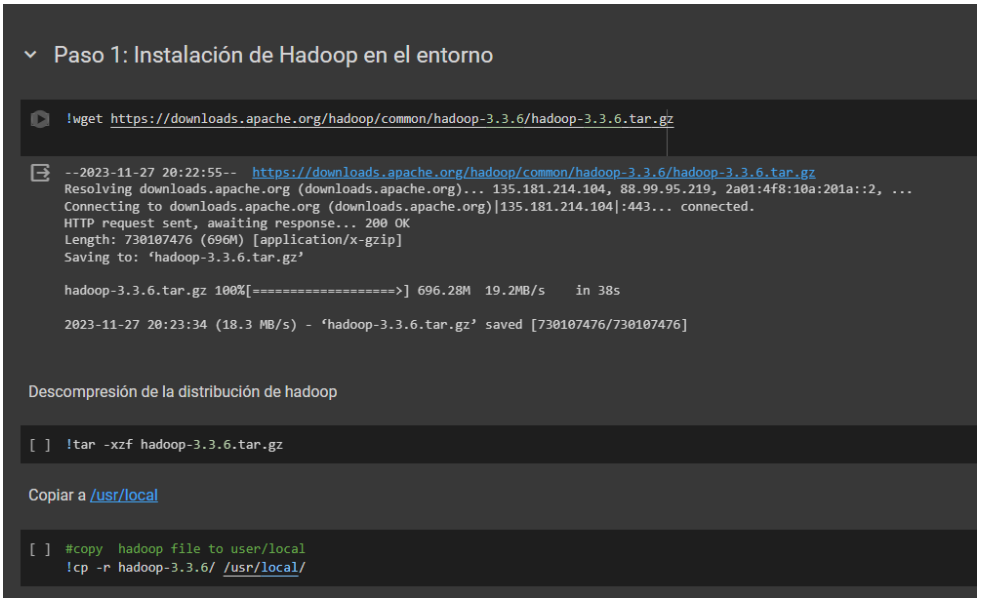
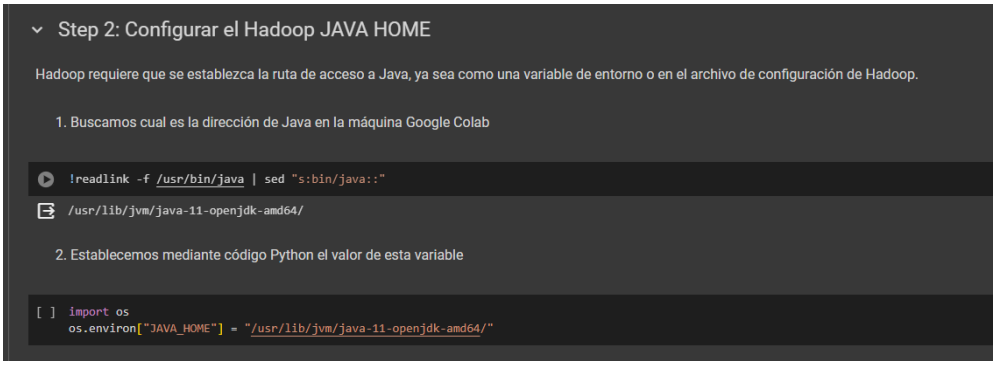
Néstor Batista Díaz

Práctica de MapReduce en Colab con Python





Paso 3: Ejecutando Hadoop

```
#Running Hadoop
!/usr/local/hadoop-3.3.6/bin/hadoop
```

```
Usage: hadoop [OPTIONS] SUBCOMMAND [SUBCOMMAND OPTIONS]
 or    hadoop [OPTIONS] CLASSNAME [CLASSNAME OPTIONS]
  where CLASSNAME is a user-provided Java class

  OPTIONS is none or any of:

buildpaths                       attempt to add class files from build tree
--config dir                     Hadoop config directory
--debug                          turn on shell script debug mode
--help                           usage information
hostnames list[,of,host,names]   hosts to use in worker mode
hosts filename                   list of hosts to use in worker mode
loglevel level                   set the log4j level for this command
workers                          turn on worker mode

  SUBCOMMAND is one of:


    Admin Commands:

daemonlog     get/set the log level for each daemon

    Client Commands:

archive       create a Hadoop archive
checknative   check native Hadoop and compression libraries availability
classpath     prints the class path needed to get the Hadoop jar and the required libraries
conftest      validate configuration XML files
credential    interact with credential providers
distch        distributed metadata changer
distcp        copy file or directories recursively
dtutil        operations related to delegation tokens
envvars       display computed Hadoop environment variables
fs            run a generic filesystem user client
gridmix       submit a mix of synthetic job, modeling a profiled from production load
jar <jar>     run a jar file. NOTE: please use "yarn jar" to launch YARN applications, not this
              command.
jnipath       prints the java.library.path
kdiag         Diagnose Kerberos Problems
```

```
[ ] !mkdir ~/input
    !cp /usr/local/hadoop-3.3.6/etc/hadoop/*.xml ~/input
```

```
!ls ~/input
```

```
capacity-scheduler.xml  hadoop-policy.xml  hdfs-site.xml    kms-acls.xml  mapred-site.xml
core-site.xml           hdfs-rbf-site.xml  httpfs-site.xml  kms-site.xml  yarn-site.xml
```

```
[ ] !/usr/local/hadoop-3.3.6/bin/hadoop jar /usr/local/hadoop-3.3.6/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.6.jar grep ~/input ~/grep_example 'allowed[.]*'
```

```
2023-11-27 20:25:16,752 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2023-11-27 20:25:17,016 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2023-11-27 20:25:17,017 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2023-11-27 20:25:17,388 INFO input.FileInputFormat: Total input files to process : 10
2023-11-27 20:25:17,427 INFO mapreduce.JobSubmitter: number of splits:10
2023-11-27 20:25:17,866 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local1156866682_0001
2023-11-27 20:25:17,866 INFO mapreduce.JobSubmitter: Executing with tokens: []
2023-11-27 20:25:18,133 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2023-11-27 20:25:18,134 INFO mapreduce.Job: Running job: job_local1156866682_0001
2023-11-27 20:25:18,143 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2023-11-27 20:25:18,153 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitterFactory
2023-11-27 20:25:18,155 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2023-11-27 20:25:18,155 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2023-11-27 20:25:18,156 INFO mapred.LocalJobRunner: OutputCommitter is org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter
2023-11-27 20:25:18,232 INFO mapred.LocalJobRunner: Waiting for map tasks
2023-11-27 20:25:18,233 INFO mapred.LocalJobRunner: Starting task: attempt_local1156866682_0001_m_000000_0
2023-11-27 20:25:18,283 INFO output.PathOutputCommitterFactory: No output committer factory defined, defaulting to FileOutputCommitterFactory
2023-11-27 20:25:18,287 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
```

```
!cat ~/grep_example/*
```

```
22      allowed.
1       allowed
```

