Predictive Analytics

Special Topics Section 011

Fall 2016

Professor Anasse Bari, Ph.D

Assignment 3

Pranit Arora

N10482611

Anand Bhave

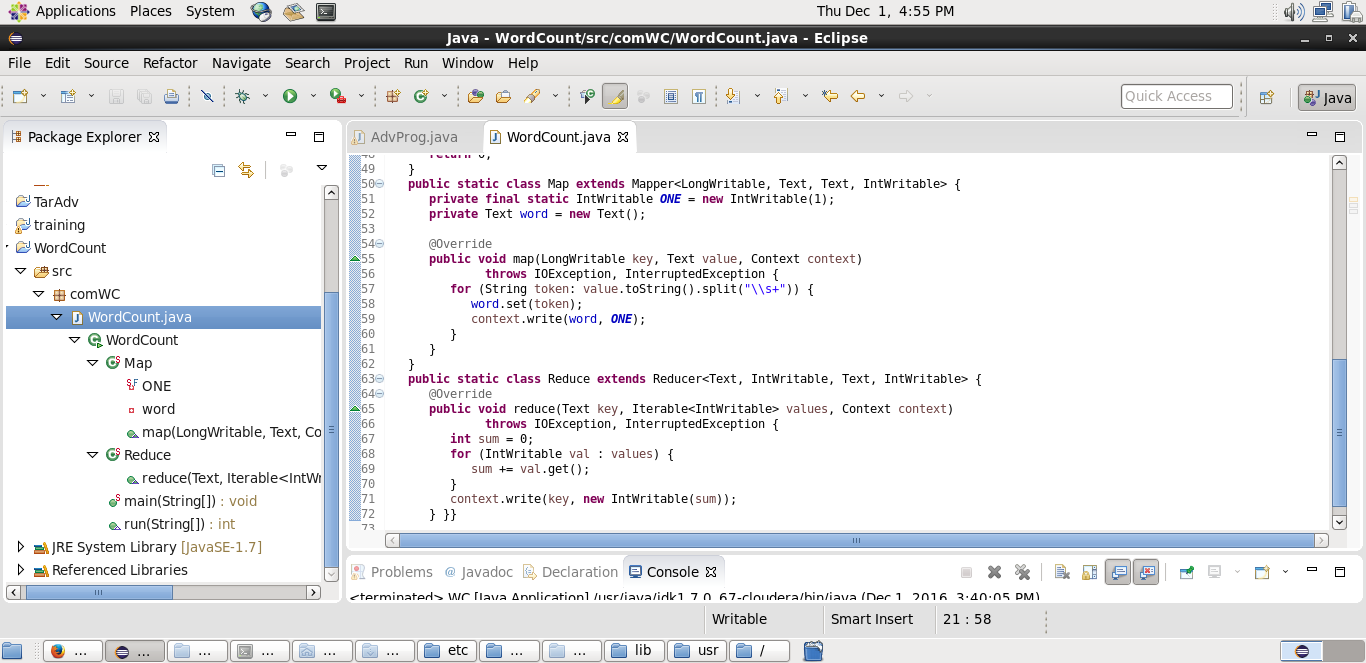
N15511185

Part 0:

