Usage Aware Average-Clicks

Kalyan Beemanapalli – University of Minnesota

Ramya Rangarajan – University of Minnesota

Jaideep Srivastava – University of Minnesota

Presenter:
Kalyan Beemanapalli

Outline

- Introduction
- Related Work
- Background
- Method
- Experiments and Results
- Key Contributions
- Conclusions and Future Work
- Questions

Related Work – Link Analysis

- Applications
  - PageRank
  - HITS
  - Average-Clicks (*Matsuo et al*)
- Disadvantage
  - Static

Related work

- Solution
  - Usage Data
- Why Usage Aware Average-Clicks?
  - Average-Clicks
    - Fairly new algorithm
    - Proposes a new definition to distance between web pages
    - Measures distance in user’s context
  - Ideas from
    - Usage Aware PageRank (*Oztekin et al*)
    - Extensions to HITS (*Miller et al*)

Average-Clicks

- Measure of distance between web pages
- Definition – An average click is *one* click among *n* links
- Probability of a random surfer on a page p to click any one of the links is

$$\alpha / OutDegree(p),$$

where α = Damping Factor

Average Clicks

- Average Click length of links on page *p* =

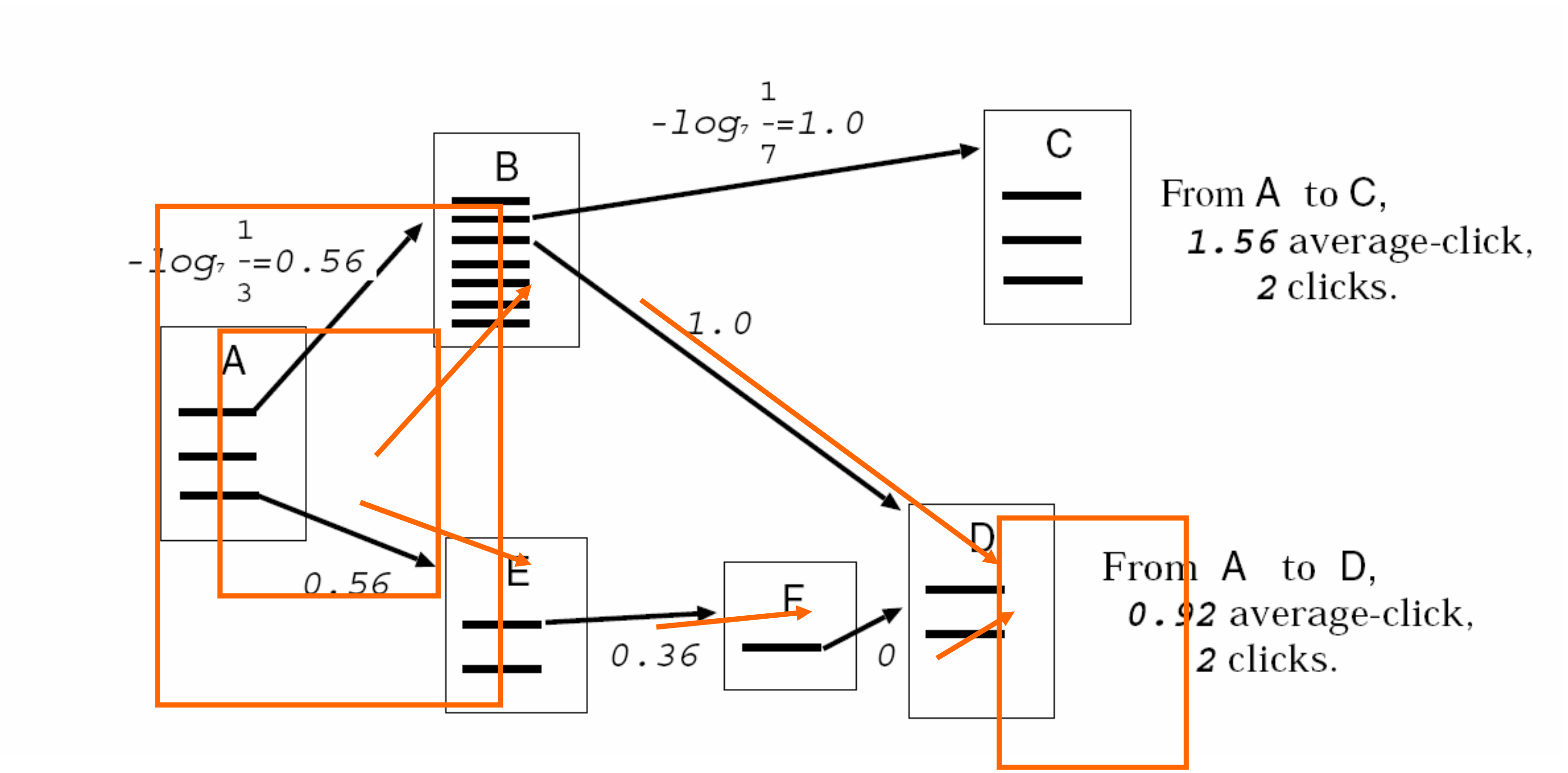
$$-log_n(\alpha / OutDegree(p)).$$

Where α = Damping Factor, n = Average Number of links on a page

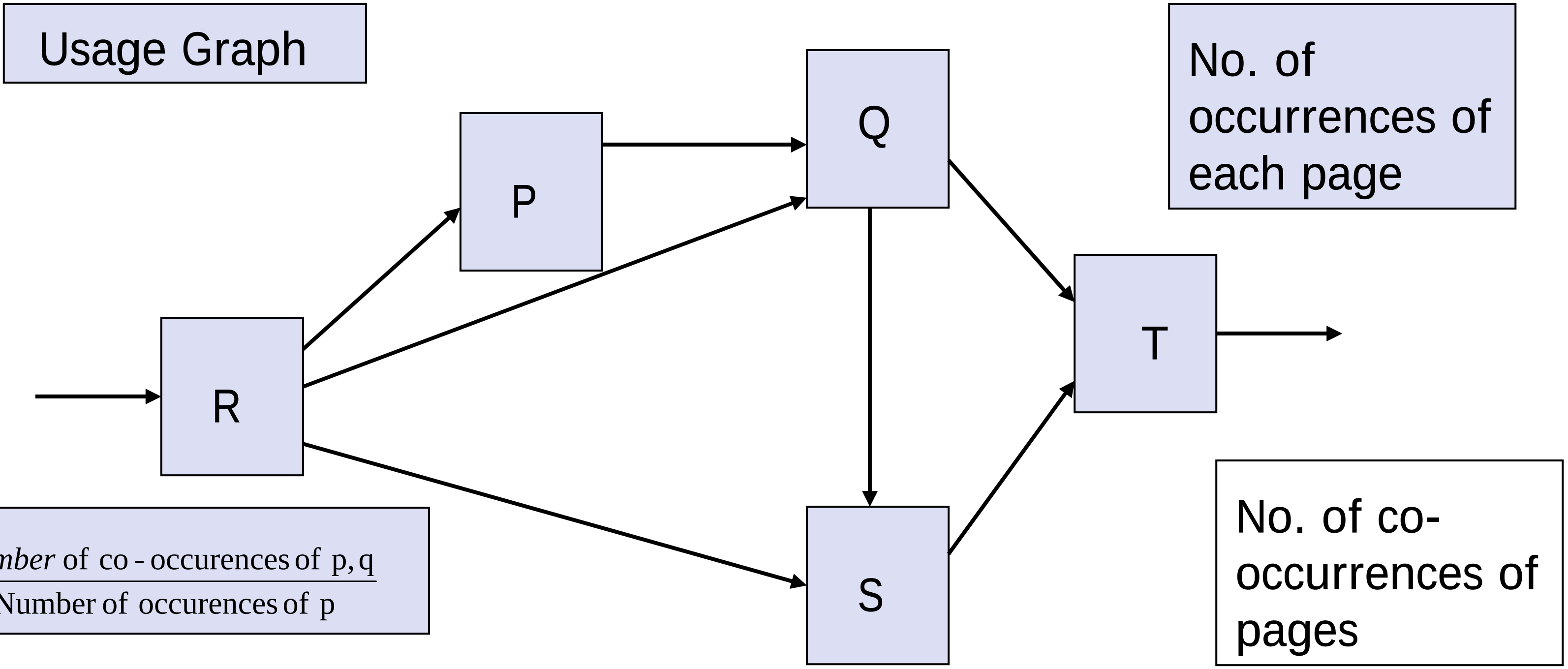
Distance between page *p* and *q*

- shortest path between the nodes representing the pages in the graph
- Path through a longer chain of links can be considered shorter than one through smaller number of links

