

1) Mapper

Initialize() {

 w = dict()
}

map(url, visitas, fecha) {

 if not url in w:

 w[url] = (visitas, fecha)

 return

 if w[url][visitas] < visitas:

 w[url] = (visitas, fecha)

}

close() {

 for ~~k~~ k in w:

 emit(k, (w[k][visitas], w[k][fecha]))

}

Reducer

```
reduce(url, [visitas; fecha]) {
```

```
    d = dict([visitas, fecha])
```

```
    max = 0
```

```
    for v in d:
```

```
        if v > max:
```

```
            max = v
```

```
    emit(url, d[v], v)
```

```
}
```

3) a - como $\frac{900}{1300} > 0.5$ se deben guardar no

ocurrencias con $p = 1 - \frac{900}{1300} = 0,308$

el b óptimo es $\left\lceil \frac{\log(2 - 0,308)}{-\log(1 - 0,308)} \right\rceil = \lceil 1,43 \rceil = 2 \neq 1$
 $\Rightarrow$ Falso

d) Falso, depende también de p que depende de la cantidad de punteros.

Por ejemplo, con 10 términos y 5 documentos y $p = 0,3$ (a con $p \in (0,38, 0,5)$ unario es óptimo)

$0,3 = \frac{x}{10 \cdot 5} \Rightarrow$ con 15 documentos unario no es óptimo

4) $M = 1$ $S = "ABABAA"$

$A = 1$

$B = 1$

$Eof = 1$

Siempre (solo exclusion).

Leo	M0	M1	M2	emito
A	$A = 1$ $B = 1$			$esc(1) A(1/3)$
B	$A = 1$ $B = 1$ $esc = 2$	$(A) B = 1$		$esc(1) esc(1/2) B(1/2)$
A	$A = 2$ $B = 1$ $esc = 2$	$(A) B = 1$ $(B) A = 1$	$(AB) A = 1$	$esc(1) esc(1) A(1/4)$
B	$A = 2$ $B = 1$ $esc = 2$	$(A) B = 2$ $(B) A = 1$	$(BA) B = 1$ $(AB) A = 1$	$esc(1) B(1/2)$
A	$A = 2$ $B = 1$ $esc = 2$	$(A) B = 2$ $(B) A = 1$	$(AB) A = 2$ $(BA) B = 1$	$A(1/2)$
A	$A = 3$ $B = 1$ $esc = 2$	$(A) B = 2$ $(A) A = 1$ $(A) esc = 2$ $(B) A = 1$	$(AB) A = 2$ $(BA) B = 1$ $(BA) A = 1$	$esc(1/2) esc(1/3) A(1/2)$
Eof				$esc(1) esc(2/3) esc(1)$ $eof(1)$

Los modelos se actualizan después de comprimir.
Cada char se comprime con un modelo de la
fila anterior.

$$5) (0,3; 0,5; p_1, p_2)$$

$$p_1 > 0,9$$

$$p_2 < 0,3$$

~~Pruebo con 8 minhashes, 2 AND y 4 OR~~

$$p_1 = 1 - (1 - 0,3^2)^4 = 0,932$$

$$p_2 = 1 - (1 - 0,5^2)^4 = 0,68 \text{ no me sirve}$$

~~Pruebo con 16 minhashes, 4 AND, 4 OR~~

$$p_1 = 1 -$$

~~Como me pide una~~

~~No puedo satisfacer lo pedido con $d=0,5$, pero~~

~~como me piden $d_2 < 0,5$~~

Con 6 ANDS y 19 ORS

$$p_1 = 1 - (1 - 0,3^6)^{19} = 0,907 > 0,9$$

$$p_2 = 1 - (1 - 0,5^6)^{19} = 0,2586 < 0,3$$

Usando 114 minhashes

C + merge sort
y repeticiones

6) Partiría el archivo en particiones y usaría un selection sort para armar particiones ordenadas distribuidas en los discos disponibles.

Luego, haría un merge-sort entre todas las etapas.

Si tengo muchos discos, entonces puedo paralelizar los seeds y no se justifica demasiado hacer el merge en varias fases, así que lo haría en una. Asumo que el I/O es cuello de botella.

4) Para que la función sea perfecta ~~more~~ el espacio de direcciones debe tener al menos 4 lugares. El m^{Kimb} mas chico que cumple con esto es ~~5~~.

con $a=1$
 $b=2$

$$1, 1+2 \bmod 5 = 3$$

$$3+2 \bmod 5 = 0$$

$$5+2 \bmod 5 = 2$$

$$12+2 \bmod 5 = 4$$