Roll NO - 18

Srn no- 201700110

Name- Kunal Agarwal

Experiment No - 2

Aim - Implement word count/ frequency count programs using map reduce.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Theory -

WordCount example reads text files and counts how often words occur. The input is text files and the output is text files, each line of which contains a word and the count of how often it occured, separated by a tab.

Each mapper takes a line as input and breaks it into words. It then emits a key/value pair of the word and each reducer sums the counts for each word and emits a single key/value with the word and sum.

As an optimization, the reducer is also used as a combiner on the map outputs. This reduces the amount of data sent across the network by combining each word into a single record.

Program:

Input File :

foo foo

quux labs

foo bar

quux

mapper.py

#!/usr/bin/env python3

import sys

for line in sys.stdin:

line = line.strip()

words = line.split()

for word in words:

print('%s\t%s' % (word,1))

reducer.py

#!/usr/bin/env python3

import sys

word2count = {}

for line in sys.stdin:

line = line.strip()

word , count = line.split('\t',1)

try:

count = int(count)

except ValueError:

continue

try:

word2count[word] += count

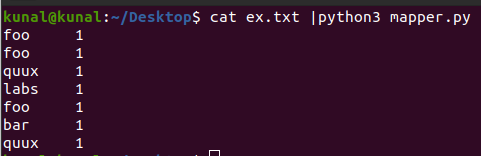
except:

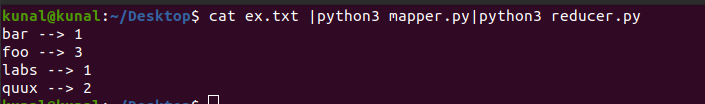
word2count[word] = count

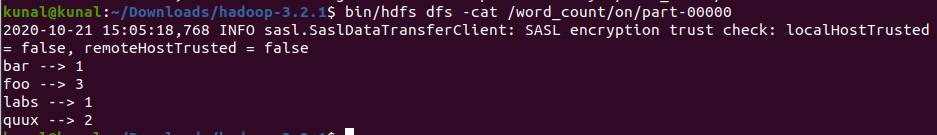
for word in sorted(word2count.keys()):

print('%s --> %s' % (word , word2count[word]))

Output







Conclusion: We have performed a count of words problem using the MapReduce algorithm in python with help of HDFS.