproc-temp-us-central1-323098012664-kn6qemw4/cf192013-0618-49a1-aa4c-61519e77ea61/spark-job-history
24/06/27 14:46:38 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_get_file_status. latencyMs=393; previousMaxLatencyMs=
```

```
24/06/27 18:48:19 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_get_file_status. latenc
0; operationCount=1; context=gs://dataproc-temp-us-central1-323098012664-kn6qemw4/cf192013-0618-49a1-aa4c-61519e77ea61/spark-job-history
24/06/27 18:48:19 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageImpl: Ignoring exception of type Google
d object already exists with desired state.
24/06/27 18:48:19 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_mkdirs. latencyMs=169;
ionCount=1; context=gs://dataproc-temp-us-central1-323098012664-kn6qemw4/cf192013-0618-49a1-aa4c-61519e77ea61/spark-job-history
24/06/27 18:48:21 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_glob_status. latencyMs=
perationCount=1; context=path=gs://page_rank_bucket1/pagerank_data.txt; pattern=org.apache.hadoop.mapred.FileInputFormat$MultiPathFilter@577024b3
24/06/27 18:48:21 INFO org.apache.hadoop.mapred.FileInputFormat: Total input files to process : 1
Iteration 2
C has rank: 1.06375
A has rank: 1.3612499999999996
B has rank: 0.575
24/06/27 18:48:30 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped Spark@1e7102e3{HTTP/1.1, (http/1.1)}{0.0.0.0:0}
24/06/27 18:48:30 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_rename. latencyMs=199;
ionCount=1; context=rename(gs://dataproc-temp-us-central1-323098012664-kn6qemw4/cf192013-0618-49a1-aa4c-61519e77ea61/spark-job-history/application_17
 gs://dataproc-temp-us-central1-323098012664-kn6qemw4/cf192013-0618-49a1-aa4c-61519e77ea61/spark-job-history/application_1719478903994_0007)
Job [2922dc89f71f49e0a374faf47cbd8db7] finished successfully.
done: true
driverControlFilesUri: gs://dataproc-staging-us-central1-323098012664-sl1dclvo/google-cloud-dataproc-metainfo/cf192013-0618-49a1-aa4c-61519e77ea61/jo
d8db7/
driverOutputResourceUri: gs://dataproc-staging-us-central1-323098012664-sl1dclvo/google-cloud-dataproc-metainfo/cf192013-0618-49a1-aa4c-61519e77ea61/
cbd8db7/driveroutput
jobUuid: 2edd815a-7265-35ed-ae5a-6cf20d4b7efc
placement:
  clusterName: page-rank-cluster
  clusterUuid: cf192013-0618-49a1-aa4c-61519e77ea61
```

- As it can be seen in the screenshot, output of the ranks of the pages of two iterations were as follows.

```
Iteration 2
C has rank: 1.06375
A has rank: 1.361249
B has rank: 0.575
```

B. Using Scala

1. Set up Scala on GCP

sudo apt-get update

sudo apt-get install scala

2. Verify Scala Installation:

scala -version

```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ sudo apt-get install scala
********************************************************************************
You are running apt-get inside of Cloud Shell. Note that your Cloud Shell
machine is ephemeral and no system-wide change will persist beyond session end.

To suppress this warning, create an empty ~/.cloudshell/no-apt-get-warning file.
The command will automatically proceed in 5 seconds or on any key.

Visit https://cloud.google.com/shell/help for more information.
********************************************************************************
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libhawtjni-runtime-java libjansi-java libjansi-native-java libjline2-java scala-library scala-parser-combinators scala-xml
Suggested packages:
  scala-doc
The following NEW packages will be installed:
  libhawtjni-runtime-java libjansi-java libjansi-native-java libjline2-java scala scala-library scala-parser-combinators scala-xml
0 upgraded, 8 newly installed, 0 to remove and 44 not upgraded.
2 not fully installed or removed.
Need to get 25.1 MB of archives.
After this operation, 28.6 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libhawtjni-runtime-java all 1.17-1 [28.8 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libjansi-native-java all 1.8-1 [23.8 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libjansi-java all 1.18-1 [56.8 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libjline2-java all 2.14.6-4 [150 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy/universe amd64 scala-library all 2.11.12-5 [9,586 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy/universe amd64 scala-parser-combinators all 1.0.3-3.1 [365 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy/universe amd64 scala-xml all 1.0.3-3.1 [615 kB]
```

