COMP5349 Assignment 1
Name: Ruoyu Zhuang
SID: 470132373
Tutor Name: Tony Huang

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

Workload: Category and Trending Correlation

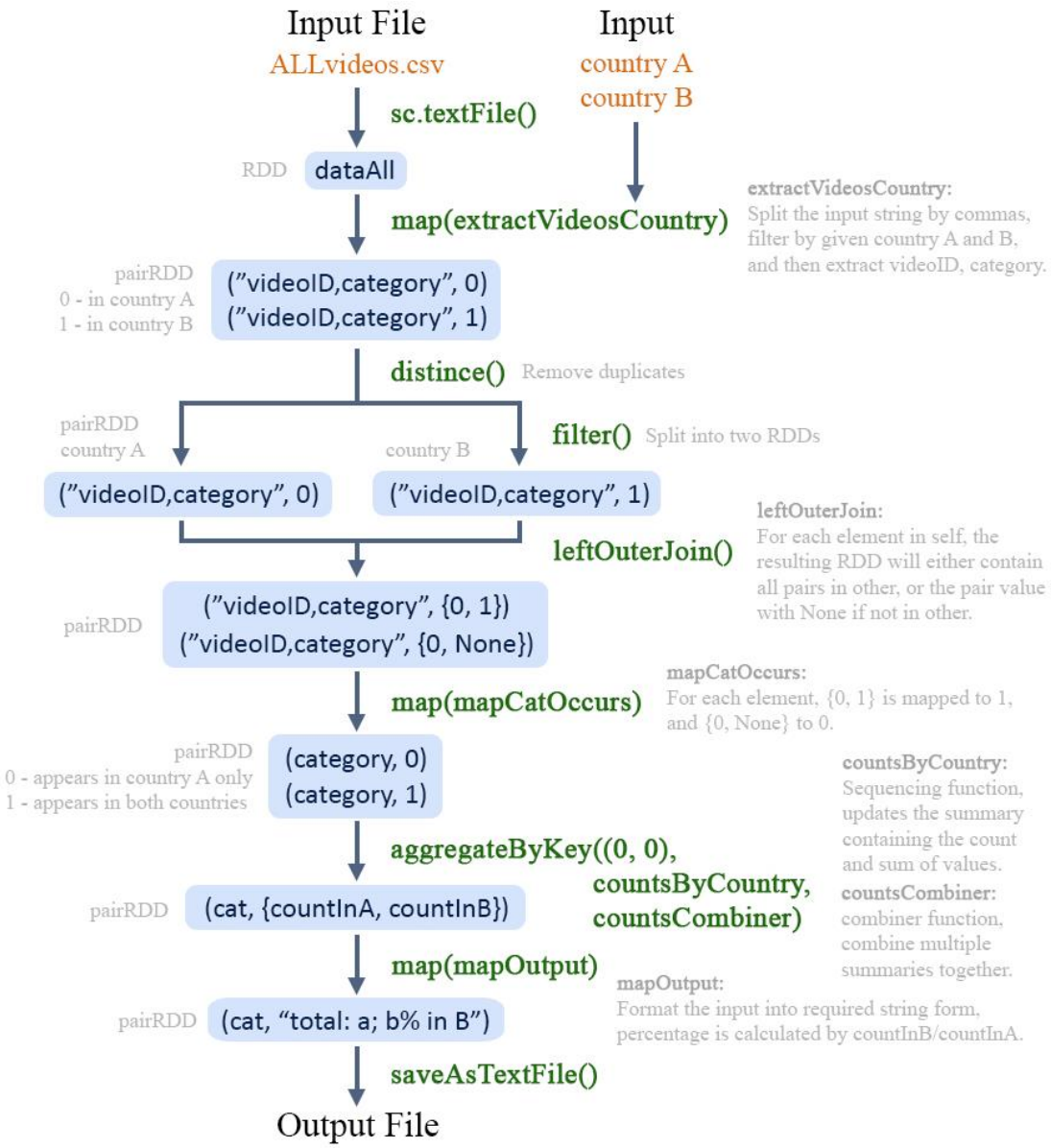

Figure 1: Spark phase for the workload.

The inputs include a csv file with all data and the given two countries, which are used later to filter and split the data into two. The duplicates are removed by distinct. Then the RDD pairs from two countries are left joined with all videos from country A and the overlapping videos from B in, following by mapping them into the form of (category, 0/1), where 0 represents occurrence in A only and 1 means it appears in both countries.

Afterwards, AggregateByKey is applied with countsByCountry and countsCombiner as sequencing and combiner functions respectively, which calculate the number of videos in country A and B for each category by count and sum operations. Finally, the results are formatted as required by another map and then output as a text file.

Parallelization

The data are read in then parallelized using the built-in SparkContent parallelize method. The distinct and filter operations can run in parallel on partitions. The join operation can also be parallelized with shuffling and pipelined with map. The aggregateByKey transformation and the following map can run in parallel on partitions, processing a few categories' data. There is also parallelization in the saveAsTextFile operation when output files.

Workload: Impact of Trending on View Number

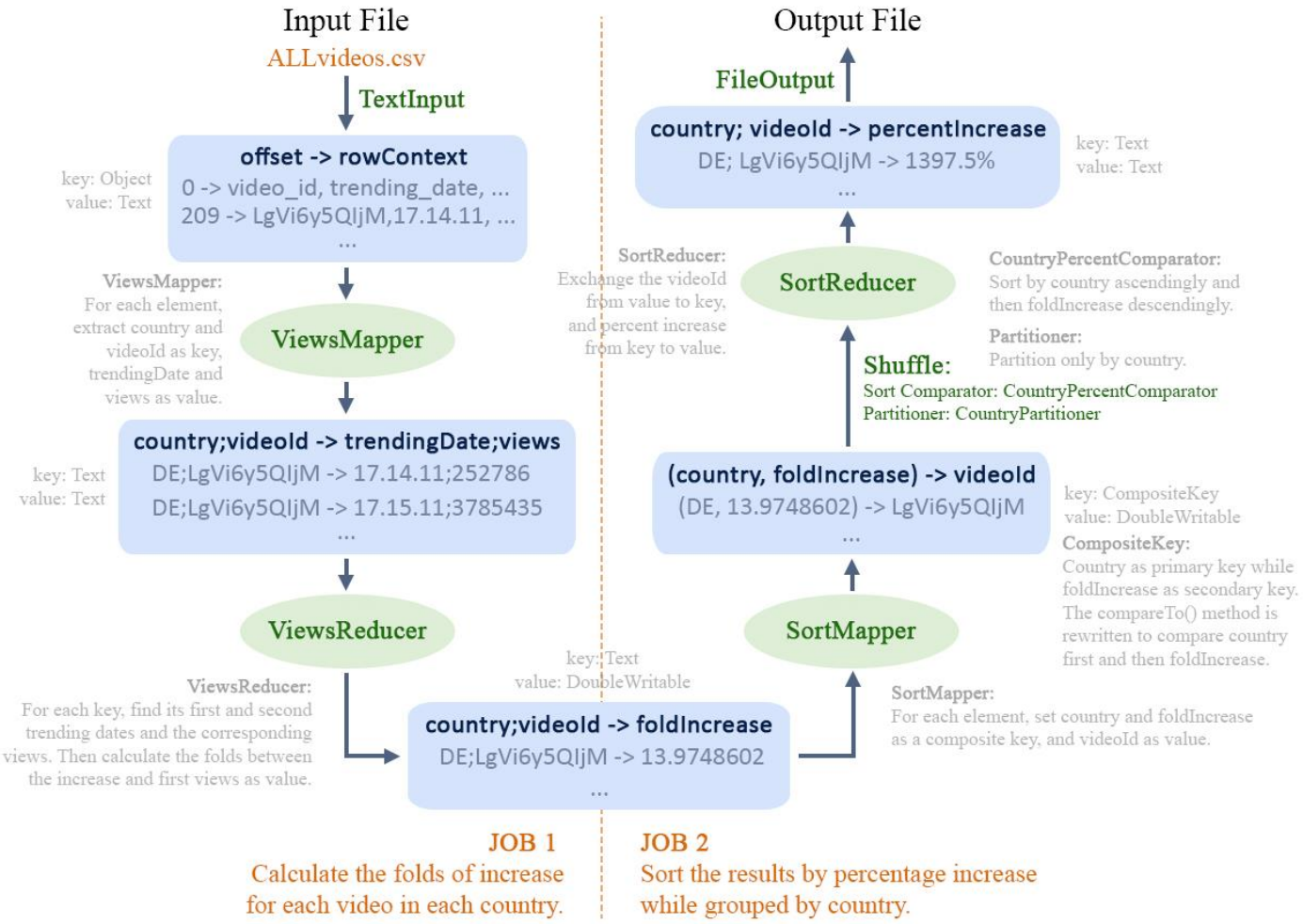

Figure 2: MapReduce phase for the workload.

Two MapReduce jobs are implemented orderly for this workload. **The first job** is to calculate the folds of increase for each video in each country, which contains a map phase extracting country, video id, trending date and views number for each record, and a reduce phase finding the first and second views by trending date as well as calculating the increasing folds. **The second job** is to sort the results by increase within the groups of country. The map phase puts the increase into the composite key, which is used when sorting in the shuffle phase. Furthermore, the customized partitioner by country only is defined to maintain the groups of country. Then the reducer exchanges the key and value for the required format as final output.

Parallelization

All mappers and reducers phases can run in parallel. Mappers run in parallel on split inputs. Two reducers are set for both jobs, running in parallel on different partitions of the intermediate results. Each partition handles data related to a subset of countries in the 2nd job.