Average Clicks - Example



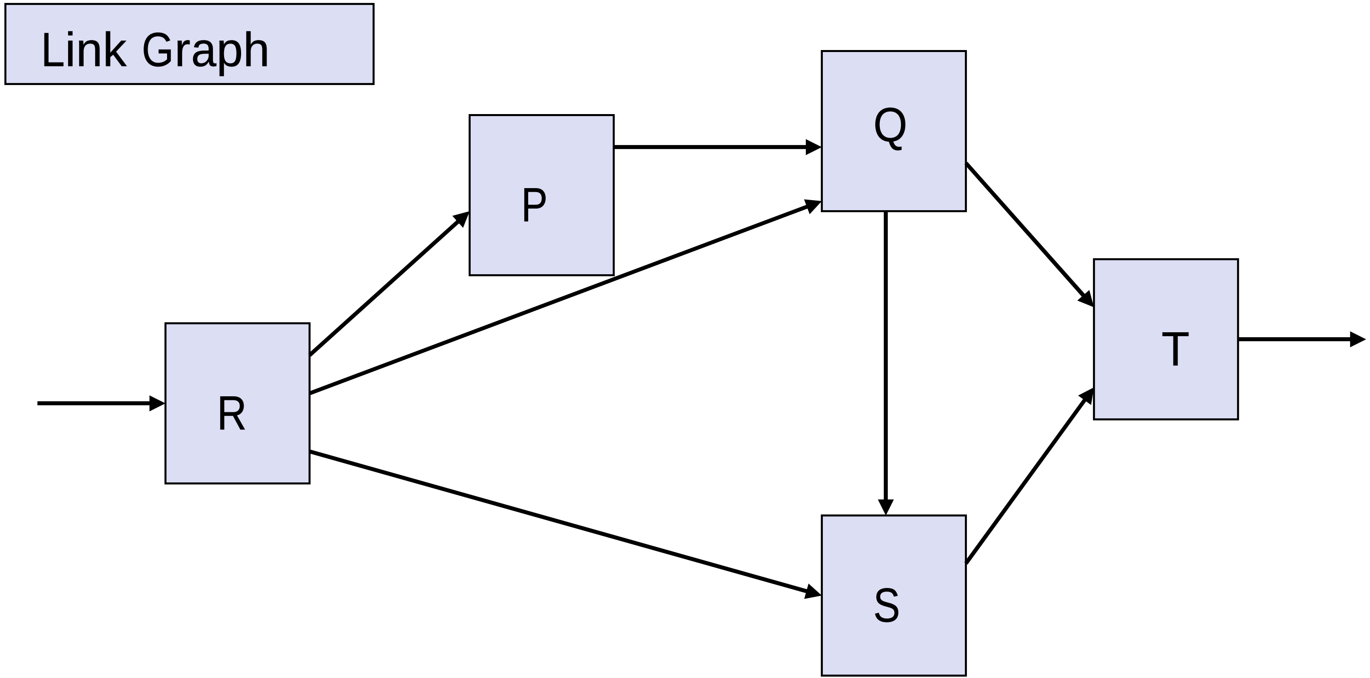
Usage Aware Average-Clicks


Usage Graph
P
Q
R
S
T
No. of occurrences of each page
No. of co-occurrences of pages

$$C(p,q) = \frac{\text{Number of co-occurences of p, q}}{\text{Number of occurences of p}}$$

$$C(p,q) = \frac{\textit{Weight of the edge from p to q}}{\textit{Weight assigned to node p}}$$

Usage Aware Average-Clicks



$$D(i, j) = \begin{cases} 1/\text{Outdegree(page i)} & \text{if there is a link to page j on page i} \\ \infty & \text{otherwise} \end{cases}$$

Usage Aware Average-Clicks

- We now have

$$C(p,q) = \frac{\textit{Number}\text{ of co-occurences of p,q}}{\text{Number of occurences of p}}$$

$$\begin{aligned} D(p,q) &= (\text{1/Outdegree(page p)) if there is a link to page j on page i} \\ &\quad \infty \text{ otherwise} \end{aligned}$$