```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ scala -version
Scala code runner version 2.11.12 -- Copyright 2002-2017, LAMP/EPFL
```

Install SDKMAN!:

curl -s "https://get.sdkman.io" | bash

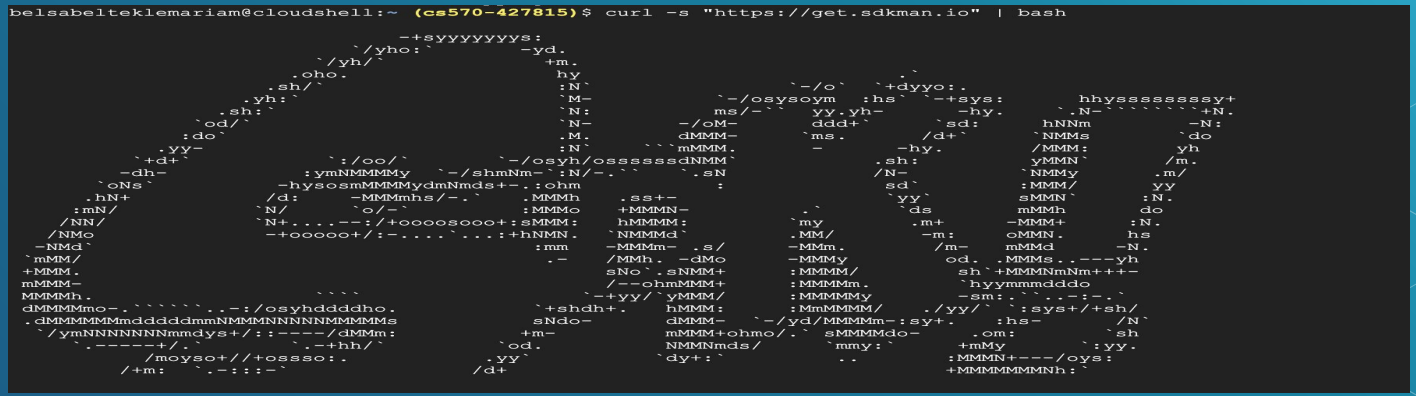
Initialize SDKMAN!:

○ source "$HOME/.sdkman/bin/sdkman-init.sh"

Install sbt:

o sdk install sbt

```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ curl -s "https://get.sdkman.io" | bash
```



```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ source "$HOME/.sdkman/bin/sdkman-init.sh"
```

```
belsabelteklemariam@cloudshell:~ (cs570-427815)$ sdk install sbt

Downloading: sbt 1.10.0

In progress...

################################################################################################################################################################

Installing: sbt 1.10.0
Done installing!


Setting sbt 1.10.0 as default.
```

2. **Solve the question using Scala**

- Create build.sbt file in current directory and add the following content:

  Vi build.sbt

```
name := "PageRank"
version := "1.0"
scalaVersion := "2.12.15"
libraryDependencies += "org.apache.spark" %% "spark-core" % "3.2.0"
~
~
~
~
```

Create a directory named src/main/scala in the current directory.

- mkdir -p src/main/scala

```
belsabelteklemariam@cloudshell:~/page_rank_project (cs570-427815)$ mkdir -p src/main/scala
```

Create a file named PageRank.scala with the following content.
This script calculates PageRank using Apache Spark in Scala:

```
belsabelteklemariam@cloudshell:~/page_rank_project/src/main/scala (cs570-427815)$ cat pagerank.scala
import org.apache.spark.{SparkConf, SparkContext}
import org.apache.spark.HashPartitioner

object PageRank {
  def main(args: Array[String]): Unit = {
    // Spark configuration
    val sparkConf = new SparkConf().setAppName("PageRank")
    val sc = new SparkContext(sparkConf)

    // Load input file
    val lines = sc.textFile(args(0))

    // Parse neighbors
    val links = lines.map { s =>
      val parts = s.split("\\s+")
      (parts(0), parts(1))
    }.distinct().groupByKey().partitionBy(new HashPartitioner(100)).persist()

    // Initialize ranks
    var ranks = links.mapValues(_ => 1.0)

    // Number of iterations
    val iterations = args(1).toInt

    // Run PageRank algorithm for `iterations` times
    for (i <- 1 to iterations) {
```

Run sbt to compile and package your Scala code into a JAR file, in the previous directory.

```
belsabelteklemariam@cloudshell:~/page_rank_project (cs570-427815)$ sbt package
[info] Updated file /home/belsabelteklemariam/page_rank_project/project/build.properties: set sbt.version to 1.10.0
[info] welcome to sbt 1.10.0 (Ubuntu Java 17.0.11)
[info] loading project definition from /home/belsabelteklemariam/page_rank_project/project
[info] loading settings for project page_rank_project from build.sbt ...
[info] set current project to PageRank (in build file:/home/belsabelteklemariam/page_rank_project/)
[info] Updating pagerank_2.12
https://repo1.maven.org/maven2/org/scala-lang/scala-library/2.12.15/scala-library-2.12.15.pom
  100.0% [##########] 1.6 KiB (2.1 KiB / s)
https://repo1.maven.org/maven2/org/apache/spark/spark-core_2.12/3.2.0/spark-core_2.12-3.2.0.pom
  100.0% [##########] 32.6 KiB (43.8 KiB / s)
https://repo1.maven.org/maven2/org/apache/spark/spark-parent_2.12/3.2.0/spark-parent_2.12-3.2.0.pom
  100.0% [##########] 128.0 KiB (3.7 MiB / s)
https://repo1.maven.org/maven2/org/apache/apache/18/apache-18.pom
  100.0% [##########] 15.3 KiB (765.0 KiB / s)
```

```
belsabelteklemariam@cloudshell:~/page_rank_project (cs570-427815)$ ls target/scala-2.12
classes  pagerank_2.12-1.0.jar  sync  update  zinc
```

Upload JAR File to Google Cloud Storage (GCS)

- gsutil cp target/scala-2.12/pagerank_2.12-1.0.jar gs://page_rank_bucket1/

```
belsabelteklemariam@cloudshell:~/page_rank_project (cs570-427815)$ gsutil cp target/scala-2.12/pagerank_2.12-1.0.jar gs://page_rank_-bucket1/
Copying file://target/scala-2.12/pagerank_2.12-1.0.jar [Content-Type=application/java-archive]...
/ [1 files][  4.0 KiB/  4.0 KiB]
Operation completed over 1 objects/4.0 KiB.
```

Submit Scala Job to Dataproc

```
belsabelteklemariam@cloudshell:~/page_rank_project (cs570-427815)$ gcloud dataproc jobs submit spark \
    --cluster=page-rank-cluster \
    --region=us-central1 \
    --class=PageRank \
    --jars=gs://page_rank_-bucket1/pagerank_2.12-1.0.jar \
    -- gs://page-rank_-bucket1/pagerank_data.txt 1 gs://page_rank_-bucket1/ranks
Job [0eb0dc1d47924e3a9a4814b4193422c9] submitted.
Waiting for job output...
24/06/27 14:11:11 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
24/06/27 14:11:11 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
24/06/27 14:11:11 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
24/06/27 14:11:11 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
24/06/27 14:11:11 INFO org.sparkproject.jetty.util.log: Logging initialized @4675ms to org.sparkproject.jetty.util.log.Slf4jLog
24/06/27 14:11:11 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.668Z; git: b881a572662e1943a14ae12e7e1207989f218b74; jvm 1.8.0_412-
b08
24/06/27 14:11:11 INFO org.sparkproject.jetty.server.Server: Started @4815ms
24/06/27 14:11:11 INFO org.sparkproject.jetty.server.AbstractConnector: Started ServerConnector@65ec8b24{HTTP/1.1, (http/1.1)}{0.0.0.0:39641}
24/06/27 14:11:12 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at page-rank-cluster-m/10.128.0.11:8032
24/06/27 14:11:13 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at page-rank-cluster-m/10.128.0.11:10200
24/06/27 14:11:13 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
24/06/27 14:11:13 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
24/06/27 14:11:14 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_1719478903994_0005
24/06/27 14:11:15 INFO org.apache.hadoop.yarn.client.RMProxy: Connecting to ResourceManager at page-rank-cluster-m/10.128.0.11:8030
24/06/27 14:11:17 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_get_file_status. latencyMs=296; previousMaxLatencyMs=
0; operationCount=1; context=gs://dataproc-temp-us-central1-323098012664-kn6qemw4/cf192013-0618-49a1-aa4c-61519e77ea61/spark-job-history
24/06/27 14:11:18 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageImpl: Ignoring exception of type GoogleJsonResponseException; verifie
d object already exists with desired state.
24/06/27 14:11:18 INFO com.google.cloud.hadoop.fs.gcs.GhfsStorageStatistics: Detected potential high latency for operation op_mkdirs. latencyMs=166; previousMaxLatencyMs=0; operat
```

