**CS5590 Big Data Programming – Lab**

**Assignment 1**

**1. Finding Facebook mutual friends using Map**

**Reduce.**

**2. Find the average, sum and count from log file using Map Reduce with Combiner Class implementation.**

**Team Structure**

**Team Id: 03**

**Member 1: Raju Nekadi**

**Class Id: 07**

**Member 2: Shushma Manne**

**Class Id: 03**

**Source Code:**

https://github.com/rnekadi/CSEE5590\_BIGDATA\_PROGAMMING\_Fall2018/tree/master/Lab1/Source

**Video/Demo:**

https://youtu.be/bPZI2KxzMLY

**Introduction**

This lab assignment deals with understanding the concepts of Hadoop -

Map Reduce and implementing a map reduce algorithm to solve below two problems.

1. Finding common friends’ from user’s facebook list.
2. Finding sum, count and average from log file for given word and value pair in record.

**Objective**

**1.** Implement Map Reduce algorithm for finding Facebook common

Friend’s problem and run the Map Reduce job on Apache Hadoop.

Show your implementation through map-reduce diagram.

**Approach**

Let us take the use case example given in problem statement to implement Map Reduce approach.

Given input is the user and friends list User --> [friends List]

**Raju Swati, Tanu, Sam**

**Swati Raju, Tanu, Sam, Golu**

**Tanu Raju, Swati, Sam, Golu**

**Sam Raju, Swati, Tanu, Golu**

**Golu Swati, Tanu, Sam**

The Map Reduce approach will divide this into different phases as described below.

**Input Split Phase:** First all Input Splits created from all record one by one.

Raju Swati, Tanu, Sam SameSam

Map Phase: Using above input Split Mapper will create key, value pair.

For map (Raju Swati, Tanu, Sam)

(Raju Swati) -> Swati, Tanu, Sam

(Raju Tanu) -> Swati, Tanu, Sam

(Raju Sam) -> Swati, Tanu, Sam

For map (Swati Raju, Tanu, Sam, Golu)

(Raju Swati) -> Raju, Tanu, Sam, Golu

(Swati Tanu) -> Raju, Tanu, Sam, Golu

(Swati Sam) -> Raju, Tanu, Sam, Golu

(Golu Swati) -> Raju, Tanu, Sam, Golu

And so on...

**Combiner/Shuffle Sort Phase:** This phase will group them by their keys before sending it to reducer:

(Raju Swati) -> (Raju Tanu Sam Golu) (Swati Tanu Sam)

(Raju Tanu) -> (Raju Swati Sam Golu) (Swati Tanu Sam)

(Raju Sam) -> (Raju Swati Tanu Golu) (Swati Tanu Sam)

(Swati Tanu) -> (Raju Swati Sam Golu) (Raju Tanu Sam Golu)

(Swati Sam) -> (Raju Swati Tanu Golu) (Raju Tanu Sam Golu)

(Swati Golu) -> (Raju Tanu Sam Golu) (Swati Tanu Sam)

(Tanu Sam) -> (Raju Swati Tanu Golu) (Raju Swati Sam Golu)

(Tanu Golu) -> (Raju Swati Sam Golu) (Swati Tanu Sam)

(Sam Golu) -> (Raju Swati Tanu Golu) (Swati Tanu Sam)

Reducer Phase: The reducer will take all the grouped key and output the same key with the result of the intersection.

For example, reduce ((Raju Swati) -> (Raju Tanu Sam Golu) (Swati Tanu Sam)) will output (Raju Swati): (Tanu Sam) and means that friends A and B have C and D as common friends.

The result after reduction

(Raju Swati) -> (Tanu Sam)

(Raju Tanu) -> (Swati Sam)

(Raju Sam) -> (Swati Tanu)

(Swati Tanu) -> (Raju Sam Golu)

(Swati Sam) -> (Raju Tanu Golu)

(Swati Golu) -> (Tanu Sam)

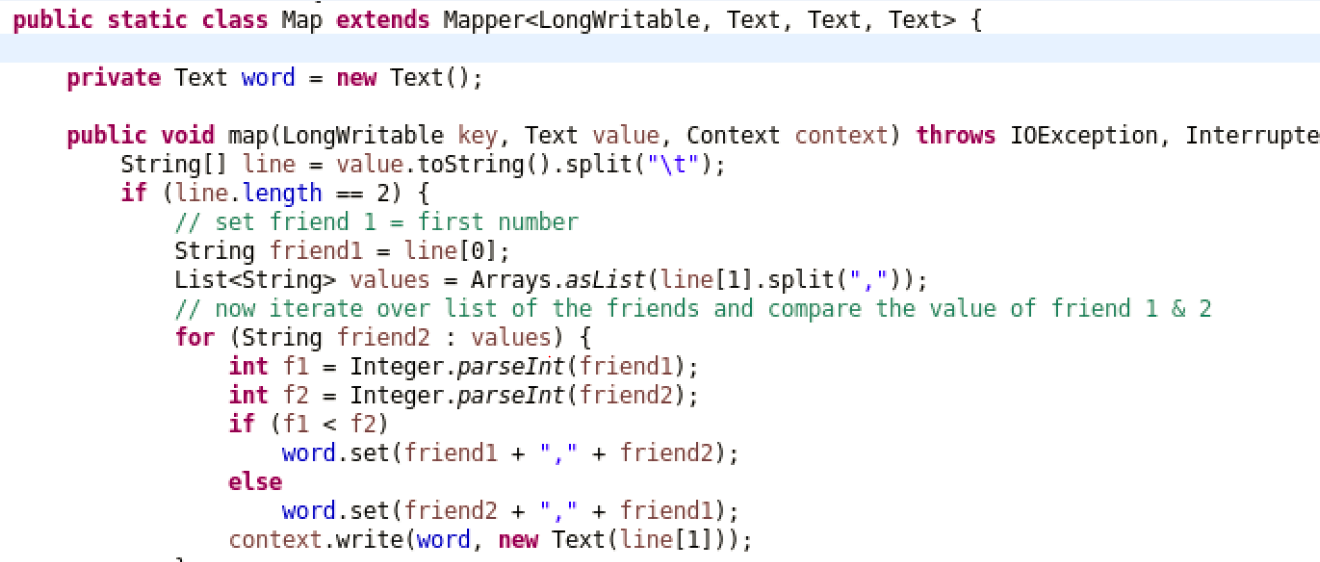
(Tanu Sam) -> (Raju Swati Golu)

(Tanu Golu) -> (Swati Sam)

(Sam Golu) -> (Swati Tanu)

**Workflow**

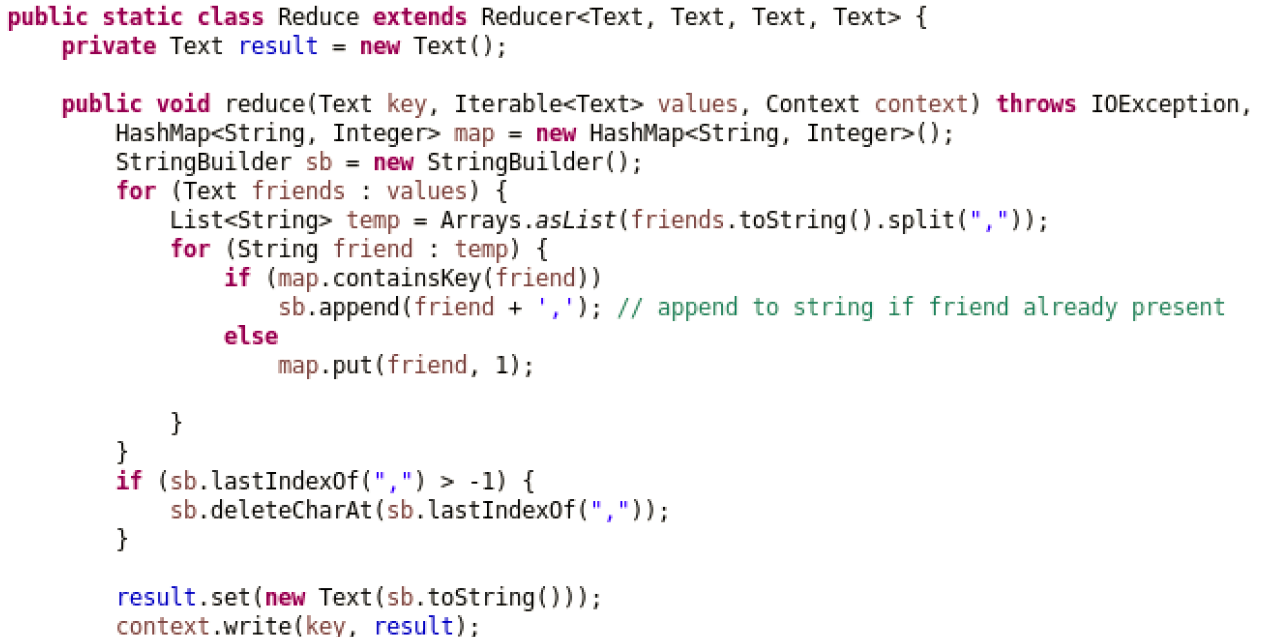
Create a mapper class as shown in the code snippet below. Each line of the input file split based on “tab” delimiter. Then its length computed as two, where the first part is source or base user and the rest of the split considered as list of friends of the user. Then the keys are prepared as (A, B) or (B, A) based on the integer values of A & B in the input.



Create a reducer class where the data grouped based on the key values

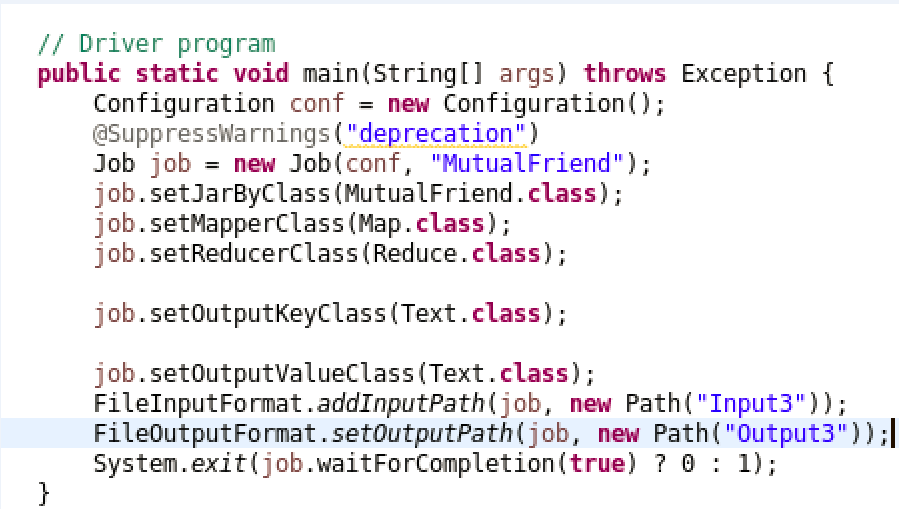
(A, B) or (B, C) and their list of friends as produced. Then finally reduced to

Find the mutual friends of (A, B).



A main method which acts as a driver to set mapper and reducer class

Which takes the input and produces the output.



**Data set and Parameter**

Please find input file and Output File at

https://github.com/rnekadi/CSEE5590\_BIGDATA\_PROGAMMING\_Fall2018/tree/master/Lab1/Source/FaceBookMutual/Input3

**Evaluation**

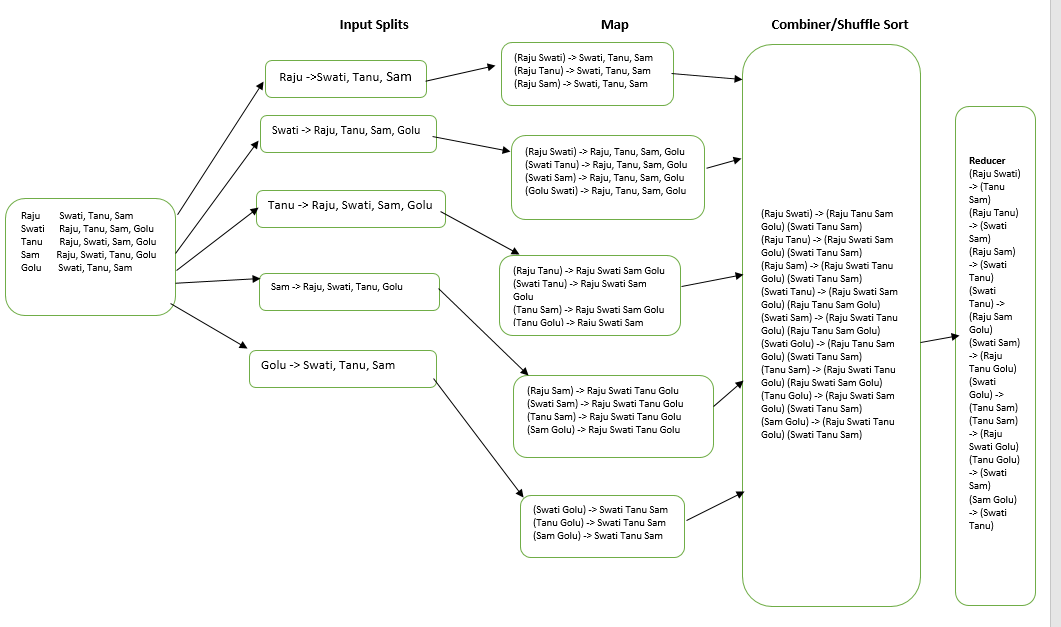
Hadoop map reduce is very efficient in finding the common/mutual friends

Of two users when their list of friends are grouped together and filtered

Using reduce operation

**Conclusion**

Representing the mutual friends problem using the map reduce diagram.



**Objective**

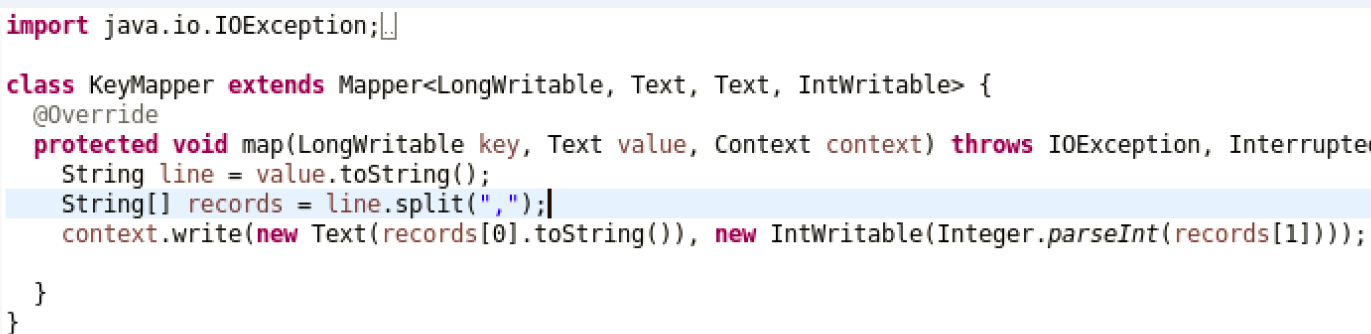
**2.** Finding sum, count and average from log file for given word and value pair in record.

**Approach**

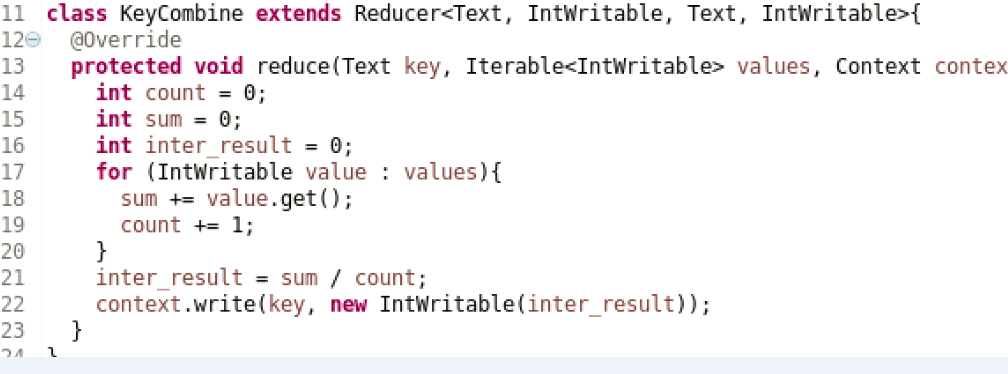
In order to find the average of each key given in input file the logic is quite simple: If all the number have the **same key**, then the mapper sent all the values you want to find the average of with that same key. Because of this, in the combiner (mini reducer) and reducer you can sum the values in the iterator. You can then keep a **counter** on number time the iterator works, which solves the issue of how many items, are to get average. Finally, after the iterator, you can find the average by dividing the sum by the number of items.

**Workflow**

Create a mapper class as shown in the code snippet below. Each line of the input file split based on “,” delimiter where the first part is key (Node) and second part is numeric value of response time. Then in map the keys are prepared as (N1 10, 20) or (N2 20, 20) (N3 30, 30).



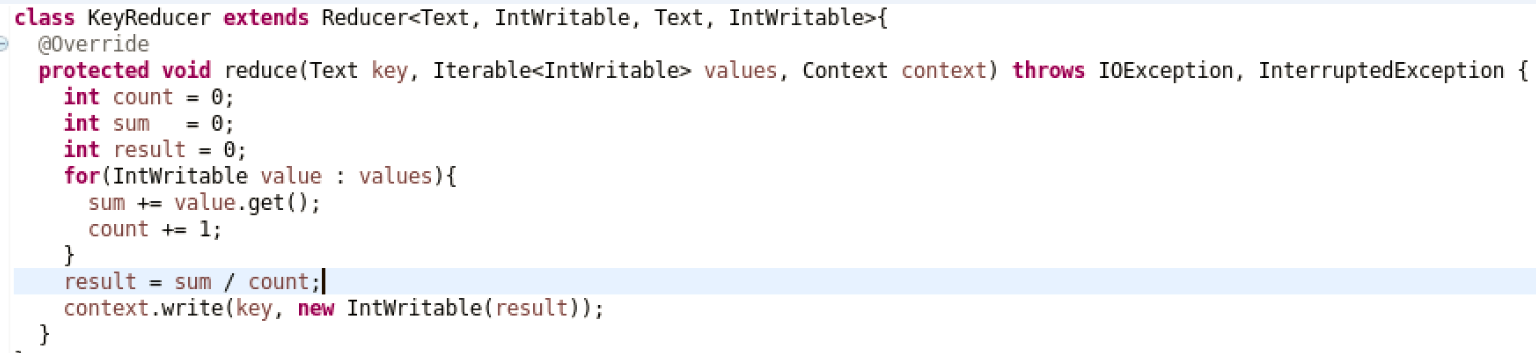
Create a Combiner also called mini reducer average, count, sum calculated based on the key values. The combiner step will decreases the amount of data that need to be processed by the reducer. Combiner improves the overall performance of the reducer.



The reducer will need to process only few of key value pair and generate output.

A main method, which acts as a driver to set mapper, combiner and reducer class

Which takes the input and produces the output.



**Data set and Parameter**

Please find input file at below location

https://github.com/rnekadi/CSEE5590\_BIGDATA\_PROGAMMING\_Fall2018/tree/master/Lab1/Source/AverageCount/Input

**Evaluation**

Hadoop map reduce with combiner is very efficient in finding the average of key given in input file. We can say that Map Reduce Combiner plays a key role in reducing network congestion. Map Reduce combiner improves the overall performance of the reducer by summarizing the output of Mapper.

**Conclusion**

Representing the Average Count Problem using the map reduce diagram.

