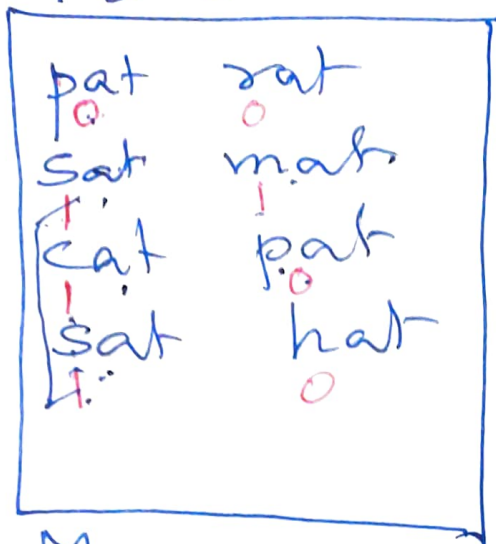
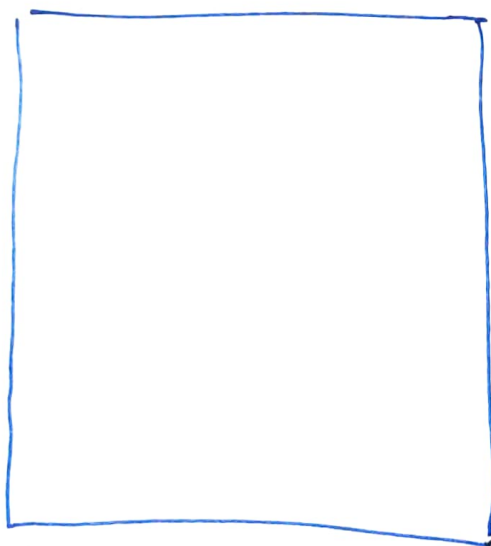


1-S-0



M0

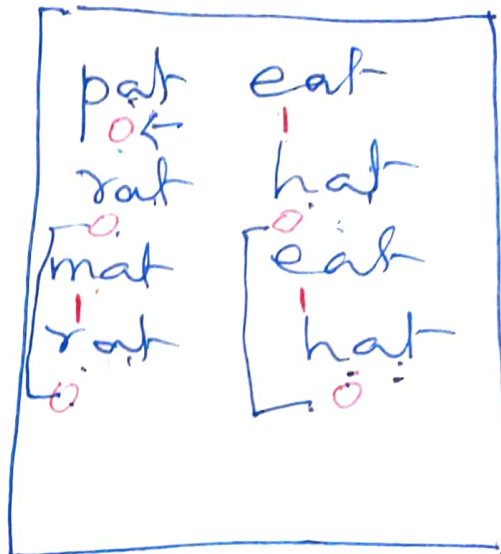


R0

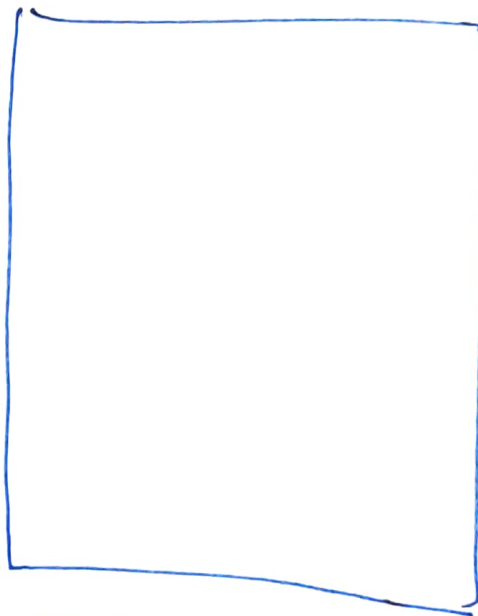
DN0

$$\text{key}[0] \% 2$$

1-S-1



M1



R1

DN1

(a) How many S-a.N. wordcount

(b) In this version of WC,

Klond Count Problem

Without local Aggregation - 9 Tokens transferred over the network.

With local Aggregation - 6 Tokens

4	0 → 1
5	1 → 0

9

3

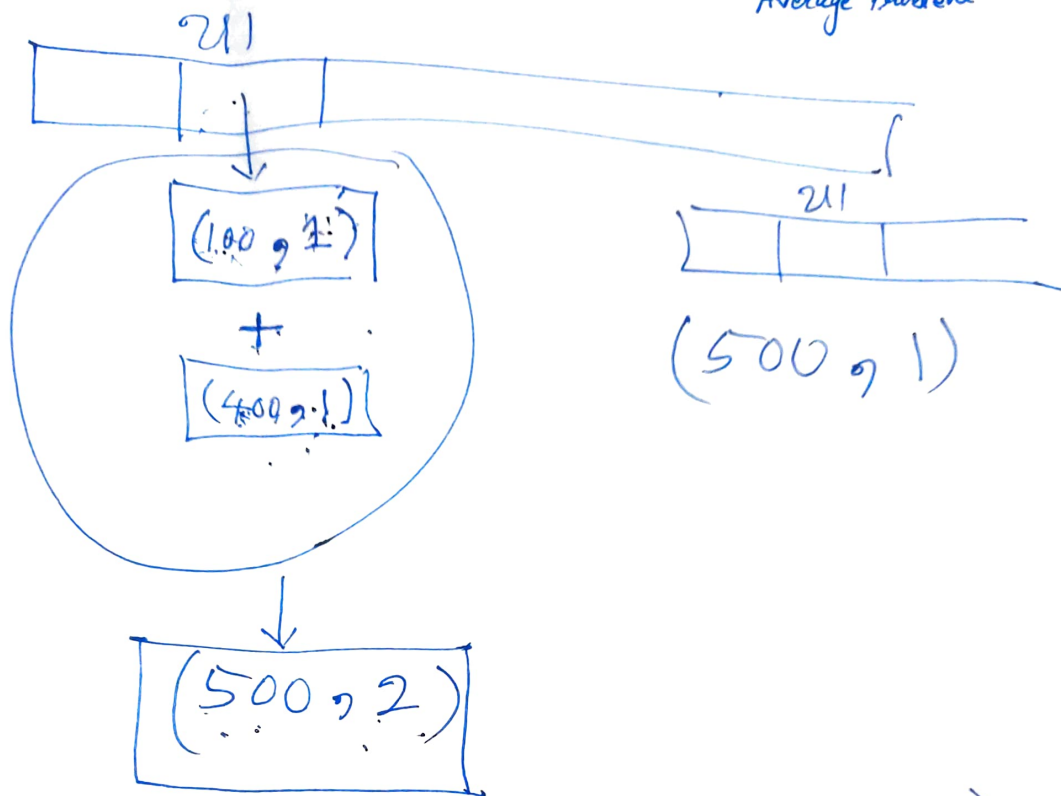
0 → 1

3

1 → 0

6

Average Problems



$$(211, [(500, 2), (500, 1)])$$

$$(211, 1000/3)$$

Explanation of Average Problem

u_1	100
u_2	200
u_3	300
u_1	400

u_1	500
u_2	600
u_3	700
u_2	800
u_3	

$$u_1 - 100 + 400 + 500 = 1000$$
$$(u_1, 1000/3)$$

$$u_2 \quad 200 + 600 + 800 \quad (u_2, 1600/3)$$

$$u_3 \quad 300 + 700 \quad (u_3, 1000/2)$$

S-S

$$(u_1, [100, 400, 500].)$$

$$(u_2, [200, 600, 800])$$

$$(u_3, [300, 700])$$

Average Problem without
using In-Mapper.

class Mapper

method map(——)

$u = r.getUserID()$

$t = r.getTime()$

Emit(u, t)

class Reducer

method Reduce($u, [t_1, t_2, \dots]$)

Sum = 0

count = 0

for all t in $[t_1, \dots]$ do

Sum += t

count++

Emit($u, \text{Sum/count}$)

class Mapper

```
method to initialize()  
    H = new Assoc.Array()
```

method map(——)

u = r.getUserID()

t = r.getTime()

if H{u} is nil

H{u} = new Pair(t, 1)

else

H{u} += new Pair(t, 1)

// element wise addition

method close

for all term u in H do

Emit(u, H{u})