webcrawlers

[webcrawlers]

- Introducción
- Un poco de historia
- Implementación
  - Características
  - Arquitectura
  - Políticas de amabilidad

- Estrategias
- Integridad y actualización
- Experiencias

[webcrawlers]

Definición:

Es un agente del tipo bot que recorre recursivamente el World Wide Web bajo algún orden predeterminado, y que recopila información acerca de los documentos que encuentra y su estructura de vínculos.

El índice de páginas generado por los crawlers es utilizado como parte central de cualquier sistema de acceso a la información en el WWW (como motores de búsqueda).

[webcrawlers]

Definición:

Es un agente del tipo bot que recorre recursivamente el World Wide Web bajo algun orden predeterminado, y que recopila información acerca de los documentos que encuentra y su estructura de vínculos.

El índice de páginas generado por los crawlers es utilizado como parte central de cualquier sistema de acceso a la información en el WWW (como motores de busqueda).

crawlear: Acción realizada por un crawler al recorrer el WWW.

[webcrawlers]

Áreas de desarrollo:

- Arquitectura e implementación
- Cómputo paralelo
- Identificación y control de crawlers por administradores de sitios
- Crawleo focalizado
- Estrategias de crawleo

Características del WWW

- Demasiado grande

  Ningún buscador conoce mas del 16% del web [Lawrence and Giles, 2000]

  Almacenar la información a Google le cuesta 4 millones de dolares

  Crawleando 100 páginas por minuto de manera continua, y considerando que sólo entra a una página una sola vez (!) tardaría 25 años en tener la base de datos de Google.

[webcrawlers]

Características del WWW

- Demasiado grande

Es indispensable priorizar el orden de crawleo!

Queremos tener al menos una página de cada dominio

Queremos identificar primero las páginas más importantes.

Características del WWW

- Cambia constantemente

  25% de las ligas cambian cada semana [Ntoulas, Cho, Olston, 2004]

  En 1997 cada mes 600 GB de texto cambiaban [Kahle, 1997]

[webcrawlers]

Características del WWW

- Cambia constantemente

Es indispensable actualizar la información constantemente!

Equilibrio entre visita y revisita

Características del WWW

- No la conocemos

  Información con la que contamos al crawlear:

  - vínculos de salida, grado de salida
  - texto
  - url
  - fecha de la última visita

[webcrawlers]

No hay referencias acerca de la implementación de crawlers exitosos.

Cuestiones comerciales

Ocultar el algoritmo y el mecanismo de calificación (ranking) de páginas para evitar spamming. (!)

[webcrawlers]

Un poco de historia

RBSE (Eichman, 1994)

Primer crawler publicado

Arquitectura basada en dos componentes: el “spider” que actualiza la base de datos y el “mite” que baja los documentos y los almacena como documentos ASCII

World Wide Web Worm (Pinkerton, 1994)

Primer índice público de títulos y urls.
Funciona con grep.

[webcrawlers]

Un poco de historia

Internet Archive (Burner, 1997)

Almacena snapshots del web.
También guarda un registro histórico de hosts e IP.

WebSPHINX (Miller et al, 1998)

Librerías en Java
Primero en utilizar multithreading y HTML parsing

[webcrawlers]

Un poco de historia

Google Crawler (Bring y Page, 1998)

- Programado en C++ y Python.
- Sólo se conoce la arquitectura de una versión temprana
- El crawler y el sistema están integrados y utilizan una arquitectura centralizada

Ubicrawler (Boldi et al, 2004)

- Crawler distribuido programado en Java
- Todas las funciones están descentralizadas
- Reportan 660 páginas/s por cada CPU

[webcrawlers]

Implementación

- Características

  - Paralelo
  - Flexible
  - Bajo costo, alto rendimiento
  - Robusto
  - Escalable

Implementación

- Arquitectura de un crawler

  Aplicación

  Decide el orden de las páginas a recorrer en función de alguna estrategia de crawleo.
  Administra la base de datos.

  Sistema de crawleo (downloaders)

  Recibe url's de la aplicación, crea la conexión con la página, parsea el texto, recopila información, genera un registro y retroalimenta al sistema.

Políticas de amabilidad

Para optimizar recursos de red y evitar la saturación de servidores, hay que espaciar las solicitudes a un sitio.

Cho y Garcia-Molina proponen un intervalo de 10 segundos

WIRE Crawler de 15 segundos

Mercator considera que si tardó t segundos en bajar un documento, entonces espera 10*t segundos antes del siguiente

Estrategias de crawleo (sin historia)

Aleatorio

Caminante aleatorio
[Boldi et al, 2004] demostraron que es muy eficiente

Depth-First

El crawler escoge la siguiente página como la última que fue agregada a la lista de espera (LIFO).

[webcrawlers]

Estrategias de crawleo (sin historia)

Breath-First

Escoge primero las que fueron agregadas a la lista primero (FIFO)

BackLink Count

Recorre primero las páginas de la lista de espera que tienen más ligas de entrada.
[Cho y García-Molina. 1998]

[webcrawlers]

Estrategias de crawleo (sin historia)

Batch PageRank

También conocida como Quality-First
Ordena las páginas en la lista de espera en función de su calidad definida a partir de PageRank.

PartialPageRank: En función de una región de la red
PageRank: La red completa (!)

No hay convergencia entre PartialPageRank y PageRank [Boldi et al.]

[webcrawlers]

Estrategias de crawleo (sin historia)

OPIC

También conocida como Weighted Backlink Count

Todas las páginas tienen una cantidad de efectivo inicial (cash), al momento de entrar a una página lo distribuyen uniformemente entre todas las páginas a las que hacen referencia.
La prioridad de una página en la lista de espera es la cantidad de efectivo que tiene.
Esta estrategia es similar a PageRank, pero al no ser un método iterativo sobre una red estática, es mas eficiente.

Estrategias de crawleo (sin historia)

Larger-Sites First

La estrategia de este método se basa en no tener muchas páginas de un mismo sitio en la lista de espera.

Esta estrategia ha sido estudiada por [Baeza et al, 2005] y afirman que funciona muy bien.

Estrategias de crawleo (sin historia)

Focused crawling

Recorre las páginas en función de una busqueda focalizada a temas predefinidos.

En vez de recorrer toda la web, compara semánticamente el documento y los urls que encuentra y elige aquellos que son relevantes para su búsqueda.

Ignora grandes porciones del web que no son relevantes a su búsqueda.

Fue propuesto por [Chakrabarti et al, 1999]

Estrategias de crawleo (con historia)

Utilizan el PageRank de las páginas recorridas anteriormente para definir la prioridad de páginas en la lista de espera. Es decir, empiezan por las páginas de mayor PageRank.

Según [Cho y Adams, 2004] el error relativo al calcular PageRank con cuatro meses de anticipación es del 78% (!)

Se supone que la mayor parte de los crawlers importantes utilizan en algún nivel el PageRank histórico, pero existen muy pocas referencias acerca de su implementación.

¿Que estrategia utilizar?

Todas las estrategias de crawleo encuentran primero las páginas con mayor PageRank. La diferencia se basa en el costo computacional, la existencia o no de registros históricos y la rapidez con la que encuentran las páginas importantes.

Batch PageRank computacionalmente es muy caro

Breath-First funciona inclusive peor que el aleatorio

Las estrategias sin historia más eficientes son Larger-Sites First y OPIC [Baeza et al. 2005]

[webcrawlers]

Integridad y actualización de registros

Edad

La edad de una página p está dada por:

$E_p(t)$= 0 si p no ha sido modificada al tiempo t
$E_p(t)$= t-ultima modificación de p

Frescura

La frescura de una página p está dada por:

$F_p(t)$= 1 si p es igual a la copia almacenada
$F_p(t)$= 0 si no

Políticas de revisita

Uniforme

Revisita todas las páginas con a misma frecuencia independientemente de su tasa de cambio.

Proporcional

Revisita más las páginas que cambian mas frecuentemente. Es decir, la frecuencia de visita es proporcional al estimado de actualizaciones de la página.