Universidad Central de Venezuela
Facultad de Ciencias
Postgrado Ciencias de la Computación
Análisis de Documentos en Archivos Web

Proyecto SEGQL

Se requiere desarrollar un programa en Ruby que permita a un usuario interactuar con Block-o-Matic y permitir realizar una segmentación de una página Web, utilizando un lenguaje similar al lenguaje SQL, que llamaremos SEGQL. Este nuevo lenguaje permite listar un *blockset* tal como se haría en un *resultset* de una base de datos relacional. Sin embargo, se incluye una operación de unión o *merge* que agrega una complejidad adicional a considerar.

En este nuevo lenguaje SEGQL funcionara por lineas, donde cada linea tiene un significado.

```
(select|merge)
('*' | <atributo>[,<atributo>]*)
from
<URL>
where
<condicion>
order by
(<atributo> ['asc' | 'desc'][,])*
params
<parametro>=<valor>[,<parametro>=<valor>]*
```

Donde,
Función*:* es la operación a ejecutar. Por ejemplo **select** o **merge.** *Select,* devuelve una lista de bloques o *blockset* que cumplen con la condición, respetando el orden especificado. Merge, devuelve un *blockset* luego de mezclar sus rectángulos, habiendo considerado las condiciones y parámetros de la segmentación.
Atributo: es cualquiera de los atributos presentes en la segmentación (*ie* el JSON producido)
URL: Es el *url* de la pagina registrada en el repositorio.
Condición: Una expresión en función de constantes, literales y <atributos>.
Params: parámetros de la segmentación. Si no se especifican se usa un valor por defecto.

Para interpretar correctamente este tipo de instrucciones, la primera linea sera el **select** o **merge**. Al leer la segunda linea, se lee una lista delimitada por comas con los atributos a

considerar. Luego se leería el **from** y así sucesivamente. Las lineas **where, order by** y **params** son opcionales, así como su condición, atributos y valores.

Debe considerar valores por defecto. Por ejemplo, si no se especifica '*desc*' en el *order by* siempre se asume 'asc'. Si no se incluye el parametro 'categoríes' de BoM se asume FLOW, así para cada uno de los parámetros.

Ejemplos

Consideremos la segmentación de la página http://edition.cnn.com/US/OJ, con los siguientes parámetros: `ptype=ffront, dc=30, area=0, categories=FLOW, method=html5 granmethod=frec, align=HVL, pa=5`, y el query siguiente:

```
select
bid,area
from
http://edition.cnn.com/US/OJ
params
pa=5
```

Produce el siguiente blockset:

bid	area
B780	560.5261875
B777	756.90631640625
B482	127.47459375

Consideremos otro query.

```
select
bid,text,pa
from
http://edition.cnn.com/US/OJ
where
pa>5
order by
pa desc
```

el blockset:

bid	text	pa
B777	trial of the century ends with simpson s acquittal october 3 the reaction…	8
B780	the victims the evidence other views simpson…	6

Finalmente un ejemplo con mezcla:

```
merge
bid,text,pa,area
from
http://edition.cnn.com/US/OJ
where
pa>5
```

bid	text	pa	area
B777-780	trial of the century ends with simpson s acquittal october 3 the reaction… the victims the evidence other views simpson…	10 (*)	1676402,82

(*) Estos valores se obtienen de unir ambos bloques, sin embargo, para poder dar un valor al **pA** del nuevo bloque B777-780 se debe realizar un calculo. Éste calculo se describe a continuación.

Calculo del pA en un merge

- Ordernar **todos** los bloques en una segmentación por el área ascendente.
- Calcular el area acumulada.

$$area\_acum(b_i) = area\_acum(b_{i-1}) + area(b_i)$$

- Tomar los valores máximos y mínimos del área acumulada como *max_ac* y *min_ac*, respectivamente.
- Dividir en 10 intervalos usando un incremento de $(max\_ac - min\_ac)/10$, desde 0 hasta *max_ac*.
- Ubicar el área acumulada de un bloque, y al que pertenezca, su indice sera el pA.

Consideremos el ejemplo anterior, el calculo del pA del bloque B777-780.

En la segmentación en general luego de mezclar B777 con B780 quedarían dos bloques: B482 con un área de 127.48cm² y B777-B780 con 1.676.402,82cm²

```
area_acum = [127.48, 1676530,29]
```

```
max_ac =  1676530,29
min_ac =  127.48
tamaño del intervalo: 167640,29
```

PA	Desde	Hasta	Bloques
1	0	167640,28	B482
2	167640,28	335280,56	
3	335280,56	502920,84	
4	502920,84	670561,13	
5	670561,13	838201,41	
6	838201,41	1005841,69	
7	1005841,69	1173481,97	
8	1173481,97	1341122,25	
9	1341122,25	1508762,53	
10	1508762,53	1676402,82	B777-B780

Bono de 2 puntos adicionales

Cada vez que se unen dos bloques es necesario recalcular el pA, dado que se incorporan nuevos bloques y ocultan otros. Si usted propone un mecanismo que permita dar valor al pA sin tener que recalcular los intervalos obtendrá dos (2) puntos adicionales sobre 20.

Observaciones finales

- El proyecto puede ser en grupo de máximo (2) personas
- Debe ser entregado vía Moodle. Entregar un documento PDF detallando su solución y los archivos .rb necesarios.
- No se pide interfaz gráfica, por lo cual es opcional, mínimo debe funcionar vía linea de comandos.
- Puede utilizar el cliente *bom* de la linea de comandos o si lo prefiere interactuar con el API directamente mediante HTTP.
- Debe incluir **obligatoriamente** los datos del grupo comentado en todos los archivos.
- Cualquier consulta se realizar vía Piazza o en horario de clases.