1. **Pair in Mapper**

Class Mapper

method INITIALIZE

H🡨 new Associative array

Mc🡨 new Associative array // for \* element array

Method Map(docid , rec d)

For w 1 to d

For u to neighbor(w)

// H{pair(u,w) }🡨 H{pair(u,w)}+1 // add 1 each time;

Mc{pair(w,\*)}🡨 Mc{pair(w,\*)}+1; //count all w element;

Method Close

For all pair p in Mc

Emit (p,Value)

// For all pair p in H

// Emit( p , H{p} value ) //emitting hash map value of each pear ;

Class Reduce

Method Reduce (pair p , counts[ c0,c1..])

S 🡨 0

For 1 to all count //getting all values of each par from different mappers and sum;

S+=c;

Emit (pair P , S sum) ; emitting each pair with final sum;

**2. Striper**

Class Mapper

Method Map()

For all w in d

H🡨 new Associative array

For all u to Neighbor(w)

H{u}=H{u}+1; //store all u elements

Emit (w,H)

Class Reduce

Method reduce (w , H…)

M🡨 new Associative array //

For all H

Sum (Hj,H) //sum of all incoming H

//now we can sum all elements inside Hj for margin

Margin 0;

For all in Hj

Margin +=Hj(u)

//frequency F for special element in Hj

F=Hj(u)

Emit (w,Hj)

4 a)

|  |  |
| --- | --- |
| **Mapper 1 Record 1** | **Mapper 2 Record 1** |
| <(cherry, mango), 1>  <(cherry, olive), 1>  <(mango, olive), 1>  <(mango, cherry), 1>  <(olive, cherry), 1> | <(mango, cherry), 1>  <(mango, banana), 1>  <(mango, olive), 1>  <(cherry, banana), 1>  <(cherry, olive), 1>  <(banana, olive), 1> |
| **Mapper 1 Record 2** | **Mapper 2 Record 2** |
| <(mango, olive), 1>  <(mango, banana), 1>  <(mango, cherry), 1>  <(olive, banana), 1>  <(olive, cherry), 1>  <(banana, cherry), 1> | <(cherry,  mango), 1>  <(cherry, olive), 1>  <(cherry, banana), 1>  <(mango,  olive), 1>  <(mango, banana), 1>  <(olive, banana) , 1> |
| **Mapper 1 Record3** | **Mapper 2 Record 3** |
| <(cherry, banana), 1>  <(cherry,  mango), 1>  <(cherry, banana), 1>  <(banana, mango), 1>  <(mango, banana), 1> | <(mango, olive), 1>  <(mango,  banana), 1>  <(mango,  olive), 1>  <(olive,  banana), 1>  <(banana, olive), 1> |
| **Mapper 1 Reducer1 output** | **Mapper 2 Reducer1 input** |
| <(banana, cherry), 1>  <(banana, mango), 1>  <(banana, olive), 2>  <(cherry, banana), 4>  <(cherry, mango), 3>  <(cherry, olive), 3>  < banana ,\*> margin =3  <(cherry,\*> margin =3  Example <(banana, olive), 2> frequency = 2/3 | <(mango,  banana), 5>  <(mango, cherry), 3>  <(mango, olive), 6>  <(olive, banana), 3>  <(olive, cherry), 2>  Margin (<(mango,\* > ) =3  Margin(<(olive , \* >) =2  Freq (<(olive, banana) >) =3/3 |

4 c)

|  |  |
| --- | --- |
| Mapper 1 Rec 1 | MApper 2 Rec 1 |
| <cherry, {(mango, 1), (olive, 1)}>  <mango  , {(olive, 1), (cherry, 1)}>  <olive  , {(cherry, 1)}>  <cherry  , {}> | <mango  , {(cherry, 1), (banana, 1), (olive,  1)}>  <cherry  , {(banana, 1), (olive, 1)}>  <banana  , {(olive, 1)}>  <olive  , {}> |
| MApper 1 Rec2 | Mapper2 Rec 2 |
| <mango  , {(olive, 1), (banana, 1), (cherry, 1)})  <olive, {(banana, 1), (cherry, 1)})  <banana, {(cherry, 1)})  <cherry, {}> | <cherry, {(mango, 1), (olive, 1),  (banana, 1)}>  <  mango, {(olive, 1), (banana, 1)}>  <olive, {(banana, 1)}>  <banana {}> |
| Mapper 1 rec3 | Mapper 2 Rec 3 |
| <cherry,  {(banana, 2), (mango, 1)}>  <banana, {(mango, 1)}>  <mango, {(banana, 1)}>  <banana, {}> | <mango, {(olive, 2), (banana, 1)}>  <olive, {(banana, 1)}>  <banana, {(olive, 1)}>  <olive, {}> |
| Red output | Rec output |
| <banana, {(cherry, 1), (mango, 1), (olive, 2)}>  <cherry, {(mango, 3), (olive, 3), (banana, 4)}>  Margin (banana) = 4  Margin (cherry)= 10  Freq ( cherry , mongo)= 3/10 | <mango  , {(olive, 6), (cherry, 3), (banana, 5)}>  <olive, {(cherry, 2), (banana, 3)}>  Margin (mongo) =14  Magin(olive)=5  Freq (mongo ,cherry) = 3/14 |