WIKIRECOMMENDER

"MÓDULO DE RECOMENDACIONES DE PÁGINAS A VISITAR EN LA WIKIPEDIA, BASADO EN LAS APORTACIONES EFECTUADAS POR LA COMUNIDAD DE USUARIOS USANDO HADOOP"

Presentado por:

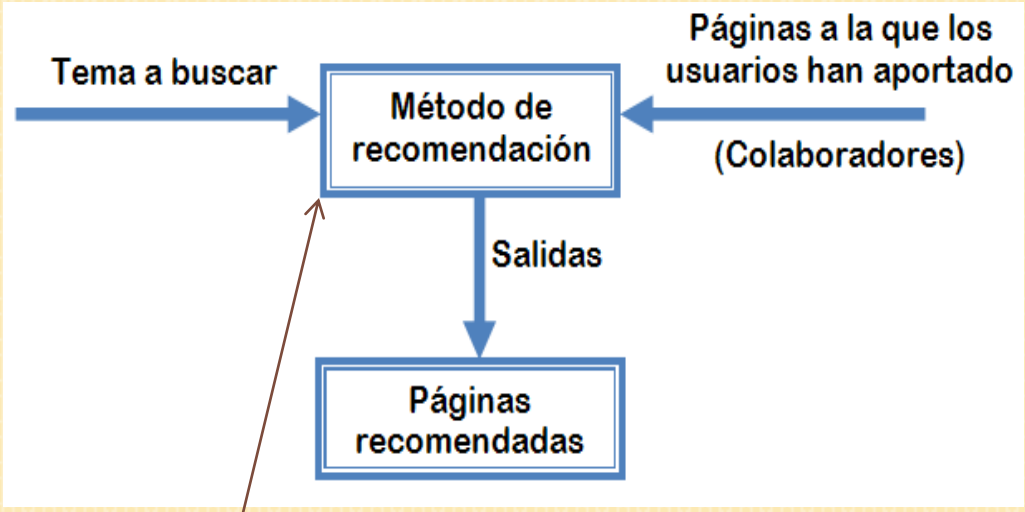
- Andrés Cantos Rivadeneira
- Bolívar Elbert Pontón

AGENDA

- Introducción
- Diseño y metodología utilizada
  - Filtrado de información
  - Valoración de páginas a recomendar
  - Selección de páginas a Recomendar
  - Arquitectura Final
- Demostración
- Conclusiones y recomendaciones

INTRODUCCIÓN

ESQUEMA DEL SISTEMA DE RECOMENDACIÓN UTILIZADO



Coeficiente de Similitud de Jaccard

Método de recomendación utilizado, para tratar en lo posible de mantener la coherencia de datos

OBJETIVOS DEL PROYECTO

Objetivo general

Desarrollar un módulo de recomendaciones para la Wikipedia basado en las *wikis* a la que los usuarios aportan, utilizando el *paradigma* MapReduce y herramientas libres para el procesamiento masivo de datos.

Objetivos específicos

- Analizar los dumps de la Wikipedia en español proporcionados por La Fundación Wikimedia utilizando Hadoop.
- Implementar el coeficiente de similitud de Jaccard en su versión Map/Reduce para seleccionar las mejores páginas a recomendar.
- Verificar si los usuarios de la Wikipedia en general, aportan sobre temas similares de parecido contenido o al contrario las aportaciones de los usuarios es arbitraria y no sigue una misma temática.
- Analizar la escalabilidad de los diferentes procesos Map/Reduce utilizados en el desarrollo del proyecto.

AGENDA

- ✓ Introducción
- ✖ Diseño y metodología utilizada
  - + Filtrado de información
  - + Valoración de páginas a recomendar
  - + Selección de páginas a recomendar
  - + Arquitectura final
- ✖ Pruebas de escalabilidad
- ✖ Demostración
- ✖ Conclusiones y recomendaciones

DISEÑO Y METODOLOGÍA UTILIZADA

Consiste en 3 etapas

1. Filtrado de Información

   Descartamos datos que podrían involucrar errores en la obtención de las recomendaciones.

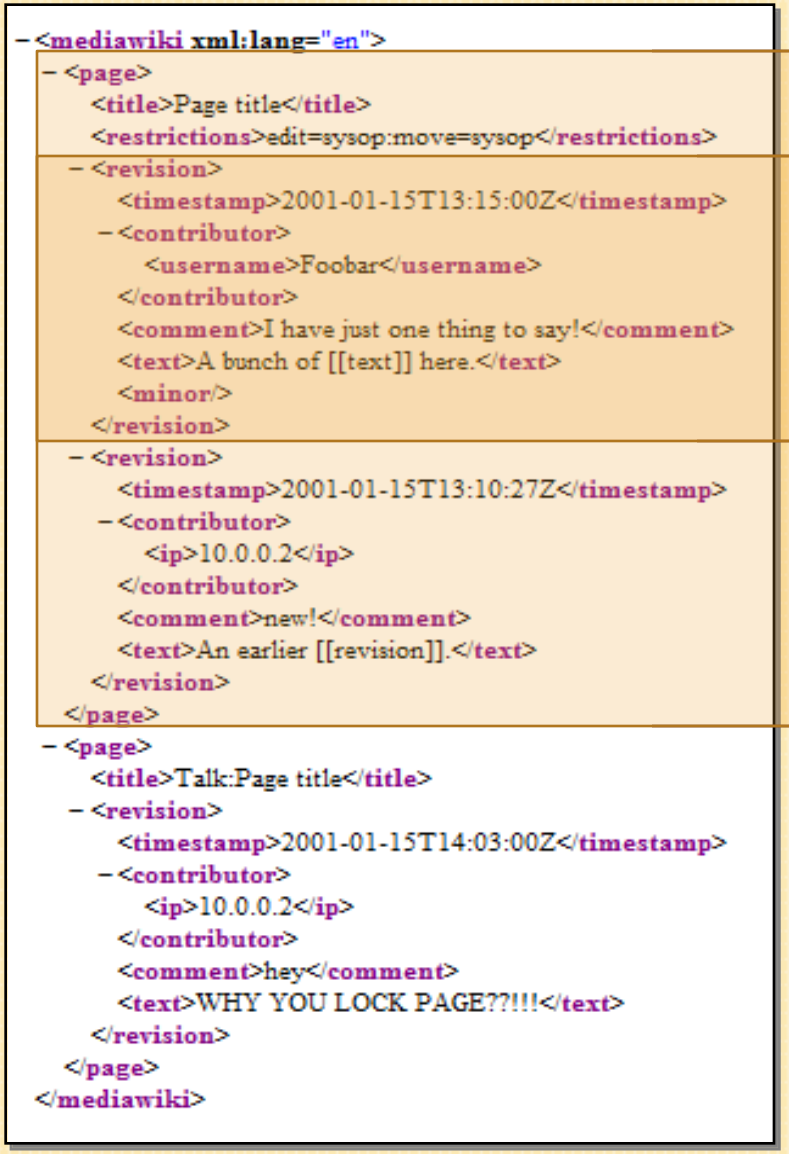
2. Valoración de páginas a recomendar

   Desarrollamos un algoritmo para calcular el coeficiente de similitud de Jaccard para todas las posibles combinaciones de páginas.

3. Selección de páginas a recomendar

   Ordenar de mayor a menor los resultados obtenidos en la etapa anterior.

FILTRADO DE INFORMACIÓN



La etapa de filtrado consiste en no tomar en cuenta datos que podrían introducir errores en la obtención de nuestras recomendaciones.

Nos hemos enfocado en dos partes :

- ❑ Usuarios
- ❑ Páginas

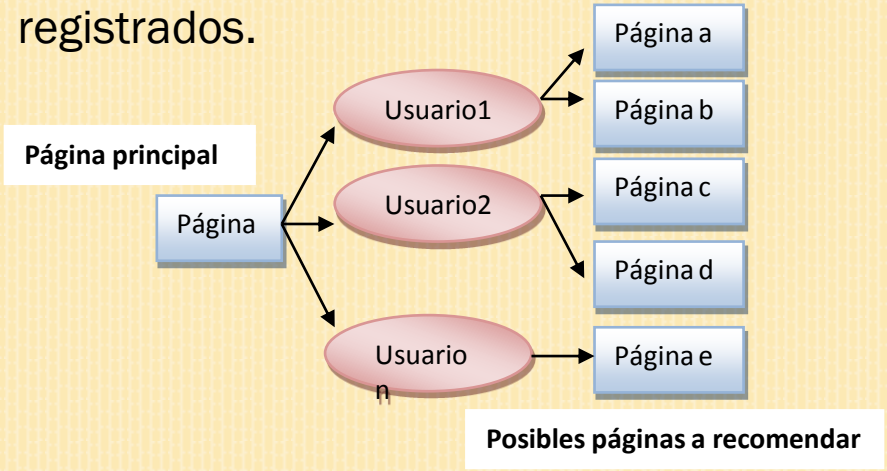
FILTRADO DE INFORMACIÓN → FILTRADO DE USUARIOS

```
- <mediawiki xml:lang="en">
  - <page>
      <title>Page title</title>
      <restrictions>edit=sysop:move=sysop</restrictions>
    - <revision>
        <timestamp>2001-01-15T13:15:00Z</timestamp>
      - <contributor>
          <username>Foobar</username>
        </contributor>
        <comment>I have just one thing to say!</comment>
        <text>A bunch of [[text]] here.</text>
        <minor/>
      </revision>
    - <revision>
        <timestamp>2001-01-15T13:10:27Z</timestamp>
      - <contributor>
          <ip>10.0.0.2</ip>
        </contributor>
        <comment>new!</comment>
        <text>An earlier [[revision]].</text>
      </revision>
    </page>
  - <page>
      <title>Talk:Page title</title>
    - <revision>
        <timestamp>2001-01-15T14:03:00Z</timestamp>
      - <contributor>
          <ip>10.0.0.2</ip>
        </contributor>
        <comment>hey</comment>
        <text>WHY YOU LOCK PAGE??!!!</text>
      </revision>
    </page>
</mediawiki>
```

Tipos de Usuarios en la Wikipedia:

- ❑Usuarios registrados
- ❑ Anónimos
- ❑Programas robots <<bots>>
  - ❑Ej. AVBOT

Solo son tomadas en cuenta las contribuciones de los usuarios registrados.



FILTRADO DE INFORMACIÓN → FILTRADO DE PÁGINAS

```
- <mediawiki xml:lang="en">
  - <page>
      <title>Page title</title>
      <restrictions>edit=sysop:move=sysop</restrictions>
    - <revision>
        <timestamp>2001-01-15T13:15:00Z</timestamp>
      - <contributor>
          <username>Foobar</username>
        </contributor>
        <comment>I have just one thing to say!</comment>
        <text>A bunch of [[text]] here.</text>
        <minor/>
      </revision>
    - <revision>
        <timestamp>2001-01-15T13:10:27Z</timestamp>
      - <contributor>
          <ip>10.0.0.2</ip>
        </contributor>
        <comment>new!</comment>
        <text>An earlier [[revision]].</text>
      </revision>
    </page>
  - <page>
      <title>Talk:Page title</title>
    - <revision>
        <timestamp>2001-01-15T14:03:00Z</timestamp>
      - <contributor>
          <ip>10.0.0.2</ip>
        </contributor>
        <comment>hey</comment>
        <text>WHY YOU LOCK PAGE??!!!</text>
      </revision>
    </page>
</mediawiki>
```

Tipos de Páginas en la Wikipedia:

- ❑Discusión
- ❑ Redirects
- ❑ Configuración
- ❑ Artículos

Son descartadas las páginas que no corresponden a un artículo

FILTRADO DE INFORMACIÓN → ALGORITMO

```
public void map(LongWritable key, WikipediaPage pagwiki, OutputCollector<Text, Text> output, Reporter
reporter) throws IOException {
Text okey = new Text(pagwiki.getMIdpage());
if( pagwiki.getTitle().indexOf(":")== -1){
            reporter.incrCounter(Counters.NUMPAGES, 1);
            String xmlpagina ="";
            xmlpagina = pagwiki.getMPage();
        try{
                    ArrayList lista = Obtenerrevisiones(xmlpagina);
                    for (int i=0; i<lista.size();i++){
                                Revision r = (Revision) lista.get(i);
                                String usuario = r.getUsername().toLowerCase();
                                if (usuario.indexOf("bot")==-1){
                                            output.collect(okey, new Text(r.getIdc()));
                                }
            }
            }catch (Exception e) { e.printStackTrace(); }
  }
}
```

El map recibe un objeto tipo WikipediaPage (Clase importada de la librería Cloud9)

Solo si la página es un artículo la considero.

No tomo en cuenta los robots, verifico si el usuario contiene la palabra bot. Solo en el caso de que no sea un robot emito el map.

VALORACIÓN DE PÁGINAS A RECOMENDAR

Para valorar que páginas son las mejores páginas para recomendar, hemos decidido utilizar el **Coeficiente de Similitud de Jaccard**.

COEFICIENTE DE SIMILITUD DE JACCARD

Medida estadística de la similitud entre los conjuntos de la muestra.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Se define como la cardinalidad de la intersección entre dos conjuntos dividido para la cardinalidad de su unión

El resultado de esta operación es un valor porcentual y va desde 0 hasta 1

Para el caso de la Wikipedia es necesario un enfoque ligeramente diferente
Tomando como CONJUNTO a los usuarios que han aportado a dos páginas específicas, es posible determinar la similitud que hay entre esas dos páginas.

VALORACIÓN DE PÁGINAS A RECOMENDAR

COEFICIENTE DE SIMILITUD DE JACCARD APLICADO A LA WIKIPEDIA

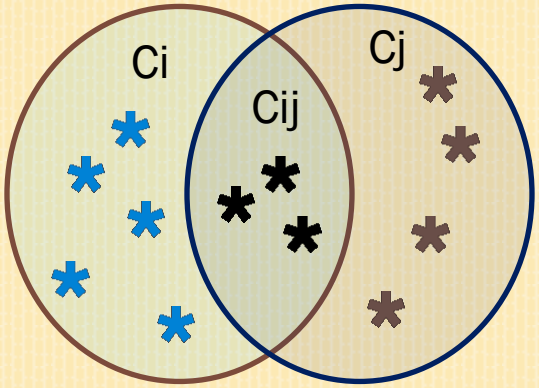
Similitud entre la página A y B es

$$J(A,B) = \frac{|X \cap Y|}{|X \cup Y|}$$

Numerador es el número de usuarios que han editado ambas páginas

Denominador es el número de usuarios que han editado una o ambas páginas

X, Y corresponden al conjunto de usuarios que se produce con A y B respectivamente



Página A Página B

$$Sim(A,B) = \frac{Cij}{Ci + Cj - Cij}$$

$$Sim(A,B) = \frac{\theta}{\theta + \emptyset - 3} = \mathbf{0.0} = 1\%$$

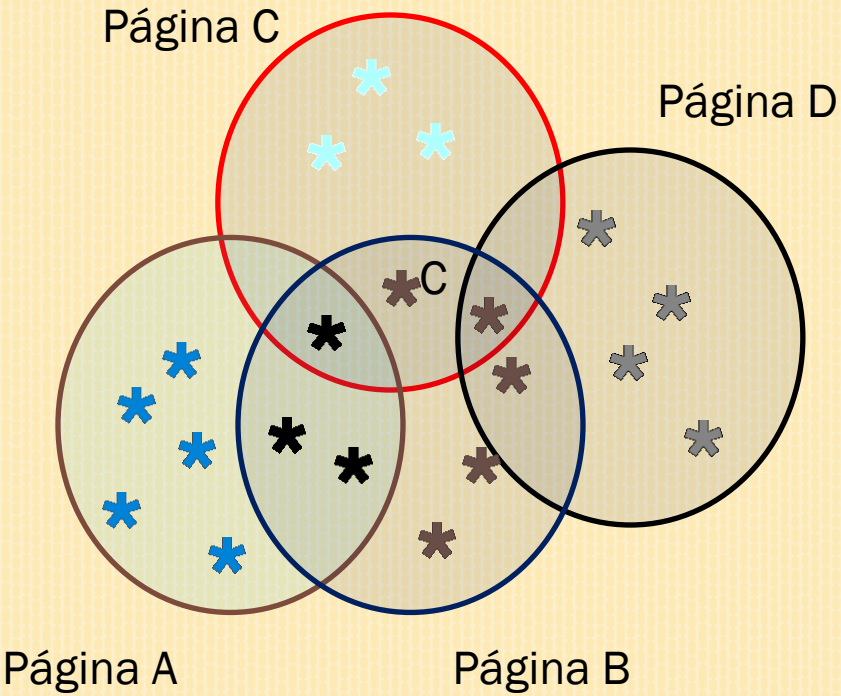
Entre la página A y la Página B hay un 1% de similitud.

Son seleccionadas las 5 páginas con el coeficiente de similitud de jaccard más elevado

VALORACIÓN DE PÁGINAS A RECOMENDAR

COEFICIENTE DE SIMILITUD DE JACCARD → ALGORITMO

$$J(A,B) = \frac{|X \cap Y|}{|X \cup Y|}$$



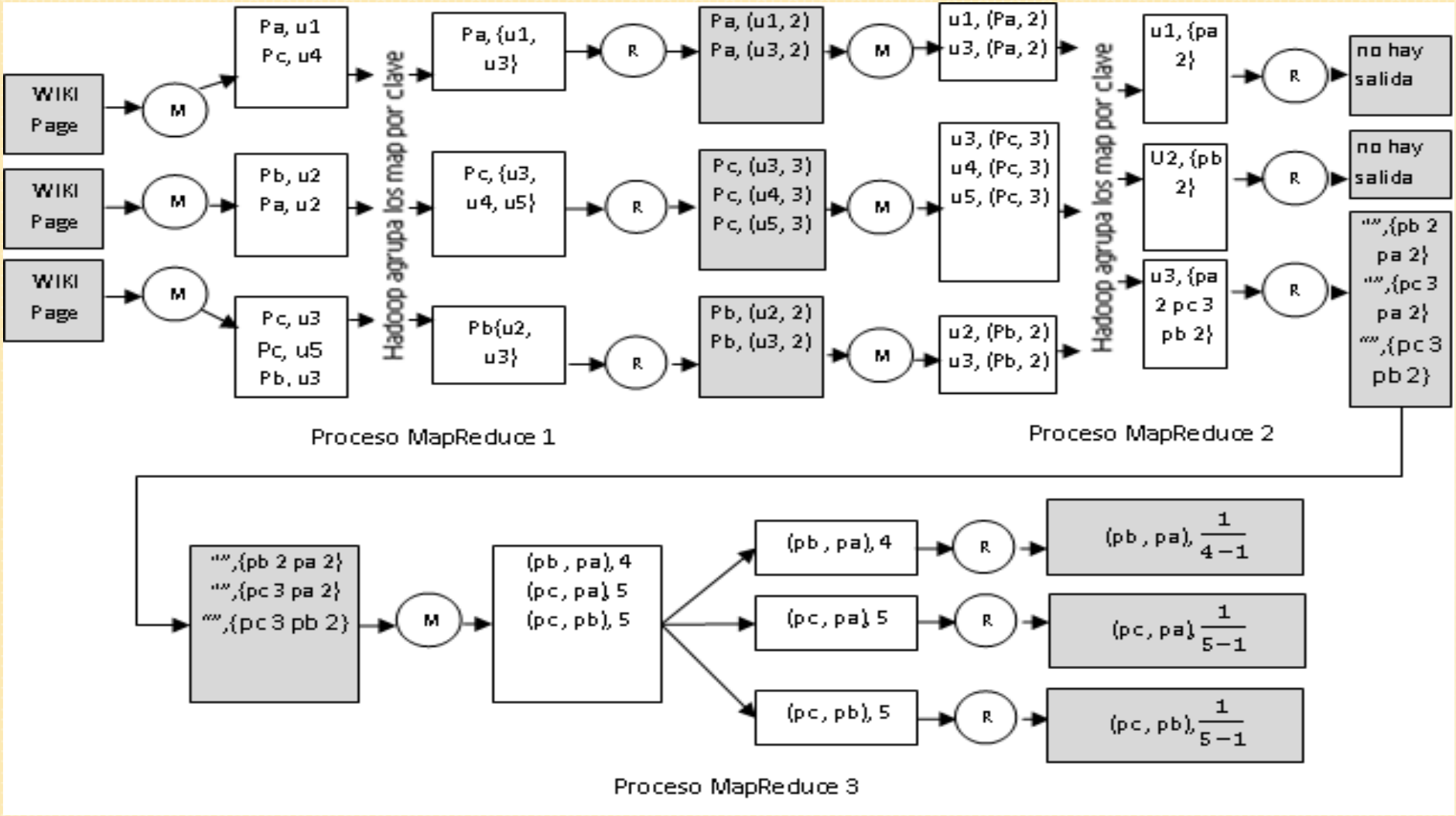
Pasos:

1. Calcular el número de usuarios que tiene cada página.
2. Generar las combinaciones posibles de páginas que tienen usuarios en común.
3. Calcular el coeficiente de similitud de Jaccard para cada par de páginas.

Cada paso constituye un Proceso Map/Reduce en nuestra solución final

VALORACIÓN DE PÁGINAS A RECOMENDAR

COEFICIENTE DE SIMILITUD DE JACCARD → ALGORITMO MAP/REDUCE

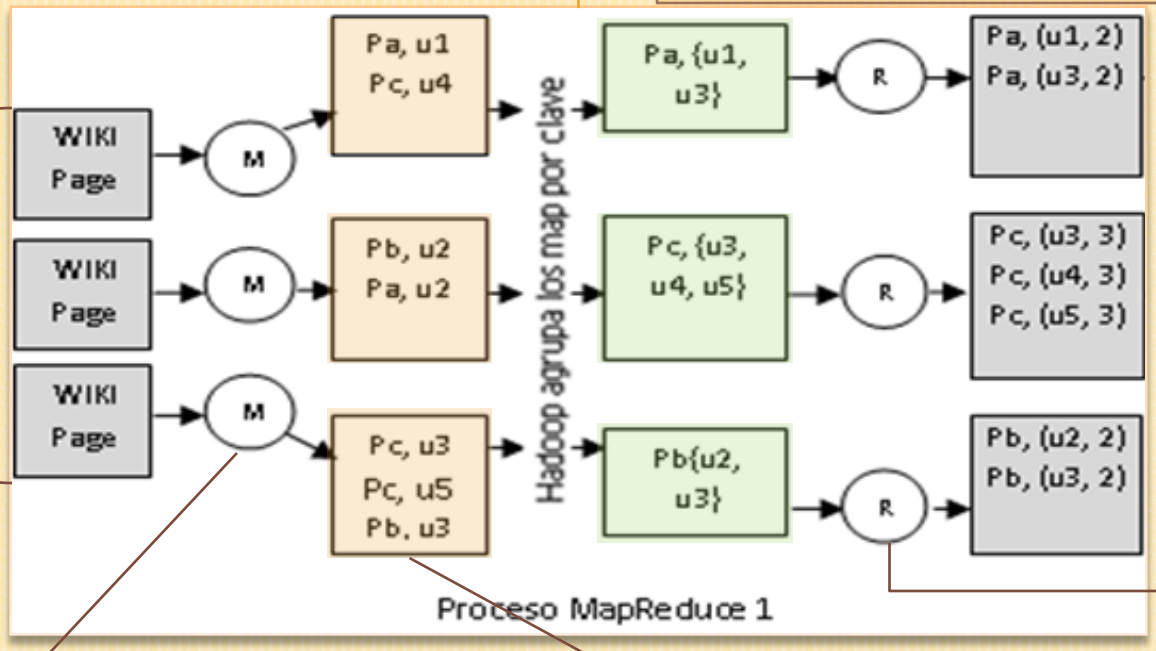


VALORACIÓN DE PÁGINAS A RECOMENDAR

COEFICIENTE DE SIMILITUD DE JACCARD → ALGORITMO MAP/REDUCE

1.- Calcular el número de usuarios que tiene cada página

Hadoop se encarga de agrupar por clave los maps emitidos; y **me retorna una lista de usuarios por página.**



Entrada: Respaldos XML de La Wikipedia

Obtiene el tamaño de la lista; por cada usuario de la lista se emite una tupla de la forma **<Pág, (Usu, Lst _size)>**

Proceso de Filtrado

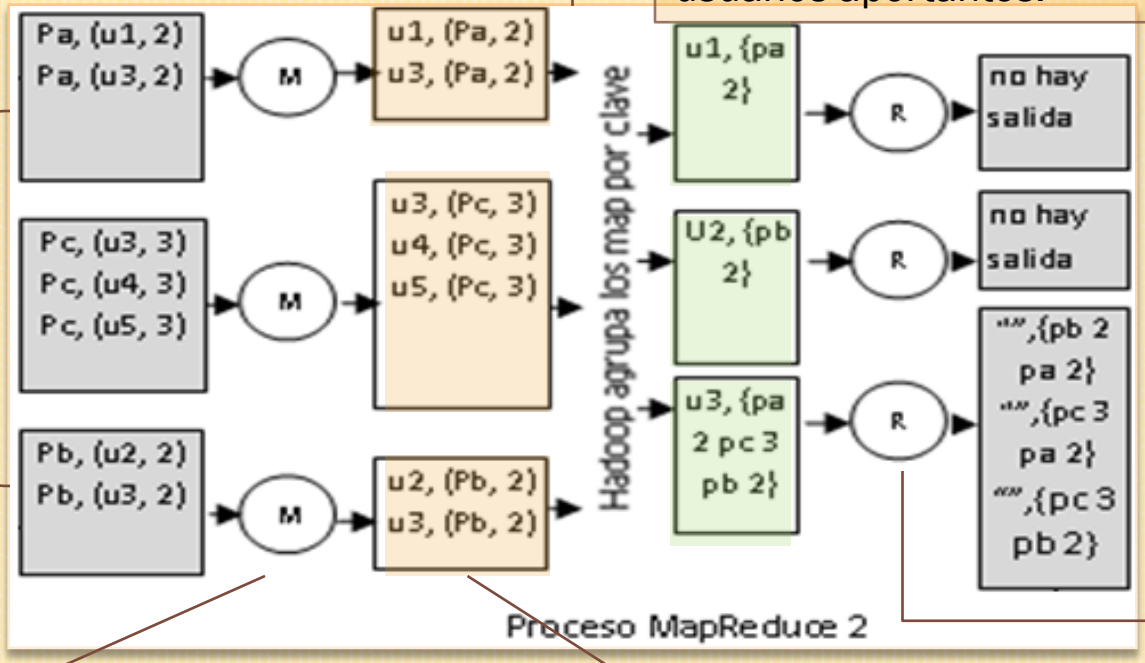
Genera maps de la forma **<Página, Usuario>** con la información de las Revisiones

VALORACIÓN DE PÁGINAS A RECOMENDAR

COEFICIENTE DE SIMILITUD DE JACCARD → ALGORITMO MAP/REDUCE

2.- Generar todas las combinaciones posibles de páginas que tienen usuarios en común.

Hadoop agrupa por clave los maps; y **me retorna una lista de páginas agrupadas por usuario**, cada página tiene asociado su numero total de usuarios aportantes.