Download 20newsgroups dataset available at http://qwone.com/~jason/20Newsgroups.

```
[ ] !wget http://qwone.com/~jason/20Newsgroups/20news-18828.tar.gz

    !tar -xzf 20news-18828.tar.gz
```

```
--2023-11-27 20:25:29--  http://qwone.com/~jason/20Newsgroups/20news-18828.tar.gz
Resolving qwone.com (qwone.com)... 173.48.205.131
Connecting to qwone.com (qwone.com)|173.48.205.131|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 14666916 (14M) [application/x-gzip]
Saving to: ‘20news-18828.tar.gz’

20news-18828.tar.gz 100%[===================>]  13.99M  13.0MB/s    in 1.1s

2023-11-27 20:25:30 (13.0 MB/s) - ‘20news-18828.tar.gz’ saved [14666916/14666916]
```

Hadoop Streaming

Hadoop streaming permite crear y ejecutar trabajos Map/Reduce con cualquier ejecutable o script como mapeador y/o reductor.

Mas información sobre Map/Reduce en el siguiente enlace

```
!find / -name 'hadoop-streaming*.jar'
```

```
/usr/local/hadoop-3.3.6/share/hadoop/tools/sources/hadoop-streaming-3.3.6-sources.jar
/usr/local/hadoop-3.3.6/share/hadoop/tools/sources/hadoop-streaming-3.3.6-test-sources.jar
/usr/local/hadoop-3.3.6/share/hadoop/tools/lib/hadoop-streaming-3.3.6.jar
find: ‘/proc/59/task/59/net’: Invalid argument
find: ‘/proc/59/net’: Invalid argument
/content/hadoop-3.3.6/share/hadoop/tools/sources/hadoop-streaming-3.3.6-sources.jar
/content/hadoop-3.3.6/share/hadoop/tools/sources/hadoop-streaming-3.3.6-test-sources.jar
/content/hadoop-3.3.6/share/hadoop/tools/lib/hadoop-streaming-3.3.6.jar
```

mapper.py

Leerá los datos de *STDIN*, los dividirá en palabras y enviará a *STDOUT* una lista de líneas que asignan las palabras a sus recuentos.

```
!cat /content/mapper.py
```

```
# -*- coding: utf-8 -*-
"""mapper.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1yCwGyMXJT2qt3_58aLOOiJXO0GIaPcJd
"""



import sys
import io
import re
import nltk
nltk.download('stopwords',quiet=True)
from nltk.corpus import stopwords
punctuations = '''!()-[]{};:'"\,<>./?@#$%^&*_~'''

stop_words = set(stopwords.words('english'))
input_stream = io.TextIOWrapper(sys.stdin.buffer, encoding='latin1')
for line in input_stream:
  line = line.strip()
  line = re.sub(r'[^\w\s]', '',line)
  line = line.lower()
  for x in line:
    if x in punctuations:
      line=line.replace(x, " ")

  words=line.split()
  for word in words:
    if word not in stop_words:
      print('%s\t%s' % (word, 1))
```

reducer.py

Leerá los resultados de mapper.py desde STDIN y sumará las ocurrencias de cada palabra hasta llegar a un recuento final, y luego enviará sus resultados a STDOUT.

```
[ ] !cat /content/reducer.py
```

```
# -*- coding: utf-8 -*-
"""reducer.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1YzJ-vUsO5VYCyMrfPMow3s2IdxXkyQ0i
"""

from operator import itemgetter
import sys

current_word = None
current_count = 0
word = None

# input comes from STDIN
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip()
    line=line.lower()

    # parse the input we got from mapper.py
    word, count = line.split('\t', 1)
    try:
      count = int(count)
    except ValueError:
      #count was not a number, so silently
      #ignore/discard this line
      continue

    # this IF-switch only works because Hadoop sorts map output
```

Néstor Batista Díaz