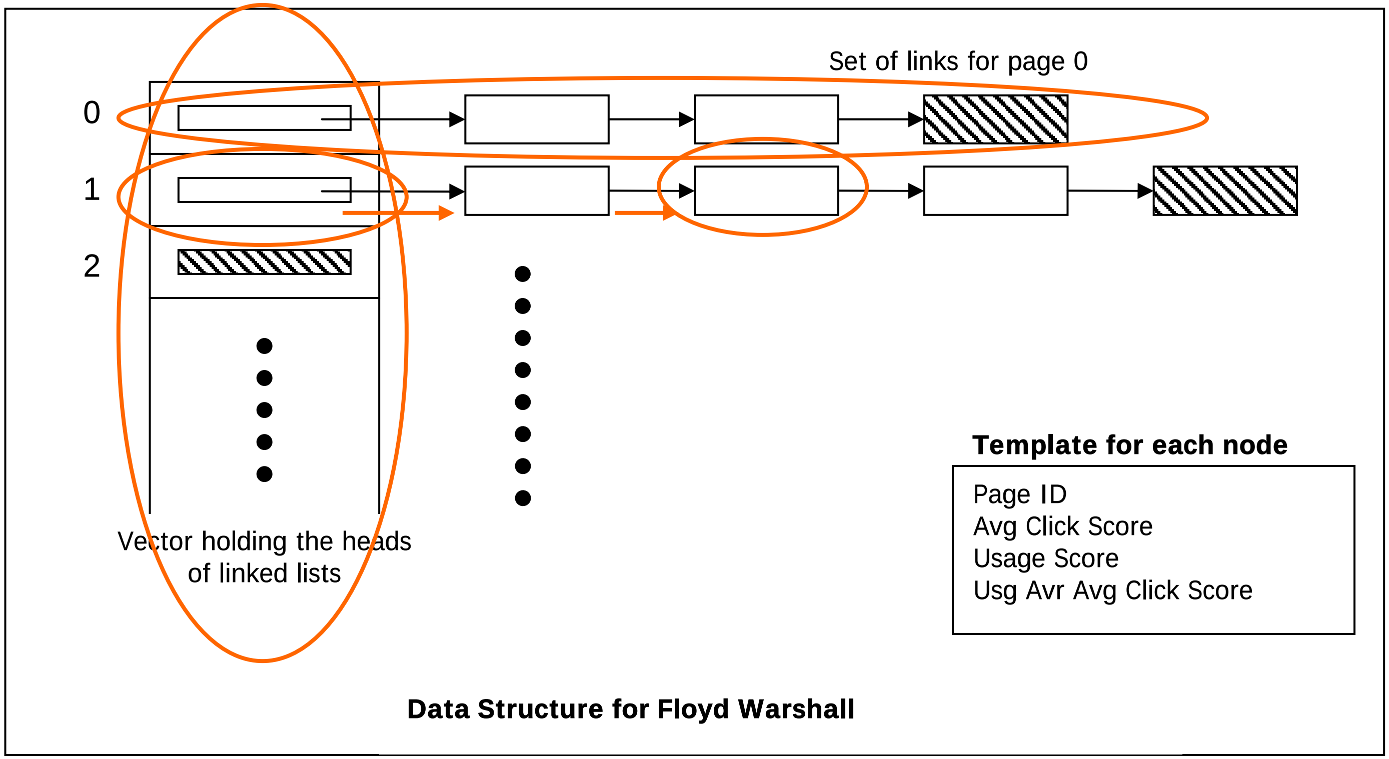
- We combine the Link Matrix and Usage Matrix to define the new definition of distance between 2 pages as follows:

$$Distance(p,q) = (1 - C(p,q)) * (-\log_n\left(\alpha / Out\deg ree(p)\right)$$

Usage Aware Average-Clicks

- Shortest distance between pairs of nodes – all pairs shortest path algorithm
- All Pairs Shortest path algorithm used – Floyd Warshall’s Algorithm
- Implementation Issues
  - Poor scalability

