

Algoritmos para Big Data:

Grafos y PageRank

Sergio García Prado

E.T.S. Ingeniería Informática, UVa

Índice General

1. Introducción
2. Algoritmos para Streaming
3. Estrategias de Sumarización
4. Algoritmos aplicados a Grafos
5. Algoritmo PageRank
6. Implementación

Introducción

Los algoritmos para **Big Data** son aquellos que se encargan de resolver problemas sobre conjuntos de datos de tamaño masivo.

Problema de Accesos a Memoria

[TODO]

Soluciones a la complejidad del Big Data

- Algoritmos para Streaming
- Técnicas de Reducción de la Dimensionalidad
- Estrategias de Paralelización
- ...

Algoritmos para Streaming

Modelo en Streaming

- Serie Temporal:
 $1, 5, 3, -4, 2, -3, 5, \dots$
- Caja Registradora (Cash-Register):
 $(2, +1), (3, +4), (1, +3), (2, +3), (4, +5), \dots$
- Molinete (Turnstile):
 $(2, -1), (3, -4), (1, +3), (2, -3), (4, +5), \dots$

Algoritmos para Streaming

Los **Algoritmos para Streaming** son aquellos que procesan la entrada de manera secuencial, teniendo en cuenta únicamente el elemento actual, junto con una estimación de los procesados anteriormente, utilizando un orden sublineal $o(n)$ en espacio respecto del rango de posibles valores en la entrada.

$$F_k = \sum_{i=1}^n m_i^k \quad (1)$$

- Algoritmo de Morris: F_1
- Algoritmo de Flajolet-Martin: F_0
- Estimación de Momentos de Frecuencia: $F_k, k \in N^*$

Algoritmo de Flajolet-Martin

[TODO]

Estrategias de Sumarización

Estrategias de Sumarización

- Muestreo Aleatorio
- Histogramas
- Wavelets
- Sketches

- Count-Min Sketch
- Count Sketch
- AMS Sketch
- Hyper-LogLog
- L_p -Samplers

Count-Min Sketch

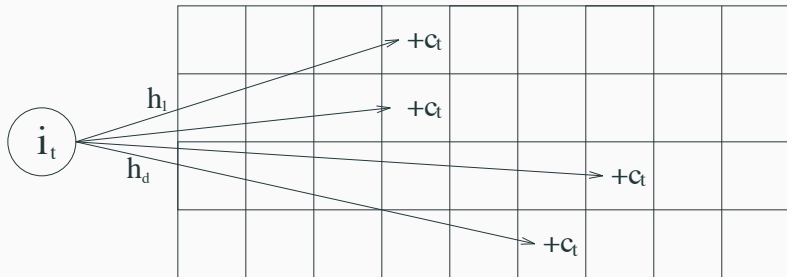


Figura 1

Algoritmos aplicados a Grafos

Algoritmos aplicados a Grafos

Sea $G = (V, E)$ un grafo formado por $n = |V|$ vértices y $m = |E|$ aristas, de tal manera que $e_i = (v_{i_1}, v_{i_2}) \in E$ y $\{v_{i_1}, v_{i_2}\} \in V$

Sobre el **Modelo en Semi-Streaming** se procesa un grafo a través del stream de aristas, en un espacio poli-logarítmico respecto del cardinal de vértices utilizando un número reducido de pasadas sobre el stream.