```
Job [0eb0dc1d47924e3a9a4814b4193422c9] finished successfully.
done: true
driverControlFilesUri: gs://dataproc-staging-us-central1-323098012664-sl1dclvo/google-cloud-dataproc-metainfo/cf192013-0618-49a1-aa4c-61519e77ea61/jobs/0eb0dc1d47924e3a9a4814b4193
422c9/
driverOutputResourceUri: gs://dataproc-staging-us-central1-323098012664-sl1dclvo/google-cloud-dataproc-metainfo/cf192013-0618-49a1-aa4c-61519e77ea61/jobs/0eb0dc1d47924e3a9a4814b41
93422c9/driveroutput
jobUuid: f52369af-521e-374d-8f36-d64cd4eadcc2
placement:
  clusterName: page-rank-cluster
  clusterUuid: cf192013-0618-49a1-aa4c-61519e77ea61
reference:
  jobId: 0eb0dc1d47924e3a9a4814b4193422c9
  projectId: verdant-legacy-427208-r5
sparkJob:
  args:
  - gs://page_rank_bucket1/pagerank_data.txt
  - '1'
  - gs://page_rank_bucket1/ranks
  jarFileUris:
name := "PageRank"
  - gs://page_rank_bucket1/pagerank_2.12-1.0.jar
  mainClass: PageRank
status:
  state: DONE
  stateStartTime: '2024-06-27T14:11:44.367249Z'
statusHistory:
- state: PENDING
import org.apache.spark.{SparkConf, SparkContext}
- state: PENDING
```

Finally The following command retrieves and displays the content of all files stored in the directory gs://page_rank_bucket1/ranks/ in Google Cloud Storage (GCS).

- gsutil cat gs://page_rank_bucket1/ranks/*

```
belsabelteklemariam@cloudshell:~/page_rank_project (cs570-427815)$ gsutil cat gs://page_rank_bucket1/ranks/*
(A,1.0)
(B,0.575)
(C,1.4249999999999998)
```

Enhancement Ideas

Graph Visualization:

- Integrate tools like D3.js or GraphX to visualize the graph structure and PageRank scores. This visualization can provide intuitive insights into page relationships and importance.

Real-Time PageRank Updates:

- Implement a streaming data pipeline using Apache Kafka or Google Cloud Pub/Sub to update PageRank scores in near real-time as new data or changes occur in the graph structure.

Performance Optimization:

- Experiment with different partitioning strategies in Spark to optimize data distribution and processing efficiency, especially for large-scale graphs.

Error Handling and Fault Tolerance:

- Enhance error handling mechanisms in your PySpark or Scala scripts to gracefully manage failures and retries during job execution on Dataproc clusters.

Conclusion

Achievements:

- Successfully implemented PageRank algorithm using PySpark and Scala on Google Cloud Platform (GCP).
- Developed a scalable solution for analyzing graph structures and computing page importance scores using distributed computing techniques.

Learnings:

- Acquired proficiency in setting up Apache Spark environments on GCP, including installation, configuration, and job submission.
- Gained insights into data preprocessing, partitioning strategies, and iterative algorithm implementation in both PySpark and Scala.

Challenges Overcome:

- Overcame challenges related to cluster configuration, data handling in distributed environments, and optimizing algorithm performance for large datasets.
- Managed complexities of cloud infrastructure, including resource allocation, job monitoring, and debugging.
-

References

Example of PageRank

PageRank Algorithm

Spark Scala - PageRank Implementation

Github Link

https://github.com/BelsabelTekle/Cloud_Computing/tree/main/Spark/Page_Rank