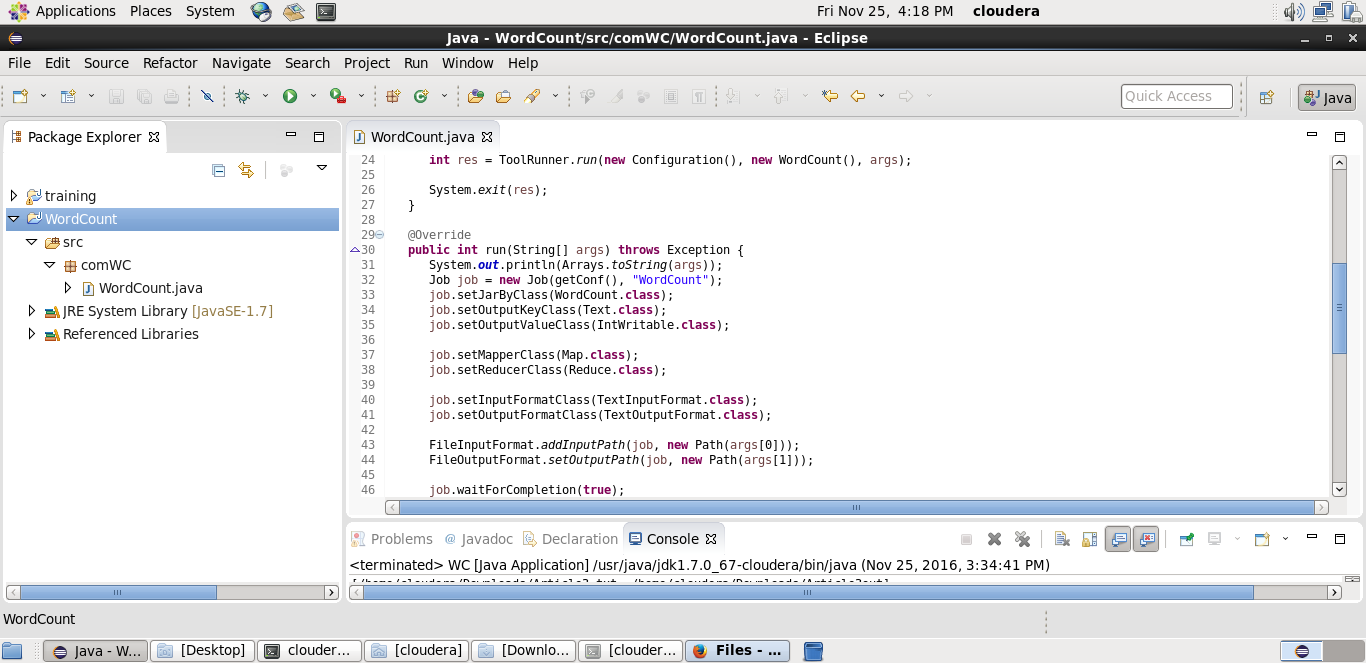
Word Count in Eclipse under Hadoop

Using eclipse and the WordCount file provided to us in the homework folder, we created the project in Eclipse as per instruction in the video tutorial.

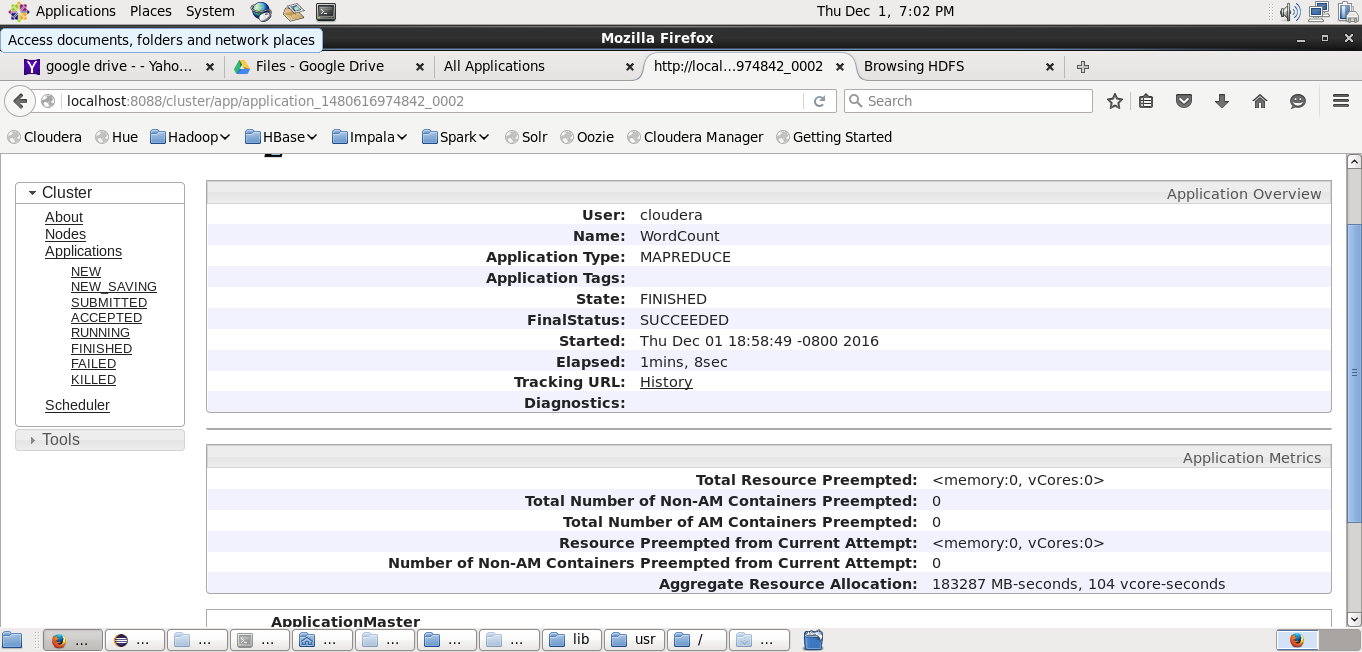
The code structure is as follows:

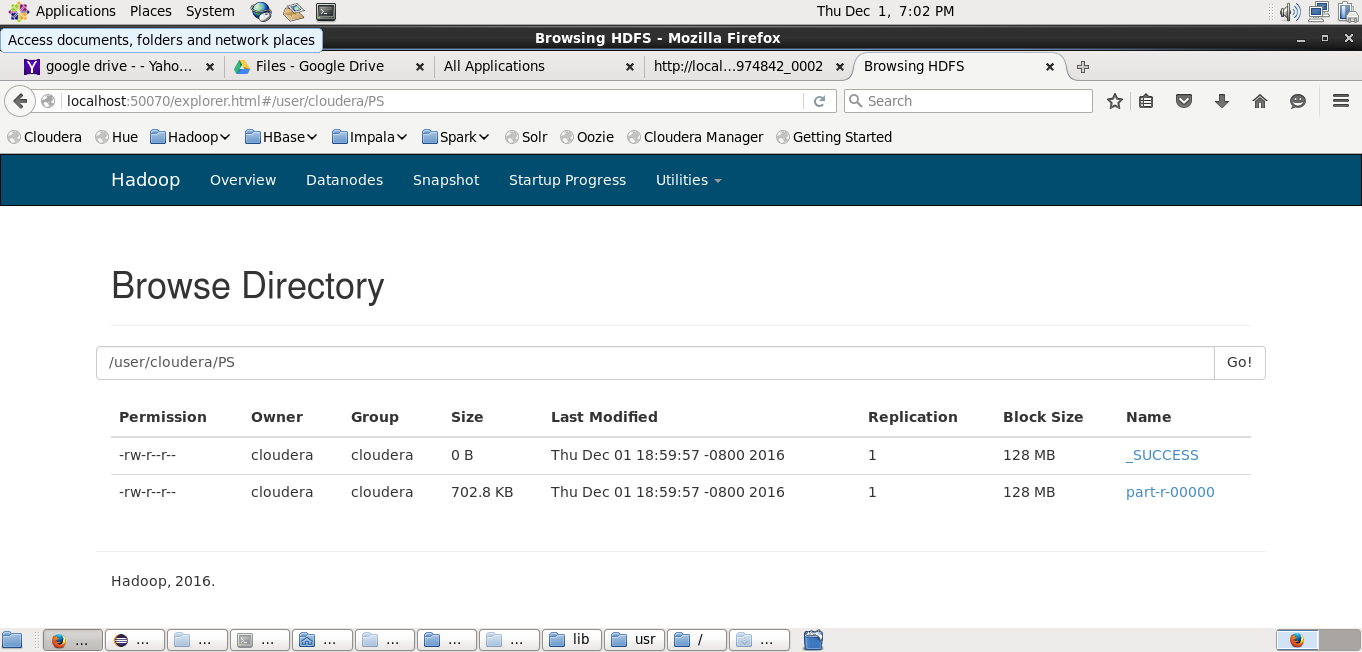


A screenshot of the code of WordCount.java in eclipse project is attached below:

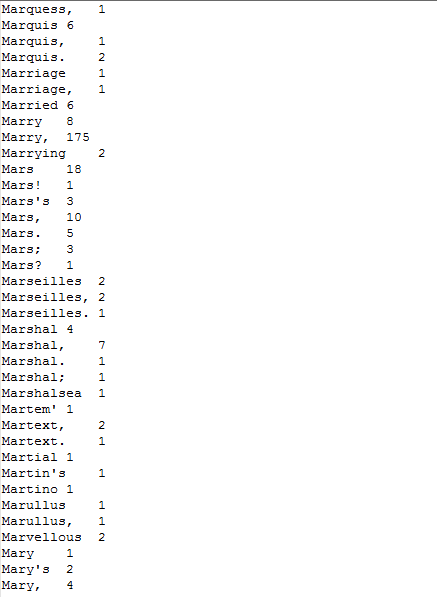


We accessed the localhost:8088 for information about the mappers and reducers.





The output of the WordCount on the file document.txt is attached below:



The output file is also available within the submission folder.

After this, we repeated the steps for Article1.txt, Article2.txt, and Article3.txt.

The output files for all are attached.

Part 1:

Targeted Advertisements using MapReduce

In this problem, we are given sales information for a business such as Amazon, who are planning to use their sales information to develop targeted advertisements. The aim is to find, for each combinations of categories, the customers with the largest purchase histories.

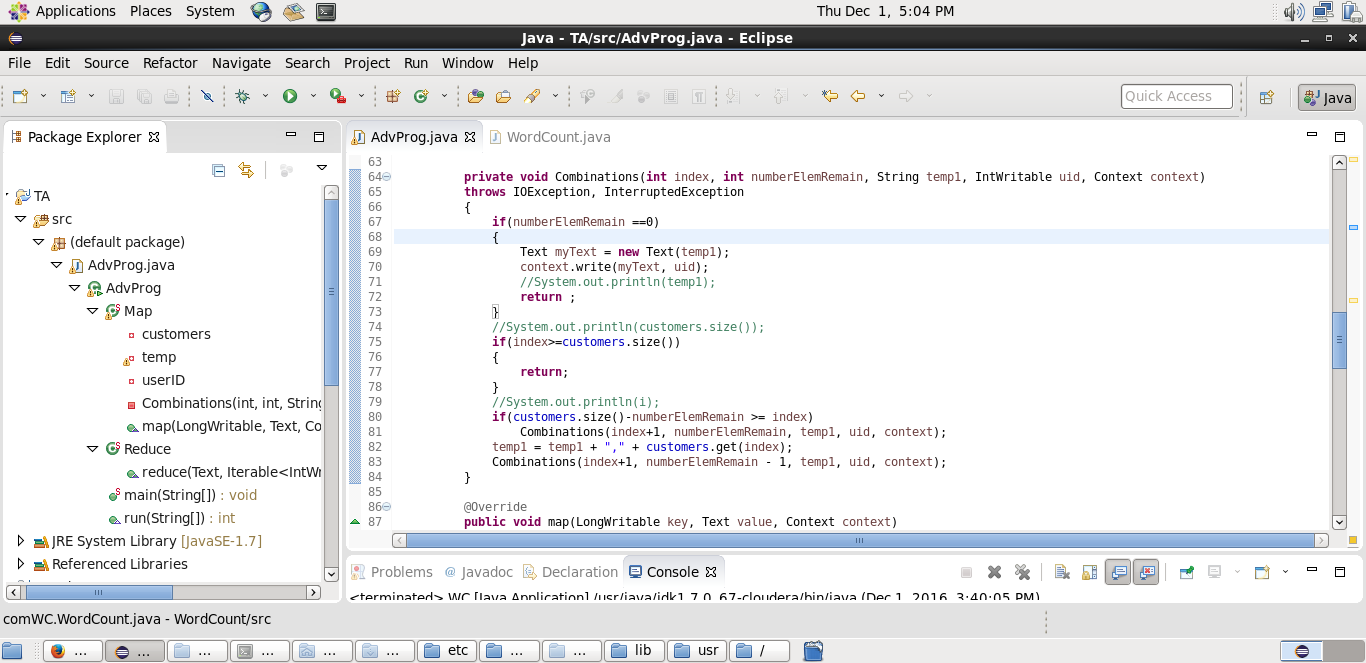
For this, we had to run a map reduce job using the Hadoop framework to figure out the target customers for advertisements.

