Pagerank

13-11-2022

He Chen

Daniel Muñoz

Grupo 12

What is (1-L)/n term in line 9 of the pseudocode mean? What is the role of P and Q?
P es un vector de pesos de cada aeropuerto del grafo G. Inicialmente todos los elementos de P están inicializados a 1/n, sea n es el número total de nodos(aeropuertos). Lo podemos interpretar como el pagerank actual.
Q es un vector de pesos auxiliar de cada aeropuerto del grafo G. En cada iteración del bucle, en Q se recalculan los pesos de cada aeropuerto en función de P como aparece en el código (líneas 6,7,8 y 9). Luego se comprueba la diferencia entre P y G para ver si converge (stop condition).
L (damping factor) es la probabilidad de seguir una arista del nodo (aeropuerto) actual. (1-L)/n es la probabilidad de teletransportación (salto a un nodo cualquiera) de cada nodo.

Representacion de una arista
Las aristas (Edges) representan las aristas del grafo y las hemos representado de la siguiente manera: Si una ruta va del aeropuerto i al j, guardamos la arista en el aeropuerto destino (j) en routehash y por cada aeropuerto origen (i), creamos una arista que contiene el aeropuerto origen (i) y su peso, donde el peso nos indica cuantas veces aparece la arista (i,j) en el grafo.

Tratamiento de aeropuertos sin aristas de salida

Como dice el enunciado, la suma de los elementos de P tiene que ser 1. En nuestra primera versión del programa, la suma no se acercaba a este valor. Esto era porque no hemos tenido en cuenta los nodos que no tenían aristas de salida o incluso de entrada. Pero sabemos que podemos llegar a estos nodos desconexos por la probabilidad de teletransportación (1 - L), por lo tanto su pagerank no es 0 y tienen aportación a otros nodos.

Teniendo en cuenta la fórmula para calcular el pagerank, sea m el número de los nodos desconexos tienen n - 1 aristas virtuales no repetidas (out = 1). Por lo tanto dado un damping factor L, la aportación de 1 nodo desconexos es L/(n - 1), por lo que la aportación total para m nodos desconexos es m*L/(n - 1).

Para tratar eficientemente los aeropuertos que no forman parte de los destinos de ninguna ruta, les asignaremos un peso muy pequeño.

Para tratar los nodos que no tienen grado de salida, y por lo tanto no contribuyen al pagerank ya que según su fórmula, out(j) está dividiendo, por lo tanto si no tienen grado de salida será 0. Para que formen parte del cómputo del pagerank hemos obtenido la lista de estos aeropuertos y, si es par, creamos dos aristas virtuales entre dos aeropuertos de la lista (una del aeropuerto i a j y otra del j al i). Si es impar, creamos un ciclo de 3 con 3 aristas virtuales y el resto los tratamos por parejas como si fuese par. De esta manera, todos los nodos tendrán un grado de salida > 0.

De esta manera evitamos de anadir n - 1 arista a cada nodo reduciendo el coste a O(|E| + |vE|), siendo vE las aristas virtuales añadidas.

Efecto de la condición de parada

Para el mismo valor de damping factor de 0,85, ya que está entre el rango habitual según el pdf. Para hacer la condición de parada, inicialmente decidimos poner la diferencia entre el Page Rank actual y el de la siguiente iteración y este valor que sea menor que un número, ya que dependiendo de la iteración, el valor será diferente y a cada iteración, el valor entre P y Q irá convergiendo hasta que sea lo suficientemente pequeño para ser menor que el otro valor. Este valor, en el codigo se llama diff y para ajustarlo a un valor óptimo, hicimos la prueba con 10^(-1), 10^(-5), 10^(-10), 10^(-15) y 10^(-20).

valor de diff	Número iteraciones	Tiempo de ejecución
0.1	1	0.0928
10^(-5)	22	1.5755
10^(-10)	93	7.30
10^(-15)	163	11.60

Hemos escogido 10^(-10) porque creemos que tiene el tiempo más adecuado ya que cuando se ejecute con el damping factor de 0.99, con el diff de 10^(-10), tardaba 113.156, con lo que si escogemos el valor de 10^(-15), el tiempo será mucho más elevado.

Efecto del damping factor

Con un valor de 0.4 del damping factor, damos más peso a (1-L)/n. Al ejecutar el programa, podemos ver que ha tardado 1.468 segundos, ha hecho 18 iteraciones, con una diferencia de 10^(-10) y las puntuaciones del pagerank van entre 0.00209 a 0.00010.

Con un valor entre 0.8 y 0.9 como ponía en el enunciado, damos parte del peso al cómputo del page rank y un poco a (1-L)/n. Ejecutando el programa con un valor de 0.85, se puede observar por la terminal, que ha hecho 93 iteraciones, en un tiempo de 6.98 segundos y con 10^(-10) de diferencia entre iteraciones y con un valor de pagerank entre 0.00354 a 2.61 * 10^(-5).

Con un valor de 0.99 del damping factor, *damos prácticamente todo el peso al cómputo del page rank. Al ejecutar el programa,* podemos ver que ha tardado 113.156 segundos, que ha hecho 1500 iteraciones, que la diferencia entre las iteraciones es de 10^(-10) y las puntuaciones del pagerank van entre 0.00426 a 1.74*10^(-6).

Como se puede ver, mientras más alto sea el valor, más iteraciones hay más tiempo tarda porque tarda más en converger.