Solution



Template for each node

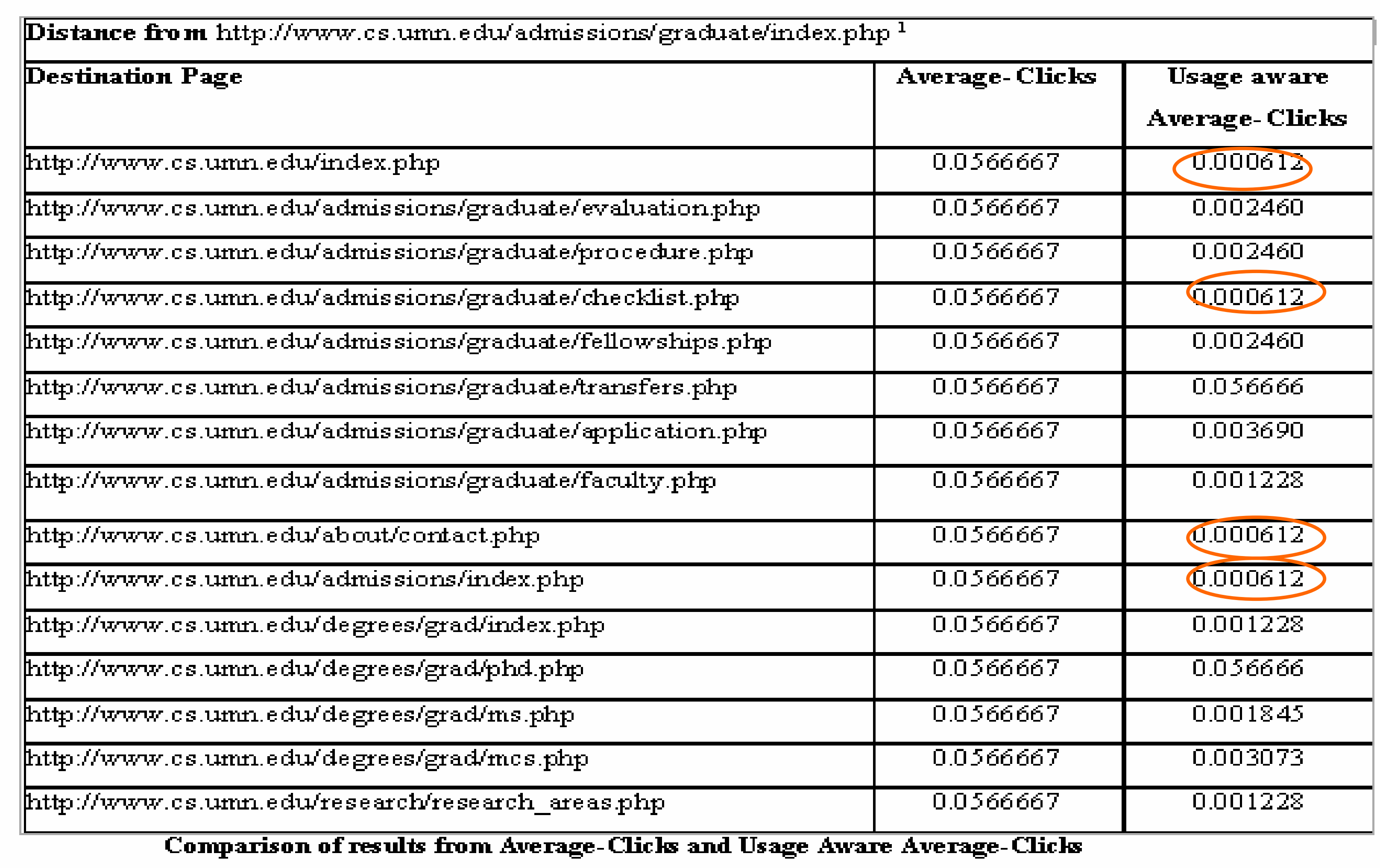
Page ID
Avg Click Score
Usage Score
Usg Avr Avg Click Score

Data Structure for Floyd Warshall

Experimental Results

- Experiments conducted on www.cs.umn.edu
- Usage data collected in Apr 2006
- Data set reduced to 100,000 sessions
- Noise removed
- Link Graph built using our crawler

Example Distances



Distance from http://www.cs.umn.edu/admissions/graduate/index.php [1]

Destination Page	**Average-Clicks**	**Usage aware Average-Clicks**
http://www.cs.umn.edu/index.php	0.0566667	0.000612
http://www.cs.umn.edu/admissions/graduate/evaluation.php	0.0566667	0.002460
http://www.cs.umn.edu/admissions/graduate/procedure.php	0.0566667	0.002460
http://www.cs.umn.edu/admissions/graduate/checklist.php	0.0566667	0.000612
http://www.cs.umn.edu/admissions/graduate/fellowships.php	0.0566667	0.002460
http://www.cs.umn.edu/admissions/graduate/transfers.php	0.0566667	0.056666
http://www.cs.umn.edu/admissions/graduate/application.php	0.0566667	0.003690
http://www.cs.umn.edu/admissions/graduate/faculty.php	0.0566667	0.001228
http://www.cs.umn.edu/about/contact.php	0.0566667	0.000612
http://www.cs.umn.edu/admissions/index.php	0.0566667	0.000612
http://www.cs.umn.edu/degrees/grad/index.php	0.0566667	0.001228
http://www.cs.umn.edu/degrees/grad/phd.php	0.0566667	0.056666
http://www.cs.umn.edu/degrees/grad/ms.php	0.0566667	0.001845
http://www.cs.umn.edu/degrees/grad/mcs.php	0.0566667	0.003073
http://www.cs.umn.edu/research/research_areas.php	0.0566667	0.001228

Comparison of results from Average-Clicks and Usage Aware Average-Clicks

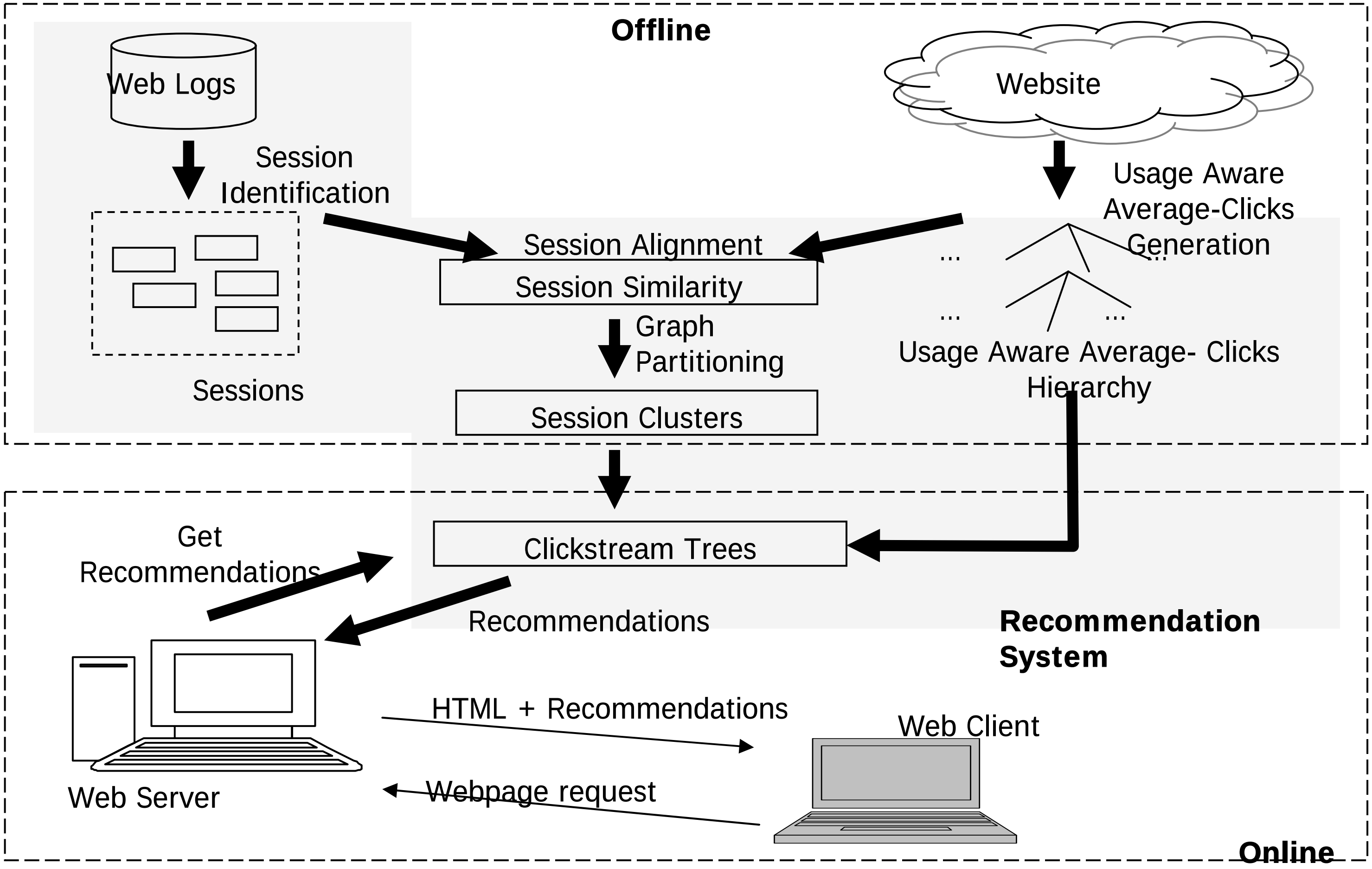
Evaluation Methodology

- Domain Expert's View
  - