COMP5349 Assignment 1
Name: Hao Guo
SID: 460127213
Tutorial：Omid Tav lab 130B Thursday 4-6p.m.

Workload	Implementation	Programming Language
Category and Trending Correlation	Map Reduce	Python
Impact of Trending on View Number	Spark	Python

Workload: Category and Trending Correlation

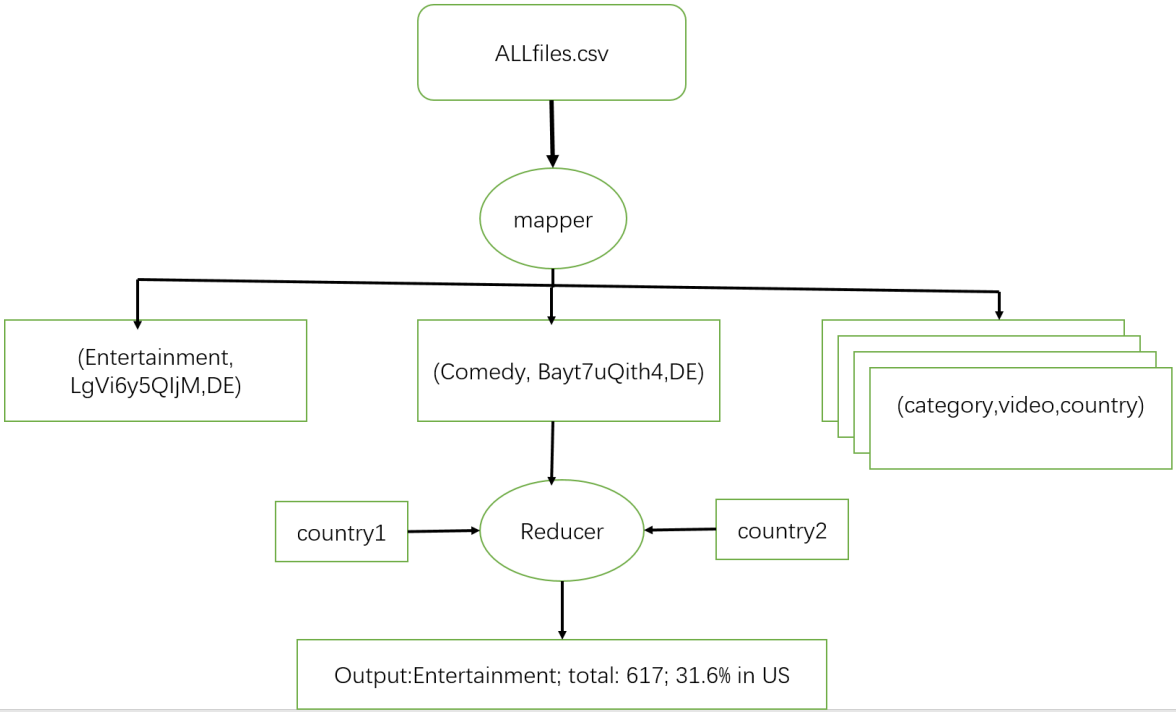

Figure 1: MapReduce phase for the workload

One MapReduce job is used for this workload. It consists of only two phases: Map then Reduce.

For each input row described above, the mapper breaks the csv file down using the python csv module and get three columns: the video ID, category name and country name. Then use the category name string as the key, set video ID and country name string as the value.

The reducer received output from mapper sorted by key, then get the key and values. In the main function of reducer, it receives two arguments, which are two countries' name. For all input lines describing the same category name, create two dictionary variables, if the key contains the first country name, append the unique value (video ID) into the first dictionary variable, Do the same thing for the second dictionary variable.

When a different category is detected, the two dictionary variables are used to calculate the amount of each category in these two countries and use a for loop, we can get the percentage of videos trending in country A that are also trending in country B.

Parallelization

Both mapper and reducer phases can run in parallel. Mappers run in parallel on different partition of the input data (ALLvideos.csv which have already uploaded into the Hadoop file system). We have set to use 2 reducers, they run in parallel on different partition of the intermediate results. Each partition handles data related with a subset of key (which is a combination of category name and the country name).

