**Assignment 3**

**Problem Statement:**

Use Movies Dataset. Write the map and reduce methods to determine the average ratings of movies. The input consists of a series of lines, each containing a movie number, user number, rating, and a timestamp: The map should emit movie number and list of rating, and reduce should return for each movie number a list of average rating.

**Objectives:**

The objective of the assignment is to implement mapReduce function on the unstructured data set like movie dataset and gain insights from the unstructured data.

**Theory:**

Mapper and Reducer-

* MapReduce is Distributed Programming Model.
* MapReduce can be easily parallelized across multiple servers.
* It splits the problem and send chunks to different m/c for processing.
* When all m/c finish its part of calculation , the o/p is combined and aggregated for final solution.
* MapReduce is implemented in multiple programming languages.
* Map: Process the chunk of input, emits key value pair.
* Intermediate Phase[Shuffle] : Combines similar key values.
* Reduce: collect intermediate key and values, produce aggregated result.

Programming steps:

Map : Works on each document, emit key value pair.

Intermediate step : Shuffling step, All similar keys are grouped and combine value with key is emitted.

Reduce : Takes the list of values, and combine intermediate results into Final one .

**Code:**

>mongo

> show dbs

admin 0.000GB

config 0.000GB

local 0.000GB

mapr 0.000GB

> use mapr

switched to db mapr

> show collections

Movie

> db.movie.find().pretty()

{

"\_id" : ObjectId("5b44f362e461378dad2cf8d0"),

"User" : "Sunil",

"Movie" : "DDLJ",

"Rating" : 4

}

var map=function(){emit(this.Movie,this.Rating}}

var reduce=function(key,val)

{ var t=0;

for (var i=0;i<val.length;i++){

t=t+val[i];

}

return t/val.length;

}

> var res=db.movie.mapReduce(map,reduce,{out:"Average"});

> db.res.find();

> db.Average.find();

{ "\_id" : "DDLJ", "value" : 4.333333333333333 }

{ "\_id" : “TITANIC", "value" : 5 }

{ "\_id" : "INFINITY WAR", "value" : 5 }

{ "\_id" : "KASAM", "value" : 1.5 }

{ "\_id" : “HAHK", "value" : 2 }

{ "\_id" : "THE GHOAST", "value" : 3 }

**Output:**

{ "\_id" : "DDLJ", "value" : 4.333333333333333 }

{ "\_id" : “TITANIC", "value" : 5 }

{ "\_id" : "INFINITY WAR", "value" : 5 }

{ "\_id" : "KASAM", "value" : 1.5 }

{ "\_id" : “HAHK", "value" : 2 }

{ "\_id" : "THE GHOAST", "value" : 3 }

**Conclusion:**

The unstructured dataset used gave the average movie rating by implementing mapReduce function in mongoDB.