Predicting Web Actions from HTML Content

By Brian D. Davison
June 2002

Presented by
Hania Abd-El-Razik & Sidrah Laldin

Abstracted Conclusion

• Research looked at how well a user’s next action can be predicted based on the content of recently visited pages. Infinite cache combined with predicting the most ‘similar’ link gave the best results.

Motivating Goal

• “to accurately predict the next request that an individual user is likely to make”

Sections

1. Introduction
2. Concepts
3. Experimental Methodology & Results
4. Issues
5. Implementations
6. Research Contributions
7. Author
Assertion

• “Unfortunately, for most people, retrieval times for many … documents are well above the threshold for perception”
  (p. 159)

Assertion : Deconstruction

• The threshold for perception is 10 seconds
  – Pages should take no longer to load
  (Designing Web Usability, p. 44)

Assertion : Deconstruction

• 813 million internet users worldwide
  (§)

Assertion : Deconstruction

• 124 million using broadband (15%)
  – Over 50% of users in North America on broadband
  – Most people are still on dial-up

http://www.internetworldstats.com/stats.htm
Web Caching

• The temporary storage of web objects for later retrieval

[http://www.web-caching.com/]

Why Use Web Caches?

• Two main reasons:
  1. reduce latency — user feels like pages load faster
  2. reduce traffic — reduces bandwidth usage

Web Caching

• Two kinds:
  – Browser cache:
    • implemented through browser, objects saved on user’s computer
  – Proxy cache:
    • objects saved on a proxy server (e.g. an intranet) which is closer to the client; shared by many users
    – proxy : A person authorized to act for another; an agent or substitute. (American Heritage Dictionary)

Non-Cacheable Resources

• Dynamic pages (e.g., .php, .asp, 28% in exp’t)
• Secure pages (eg. over SSL, https)
• Streaming video, audio
• Sites using IP authentication (eg. ACM website)

HTTP/1.1 200 OK
Date: Fri, 30 Oct 1998 13:19:41
GMT Server: Apache/1.3.3 (Unix)
Cache-Control: no-cache
Definition: Link Prefetching

- Using idle time to download pages that the system thinks the user might choose
  - If the user does select that link, the document can be retrieved faster
- Cost of reduced latency is increased network traffic

Definition: “Thinking-Time”

- The time available to choose and prefetch a link before the user makes another selection
  - “typically less than one minute”

Prefetchable Resources

- Those considered likely to be cacheable
- Only about 25% of resources

Prediction

- Two main types:
  1. History-based
  2. Content-based
History-Based Prediction

- User model is based on previous visits
  - If faced with the same choice again, user will likely make the same decision

Content-Based Prediction

- Models user’s interests on content of current and recently visited pages
  - Determines which links are most similar based on anchor text

Premise

- “anchor text [is] a reasonable descriptor for the page to which it [points]”

  √ For background information on the blue-nosed honey-bee, click here.
  ✓ We have additional background information about the blue-nosed honeybee.
    (Designing Web Usability, p. 55)

Experiment

- Computer science students
  (problems??)

- Distinguish clients based on IP address

- Dynamic IP addresses seen as distinct users on each session
Experiment - Methodology

• Log does not reflect the requests made by users that are satisfied by browser caches. (why would that be helpful?)

• Create a dataset (eg. d1, d2, d3, d4, d5) and predict d6.

Experiments - Methodology

• Retain responses for some error codes (eg. 301, 302) but eliminate others (400s, 500s)

• Filter URL for the more popular advertising hosting services. (Potential drawbacks??)

Experiments - Methodology

• Content text of the page includes non-HTML text and title

• The basic representation of each textual item is “bag-of-terms” with term frequency.

Definition: Bag-of-Words

• hash table that stores each word and its associated frequency
Experiment–Prediction Methods

- Two simple methods
  - Baseline random ordering
    - Links selected randomly to be pre-fetched
  - Original rank ordering
    - First link on the page to be prefetched

- Two other methods
  - Similarity
  - Cumulative

Experiment–Prediction Methods

A metric to compare between two documents:

\[
\text{Text-Similarity}(D_1, D_2) = HTML[2*2] + \text{responses}[2*1] + \text{would}[1*2] + \text{prediction}[1*0]
\]

Experiment–Prediction Methods

- Test the addition of terms before and after the anchor. (Allows to evaluate context).

The `<a href="http://www.acm.org/acm1/">ACM</a>` Conference will take you beyond cyberspace.

- 0 additional terms: {ACM}
- 5 additional terms: {The, ACM, Conference, will, take, you, beyond}

- Not allow overlapping text

Experiments-Predictive Methods

Similarity

Running "Total" document:

- HTML
- great

New Document

- HTML
- is great!
- HTML
- is not so.

Let’s visit Google

Link to same Page, but first link used.

Because links to the same page, use the link that has anchor more similar to the previous document.
**Experiments-Predictive Methods**

**Experiment-Evaluation Scheme**

- Evaluate predictive accuracy of ranking methods – three cases tested
  1. Evaluate algorithms that use their best 1, 3, or 5 predictions. (Immediate prediction w/o inf. cache)
  2. Consider the case with infinite cache to hold prefetched predictions. (with 1, 3, or 5 predictions)
  3. Case with Cache only

---

**Experiment-Evaluation Scheme**

- Prediction set “correct” if one of them becomes a requested page.
- Not predictable pages include:
  - Form entries
  - Dynamically generated pages.
- “Free” pages
  - used to show lower bound
  - Number of links in page is no larger than number of predictions permitted

---

**Experiments-Results**
Methodology Issues

• Is Infinite cache a reasonable assumption?

• Research doesn’t consider thinking-time or prefetching costs

Methodology Issues

• JavaScript links excluded
• Image alternate text and meta content tags not considered
  – <meta content="...">
  – <img alt="...">

• Ads manually filtered out

Prefetching Issues

• Prefetching may compete for bandwidth
  – with other computers on network
  – with internet-accessing applications
  – with multiple browser windows
• Lower cost often trumps speed if bandwidth is billed at a flat rate
• Can prefetching be used to download potentially malicious code?

Prefetching: Research Prototypes

• WebCompanion – loads the page that would take longest
• Wcol – load first $n$ images, $m$ documents
Prefetching: Implementations

- Mozilla browsers support limited link prefetching:
  - `<link rel="prefetch" href="1.html">`
  - `<link rel="next" href="2.html">`
- Author & others implemented a Mozilla module with content & history-based prediction

Contribution of Research

- Prediction relies on content of previous and current pages
  - Reflects changing user interests
  - useful for new page views
- “does not noticeably interfere with user experience”—“invisibly improves perceived performance”

Author: Brian D. Davison, Ph.D.

- Assistant Professor, College of Engineering and Applied Science Lehigh University
- Research:
  - Caching, Measurement, and Performance of WWW
- Seems like the main authority on caching & prefetching

References

8. [http://www.geog.ucl.ac.uk/casa/martin/atlas/geographic.html](http://www.geog.ucl.ac.uk/casa/martin/atlas/geographic.html) [very cool maps]