Entrada: Salida del proceso MapReduce anterior

Aquí se generan todas las combinaciones posibles de páginas por usuario. Se emite un reduce por cada combinación y de clave se **utilizó** un string vacio.

<"", (pága Lst_size, págb Lst_size)>

Por cada línea del archivo se genera un Map

El map emitido tiene como clave al usuario y tiene la forma
<usuario, (página, Lst_size)>

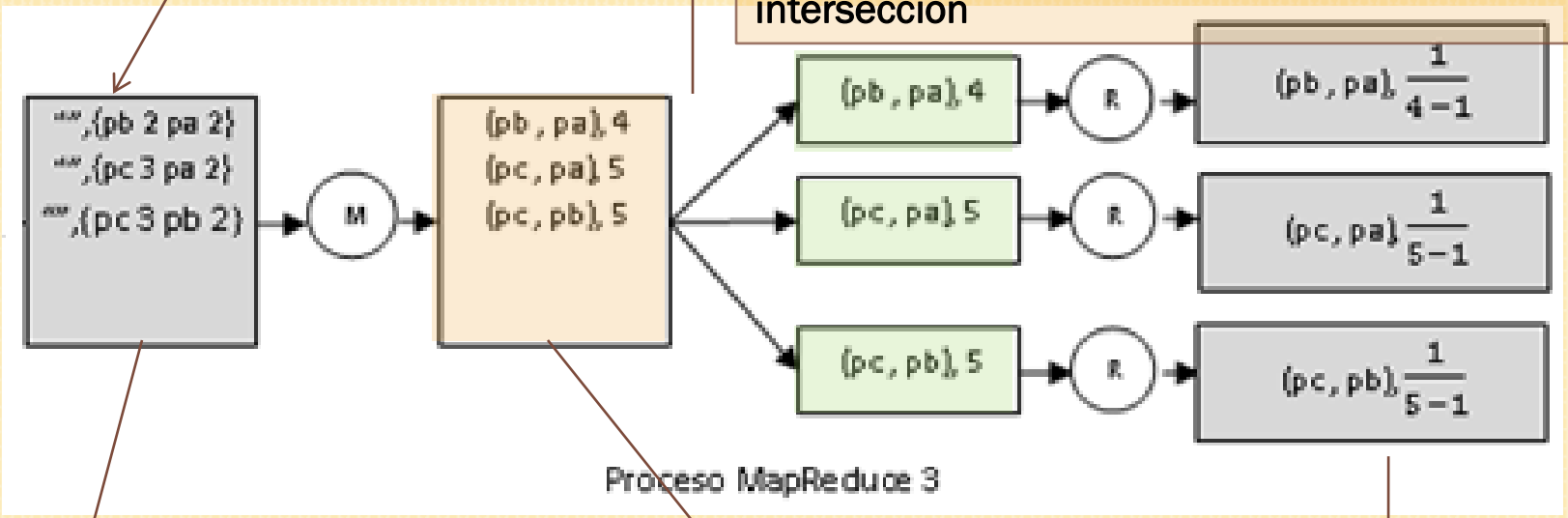
VALORACIÓN DE PÁGINAS A RECOMENDAR

COEFICIENTE DE SIMILITUD DE JACCARD → ALGORITMO MAP/REDUCE

2.- Calcular el coeficiente de similitud de Jaccard para cada par de páginas

Entrada:
Archivo de Salida generado en el proceso MapReduce anterior

Hadoop agrupa por clave los maps; y **me retorna una lista donde cada elemento es el valor de la suma del número de aportantes de cada página Y EL TAMAÑO DE LA LISTA representa la intersección**



Por cada línea del archivo se genera un Map

La clave del map es el par de páginas y el valor es la suma del número de aportantes de cada página
<(págia, págb), suma_Lst_size>

$$Sim(A,B) = \frac{Cij}{Ci + Cj - Cij}$$

SELECCIÓN DE PÁGINAS A RECOMENDAR

Archivo de salida del proceso de valoración de páginas:

(Pag1, Pág. a)	0.20
(Pag1, Pág. b)	0.40
(Pag1, Pág. c)	0.70
(Pág. a, Pág. b)	0.10
(Pág. a, Pág. c)	0.01
(Pág. b, Pág. c)	0.03
.	
.	

Esta etapa consiste en ordenar de mayor a menor las recomendaciones para una página.

Sim (Pag1, Pág. a) = 0.2
Sim (Pag1, Pág. b) = 0.4
Sim (Pag1, Pág. c) = 0.7

La mejor recomendación para la Pag1 es la Pag c

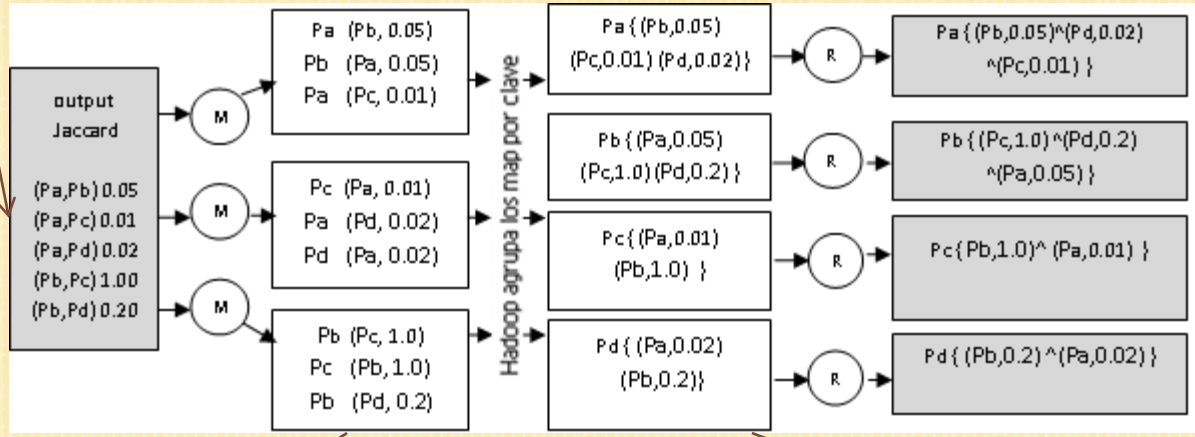
Salida final:

Pag1 Pág. c, 0.7 ^Pág. b, 0.4 ^Pág. a, 0.2
Pág. a Pag1, 0.2 ^Pág. b, 0.10 ^Pág. c, 0.01
Pág. b Pag1, 0.4 ^Pág. a, 0.10 ^Pág. c, 0.03

SELECCIÓN DE PÁGINAS A RECOMENDAR

ALGORITMO

Entrada:
Archivo de Salida generado en el proceso MapReduce anterior



Por cada línea del archivo emito dos map, donde la clave es la página a recomendar y el valor la página recomendada con su valor de similitud

Recibo una lista de recomendaciones para una determinada página.
Ordeno de mayor a menor las recomendaciones, utilizando el método Sort de la clase Collections de Java

ARQUITECTURA FINAL

Pag1 Pág. c, 0.7 ^Pág. b, 0.4 ^Pág. a, 0.2
Pág. a Pag1, 0.2 ^Pág. b, 0.10 ^Pág. c, 0.01
Pág. b Pag1, 0.4 ^Pág. a, 0.10 ^Pág. c, 0.03

El proceso de generar las recomendaciones no es un proceso en línea.

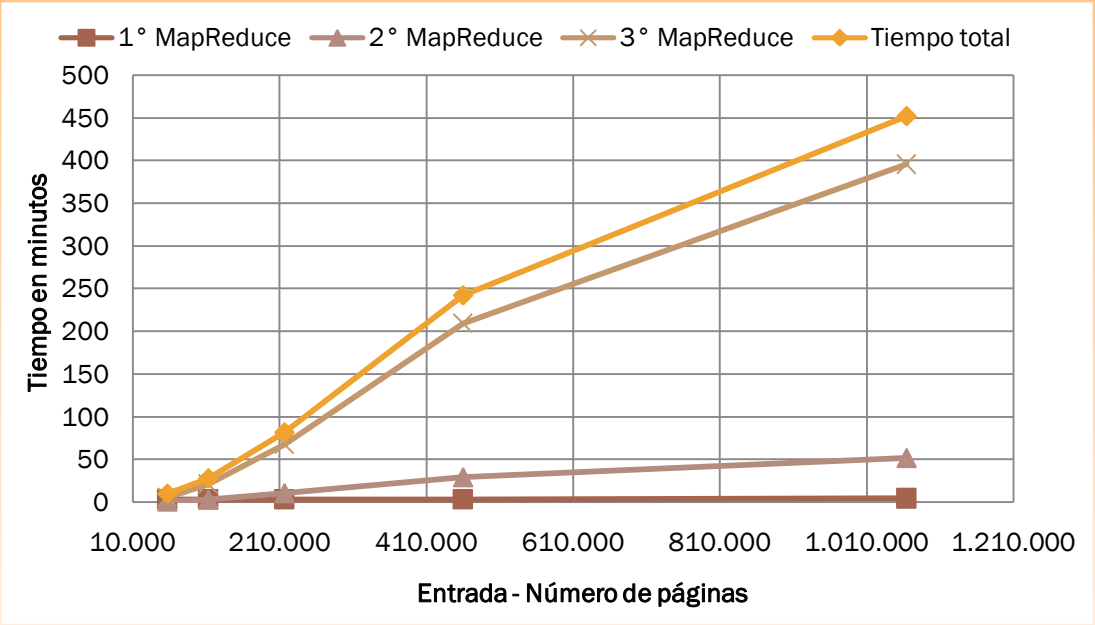
AGENDA

- ✓ Introducción
- ✓ Diseño y metodología utilizada
  - + Filtrado de información
  - + Valoración de páginas a recomendar
  - + Selección de páginas a recomendar
  - + Arquitectura final
- ✖ Pruebas de escalabilidad
- ✖ Demostración
- ✖ Conclusiones y recomendaciones

PRUEBAS DE ESCALABILIDAD

Obtención del Coeficiente de similitud de jaccard

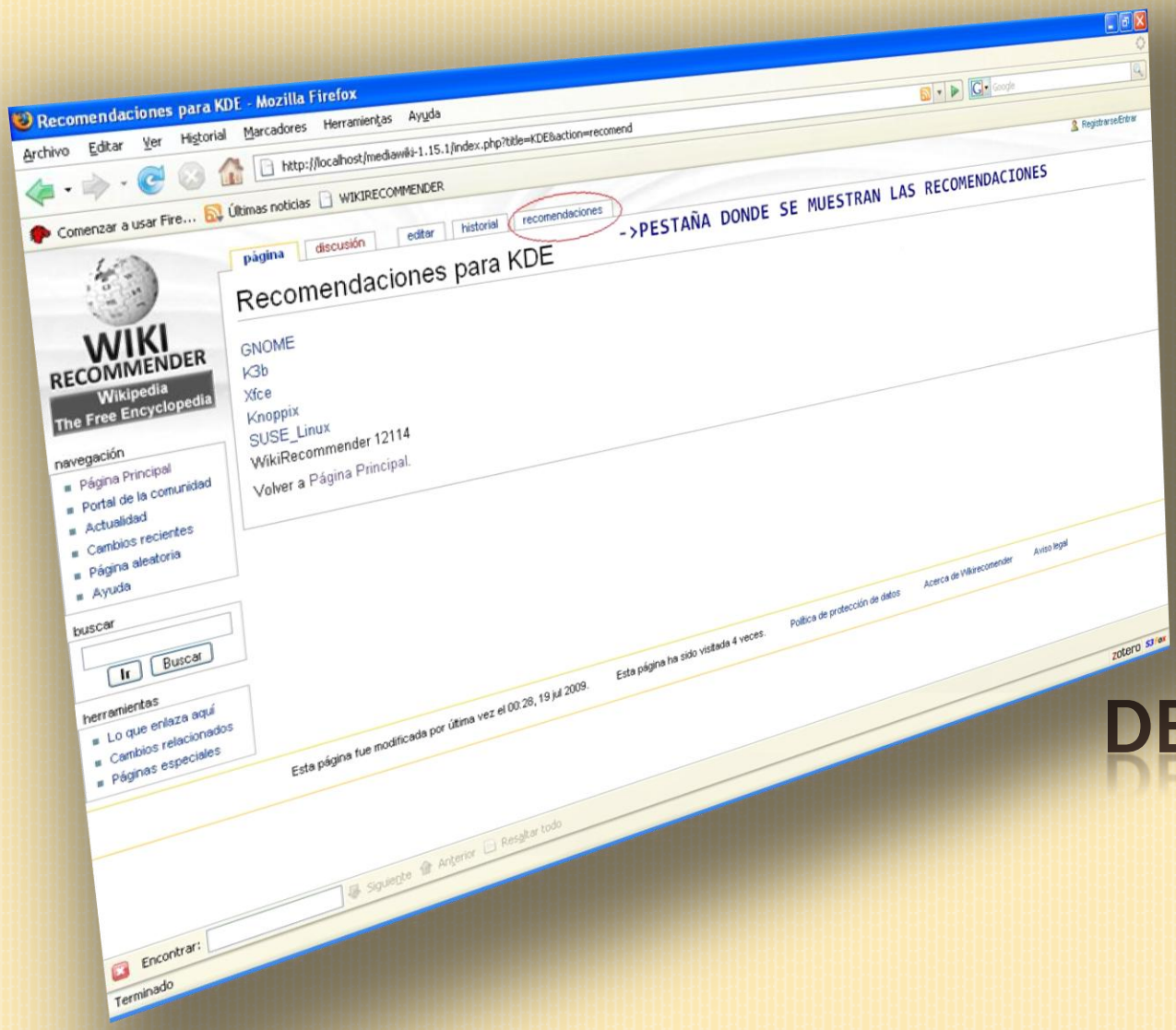
Proceso MapReduce / N° páginas entrada	57.000	112.468	216.903	459.866	1.064.418
1° MapReduce	3,07	3,15	3,27	3,45	4,39
2° MapReduce	1,21	3,56	11,05	29,39	52,25
3° MapReduce	5,56	21,21	67,57	209,56	395,42
Tiempo Total	9,84	27,92	81,89	242,40	452,06



AGENDA

- ✓ Introducción
- ✓ Diseño y metodología utilizada
  - + Filtrado de información
  - + Valoración de páginas a recomendar
  - + Selección de páginas a recomendar
- ✓ Arquitectura final
- ✓ Pruebas de escalabilidad
- ✖ Demostración
- ✖ Conclusiones y recomendaciones

WIKIRECOMMENDER



DEMOSTRACIÓN

AGENDA

- ✓ Introducción
- ✓ Diseño y metodología utilizada
  - + Filtrado de información
  - + Valoración de páginas a recomendar
  - + Selección de páginas a recomendar
- ✓ Arquitectura final
- ✓ Pruebas de escalabilidad
- ✓ Demostración
- ✖ Conclusiones y recomendaciones

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

1. El costo total de generar las recomendaciones fue de tan solo $0,50 y para almacenar la salida de los procesos $1,50.

2. Los usuarios de la Wikipedia en ocasiones aportan mayormente a *wikis* que tienen que ver con la misma temática y esto nos permite mantener una consistencia de información.

3. Si lo que se quiere es tener recomendaciones basadas totalmente en contenido con un 100% de consistencia de datos, no es suficiente basar las recomendaciones en las aportaciones de los usuarios.

4. Para el caso particular de la Wikipedia que no almacena nada de información sobre gustos o preferencias de los usuarios, utilizar las aportaciones de los usuarios es un método valedero y rápido para generar recomendaciones

CONCLUSIONES Y RECOMENDACIONES

Recomendaciones

1. Evitar utilizar cadenas de caracteres largas para las salidas MapReduce. En nuestro caso utilizamos los Id de las páginas que son enteros de longitud 10 y notamos una mejora en el tiempo de los procesos.

2. Hadoop después de 10 minutos de detectar inactividad en un proceso procede a matar el proceso, es recomendable utilizar la instrucción reporter.progress() dentro de los *bucles* que toman más de 10 minutos, para indicarle a Hadoop que el proceso sigue ahí.

3. Además de utilizar el índice de similitud de Jaccard, se podría mejorar el sistema de recomendación añadiéndole características como retroalimentación y *heurísticas* que permitan obtener recomendaciones basándose en otros criterios.

¿PREGUNTAS?

GRACIAS POR SU ATENCIÓN