Spanners y Sparsifiers

- Spanner:
- Sparsifier:

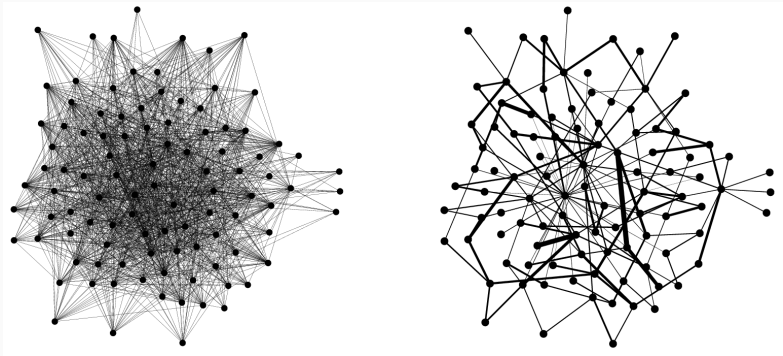


Figura 2

- Verificación de Grafo Bipartito
- Conteo de Triángulos
- Árbol Recubridor Mínimo
- Componentes Conectados
- Paseos Aleatorios
- ...

Algoritmo PageRank

Algoritmo PageRank

Ranking de los vértices de un grafo basado únicamente en la estructura de relaciones generada por las aristas del grafo.

Los vértices incidentes con vértices populares tendrán mayor popularidad.

Distribución estacionaria de la **Cadena de Markov** subyacente.

Cadena de Markov

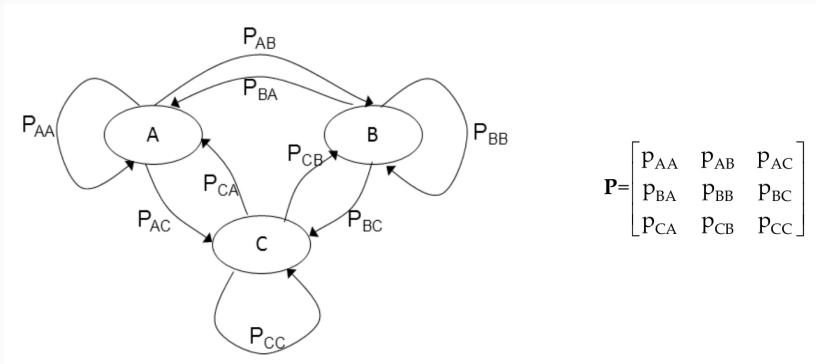


Figura 3

Vértices Sumidero no cumplen la propiedad de Cadena de Markov.

Salto Aleatorio como solución al problema de los vértices sumidero.

¿Cómo calcularlo?

- Algebraica [TODO]
- Iterativa [TODO]
- Paseos Aleatorios

[TODO]

- HITS
- SALSA
- SimRank

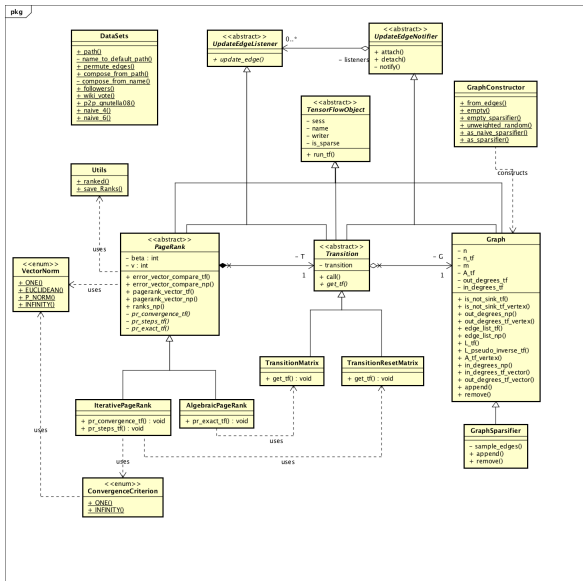
Implementación

Implementación

La implementación realizada consiste en una biblioteca de grafos (**tf_G**) utilizando como base la plataforma de cálculo matemático intensivo **TensorFlow**.

tf_G se encuentra en una fase muy temprana, formando únicamente el conjunto de métodos para el cálculo del **PageRank**.

Diagrama de Clases



Trabajo completo en:

[`https://github.com/garciparedes/tf_G`](https://github.com/garciparedes/tf_G)

¿Preguntas?