COMP5349 Assignment 1
Name: Hao Guo
SID: 460127213
Tutorial：Omid Tav lab 130B Thursday 4-6p.m.

Workload: Impact of Trending on View Number

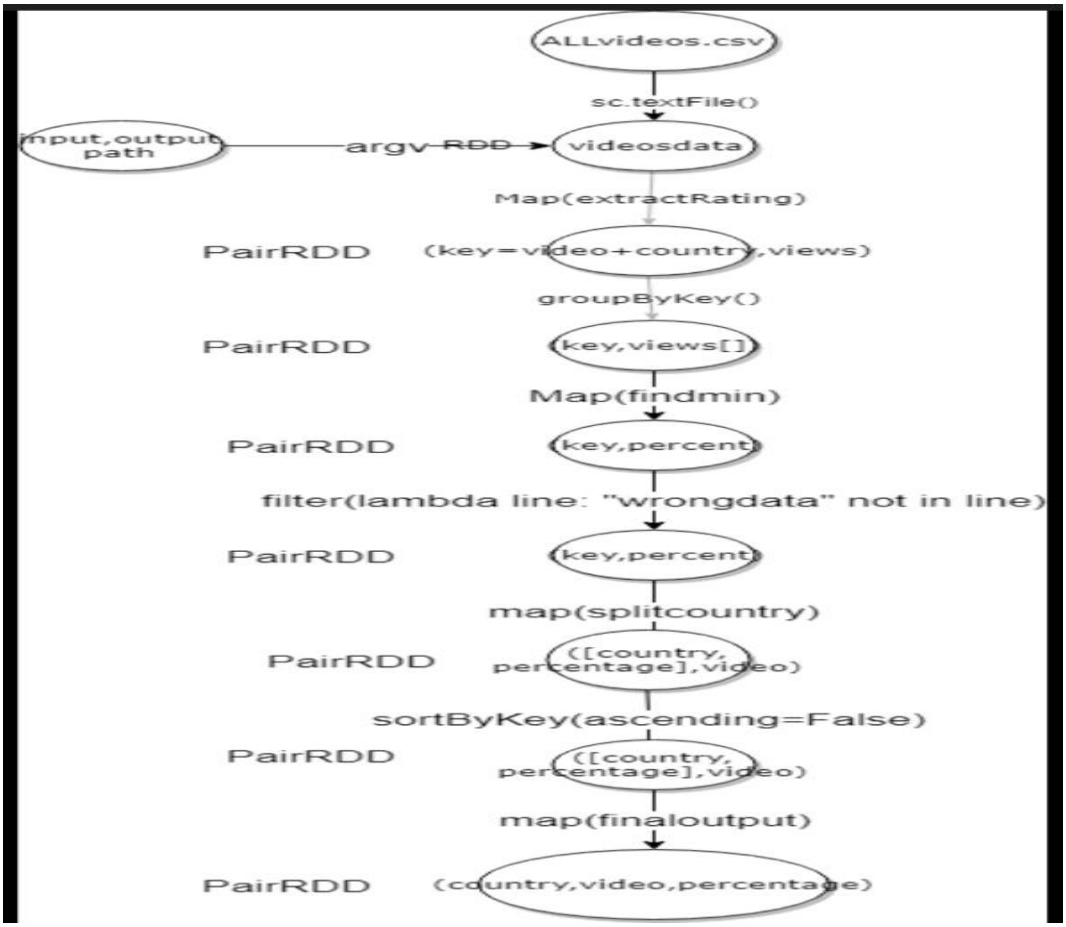

Figure 2: Spark phase for the workload.

ALLvideos.csv file is read in using sc.textFile() and mapped to create (key, views)RDD pair(key is the combination of video ID and country).

GroupByKey transformation is then applied. Another map uses to find the it second and first trending viewers. Then using the filter to delete the data that does not meet the requirements. After that, use the third map to spilt country name from the key and create a new pair [country, percentage] as the key, Finally, use sortByKey descending and the last map to output in the right format with an action to save the result as text file is called to conclude the program, output format is like this: DE; V1zTJIfGKaA, 19501.0%.

Parallelization

The map(findmin) and filter operations can run in parallel on different partitions of the key. Others have to run by order.