We are given input as a file purchase\_history.txt containing lines with user id followed by a list of categories from which the user made a purchase.

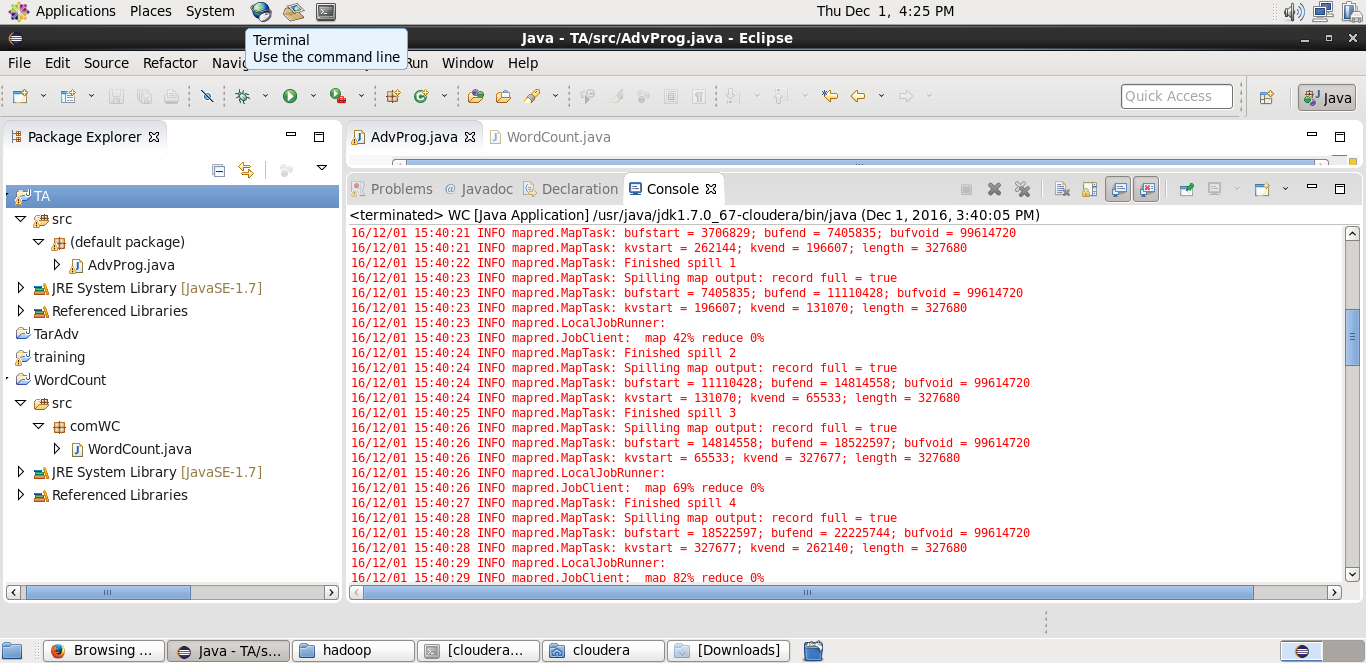
We created a mapping from each possible subset of categories to the user and added them in our mapper. This was done with the help of a function Combinations, which created each subset. Then, the reducer would read from the mapping we created and maintain a count for the number of times each user ordered a particular combination of categories. Once the list has been processed, we can write the output to the file.

The code for this part is provided in the submission folder. Some screenshots related to this problem are provided below.

A Screenshot of the code structure:



A screenshot of the execution log:



A screenshot of the generated output file:



Part 2:

Recommender System Using Apache Mahout

For this part of the assignment, we followed the instructions provided in the Mahout website through the link provided in the assignment pdf.

The code has been provided in the submission folder. Some screenshots from the program are attached below

