Architecture of Web IR System

When I first began to cut up bullocks, I saw before me whole bullocks. After three years’ practice, I saw no more whole animals.

——《Chuang Tzu》

Yiqun Liu

Department of Computer Science and Technology
Tsinghua University
Architecture of Web IR System

• Our first glance at search engines
Architecture of Web IR System

- Our first glance at search engines
Architecture of Web IR System

- Search engine is much more than their simple user interfaces...

Google data center in Dalles, Oregon siphons off vast amounts of energy (1.8GW, about 1/12 of the Three Gorges Dam)

Google’s planning for a data center that would float out on the ocean
Architecture of Web IR System

- Wikipedia: System architecture is the conceptual model that defines the **structure**, **behavior**, and more views of **a system**.

- Answer.com: System architecture is the discipline that combines system elements which, **working together**, create unique **structural** and behavioral capabilities that none could produce alone.
Architecture of Web IR System
Outlines

• Data crawling subsystem
• Indexing subsystem
• Retrieval subsystem
• Hyperlink analysis subsystem
• Principles for search engine system designing
Data crawling

- Crawler, Spider, ...

- Major functions
  - Collect huge amount of high quality Web data timely and efficiently, and retain their hyperlink structure information as well
  - User requirements: huge amount, high quality, timely
  - System requirements: efficiently (storage, bandwidth)
  - Data: Web data (page, text, multimedia), hyperlink structure information
Data crawling

- Basic ideas

```
Spider (S)  // 输入种子集合 S 对应的 URL
{
    Get (S);  // 抓取 S 中的所有页面
    S' = Resolve (S);  // 提取 S 中各个页面包含的超链接 URL
    Spider(S');  // 以 S' 为种子继续进行抓取
}
```
Data crawling

• Crawling strategies
  – *Cumulative crawling*
    • Mainly adopted in index construction
    • Time consuming: days to months (1 month appr. for Baidu)
    • Crawling based on hyperlink structure
  – *Incremental crawling*
    • Mainly adopted in index updating
    • Time consuming: seconds to days (Google RT search)
    • Crawling based on hyperlink or APIs, focused crawling
• Performance Requirements: freshness

– Larry Page: Google should be scanning the entire Web every second

– Freshness of Web information:
  • long-term stable:
    – baike.baidu.com: 汶川县
  • relatively stable:
    – personal homepages, blogs, ...
  • instantly changing:
    – stock price, breaking news, ...
Data crawling

- Performance Requirements: Quantity
  - Index Size War

The number of pages (needed by users) will be bounded by the population. (*Mei et.al.*, WSDM2008)
Data crawling

• Performance Requirements: Quantity
  – Deep Web data mining
  • Baidu: Aladdin project
    – http://open.baidu.com/
• **Performance Requirements: Quality**

  – Too many low quality or even spam pages ...
Data crawling

• Performance Requirements: effectiveness
  – Assuming that:
    • within a certain time period $T$, the amount of Web resources that meet users' requirement is $S$.
    • The bandwidth required by data crawling subsystem ($B$) should satisfy
      $$ B > \frac{S}{T} $$

      $$ B' = \frac{S'}{T} = \frac{S / R}{T} = \frac{S}{T \cdot R} $$

      $$ B'' = \frac{B'}{U} = \frac{S}{T \cdot U \cdot R} $$

      R is the percentage of high quality Web resources

      Usage of bandwidth is U
- Performance Requirements: effectiveness
  - How to improve $R$?
- Focus on high quality resources
• Performance Requirements: effectiveness

  – How to improve $U$?

• Coordinate with information suppliers

[摘自http://tb.donews.net/TrackBack.aspx?PostId=866456]

wiki.donews.com连续几天，定时被百度的抓取机器人抓到系统停止响应。

拜托百度，不要这样抓内容了。就算抓，也应该用1个线程来抓，只抓更新的内容，何必每天抓一次，而且用无数个线程，而且每次都要抓全部内容，还不放过任何wiki的历史页面，甚至连错误信息都要原样搬走。

这种抓取方法，谁受得了？DOS attack？Robots.txt
Indexing

• Human memory v.s. search index
  – Sensory memory (cache?) v.s. Short-term memory v.s. Long-term memory
    • Long-term memory is intended for permanently storing, managing, and retrieving of information for later use.
    • Retrieval of information in our long-term memory requires “cues”
    • Brain try to organize information with “cues”
  – Search engine try to organize information with index terms (as cues)
Indexing

• Inverted index
  – Key = term; Value = a list of (document, position)
  – Stores a list of documents for each term

<table>
<thead>
<tr>
<th>Term 1</th>
<th>Doc 1, pos 1</th>
<th>Doc 1, pos 2</th>
<th>…</th>
<th>Doc p, pos q</th>
</tr>
</thead>
<tbody>
<tr>
<td>Term 2</td>
<td>Doc 1’, pos 1’</td>
<td>Doc 1’, pos 2’</td>
<td>…</td>
<td>Doc p’, pos q’</td>
</tr>
<tr>
<td>…</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Term N</td>
<td>Doc 1^(n), pos 1^(n)</td>
<td>Doc 1^(n), pos 2^(n)</td>
<td>…</td>
<td>Doc p^(n), pos q^(n)</td>
</tr>
</tbody>
</table>

  – Implementation: Hash table, Suffix trees, ...
  – Mrs. Clarke spent 16 years to index “the complete works of Shakespeare” in the middle 1800s
# Indexing

- **Forward index**
  - Key = document, value = a list of (term, position)
  - Stores a list of words for each document

<table>
<thead>
<tr>
<th>Doc 1</th>
<th>Term 1, pos 1</th>
<th>Term 1, pos 2</th>
<th>…</th>
<th>Term p, pos q</th>
</tr>
</thead>
<tbody>
<tr>
<td>Doc 2</td>
<td>Term 1’, pos 1’</td>
<td>Term 1’, pos 2’</td>
<td>…</td>
<td>Term p’, pos q’</td>
</tr>
<tr>
<td>…</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Doc N</td>
<td>Term 1(^{(n)}), pos 1(^{(n)})</td>
<td>Term 1(^{(n)}), pos 2(^{(n)})</td>
<td>…</td>
<td>Term p(^{(n)}), pos q(^{(n)})</td>
</tr>
</tbody>
</table>

- Usage of forward index in search engine system:
  - Document parsing, Snippet generation
  - Transformed into inverted index
Indexing

• Major functions
  – Transformation from content to set of terms
  – Indexing/looking up information effectively
  – Information storage with limited system resources
Indexing

- Transformation from content to set of terms
  - English and other Latin languages
    - Text is segmented by spaces
    - Words don’t equal to “terms
      - Proper Nouns: Los Angeles, Peter the Great
      - Terms including punctuations: O’Neill, C++, C#, B-52
  - Chinese (Japanese, other language with characters)
    - Text should be processed by segmentation algorithms
    - Full-width/Half-width: \(1 = 1 = 一 = 壹\)
    - Encodings: UTF-8, GBK, BIG5
Indexing

• Indexing/looking up information
  – Huge amount of data to be indexed
    • Performance requirements: look up speed, index size, fault tolerance, ...
  – One important fact: search engine visits indexing terms at extremely unbalanced frequencies.
    • Hot terms: car, photo, china, ...
    • Cold terms: Yiqun Liu, 138xxxxx0265, ...
    • 5% data can meet 92% user requests
    • Hierarchical index structure (like memory hierarchy)
Indexing

- Information storage with limited system resources
  - Disk I/O is a bottleneck for search efficiency
  - Query processing, electricity power, disk failure, ...
- How to avoid unnecessary disk I/O?
  - record only the most necessary information

<table>
<thead>
<tr>
<th>Term 1</th>
<th>Doc 1, pos 1</th>
<th>Doc 1, pos 2</th>
<th>…</th>
<th>Doc p, pos q</th>
</tr>
</thead>
<tbody>
<tr>
<td>…</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Term N</td>
<td>Doc 1(^{(n)}), pos 1(^{(n)})</td>
<td>Doc 1(^{(n)}), pos 2(^{(n)})</td>
<td>…</td>
<td>Doc p(^{(n)}), pos q(^{(n)})</td>
</tr>
</tbody>
</table>

- Avoid redundancy in position information (with N bits, may cause information losses)
- Adoption of index compression
Retrieval

• Major functions
  – Content retrieval based on similarity
  – Understanding users’ information needs

• Information Retrieval
  – Coined by Calvin Mooers in 1950
  – Boosted by TREC organized by NIST and DARPA
  – Relevance = (Query, Document, Collection)
Retrieval

• Traditional IR system vs. search engine system
  – Traditional IR system:
    • Relevance = (Query, Document, Collection)
  – Search engine system
    • Relevance = (Query, User, Document, Collection)
  – IR v.s. SE: user
    • library experts v.s. common netizens
    • different user, same/different information needs
      – information retrieval: definitions? books? courses?
Retrieval

• IR v.s. SE: collection

Right design at X may be very wrong at 10X or 100X
Jeff Dean, Google Fellow
Retrieval

• IR v.s. SE: relevance

Relevant

Relevant

Relevant

Not Relevant

Not Relevant
Retrieval

• Ranking factors
  – IR system: content similarity
  – Search: many more
    • SEO site: 100+
    • Yahoo LTR task: 700+
    • link structure, content, user click, page structure, freshness, stability, ...
Retrieval

• Understanding users’ information needs
  – How search engine interact with users?
    • Keyword based query + selectively navigation
    • A reasonable solution, yet cause the problem of understanding users’ information needs
  – Ambiguity in query content
    • Apple: computer? music player? fruit? movie?
    • http://clusty.com
Retrieval

- Understanding users’ information needs
  - TREC 2004 query type identification task

Best result:
Precision = 61.3%
Retrieval

• Understanding users’ information needs
  – Analysis into user previous query/click behaviors
  • How did users interact with the result list?
  • Query: Tsinghua University
    – www.tsinghua.edu.cn
    – news.tsinghua.edu.cn
    – www.join-tsinghua.edu.cn
  • Query: Guo Degang (a famous cross-talk star)
    – http://baike.baidu.com/view/5444.htm
    – http://blog.sina.com.cn/guodegang
Hyperlink analysis

• Major functions
  – Quality estimation based on hyperlink analysis
  – Extending Web page descriptions
  <a href="http://100.tsinghua.edu.cn" target="_blank">Tsinghua Century Anniversary</a>
Hyperlink analysis

• Quality estimation based on hyperlink analysis
  – Oriented from literature citation analysis (SCI, EI)
  – Key algorithms for search engines
    • PageRank: Page and Brin, 1998
    • HITS (Hyperlink-Induced Topic Search): Kleinberg, 1998

Recommendation

A -> B

Topic related

A -> B
Hyperlink analysis

• Quality estimation based on hyperlink analysis
  – PageRank: quality estimation for the entire Web
    • Random walk model
    • $\text{PageRank}(p) = \text{PR(a random user visits } p)$
  – HITS: quality estimation for search results
    • Authority value v.s. Hub value
    • Web page with high hub value usually links to Web pages with high authority values
    • Web page with high authority value is usually linked by Web pages with high hub values
Hyperlink analysis

• Quality estimation based on hyperlink analysis
  – Challenge: how to deal with newly-appeared pages
    • freshness v.s. quality
    • whether pages can inherit quality scores from its sites?
  – Challenge: how to avoid noises in hyperlinks
Hyperlink analysis

• Extending Web page descriptions
  – Web page content: described by page authors
  – Search request: described by search users
  – Even for a same object, descriptions may be different
    • There is no word “北大” (PKU) on http://www.pku.edu.cn/
    • Whether we can get PKU homepage for the query “北大”?
Hyperlink analysis

• Extending Web page descriptions
  – Anchor text: text adopted to describe hyperlinks
  – The description is for the page it is linked to instead of the page it is located.

```
<A href="http://www.pku.edu.cn">PKU</A>
```
Hyperlink analysis

- Extending Web page descriptions
  - Descriptions don’t match
  - Wisdom of the crowds
Hyperlink analysis

• Extending Web page descriptions
  – Challenge: how to combine anchor text with original page content
    • Data preprocess: send anchor texts to its target pages.
    • which is more important?
  – Challenge: noises in anchor texts
    • Similar with hyperlinks, anchor texts are added by page authors without supervision
    • Relatively more reliable than hyperlink data
    • Amount and quality of anchor text sources
Principles for search engine designing

- Major functions and performance requirements
- What are the common designing principles?
Principles for search engine designing

- User-oriented
  - What data / Which Web pages should be crawled?
  - How often should a certain Web page be updated?
  - Which pages should be placed in a higher level index?
  - What is the information need of the query?
  - Which pages are more preferred by search users?
Principles for search engine designing

• Lossy optimization:
  – Limited resources should be focused on important and necessary aspects.
  – Some Web pages should be updated more frequently than others
  – Some Web page are stored in index servers with more powerful hardware
  – Position information for some words in inverted index may be not accurate
Principles for search engine designing

- Efficiency comes first
  - Crawling: save bandwidth, save money
  - Indexing: save disk I/O, save money
  - Retrieval: complicated NLP/semantic analysis algorithms shouldn’t be adopted
  - Hyperlink analysis: only offline algorithms should be considered
Principles for search engine designing

• Scalability
  – Indexing: newly-appeared words/pages
  – Retrieval: newly-adopted ranking factors
  – Hyperlink analysis: newly-appeared hyperlink data
Thank you!

Questions or comments?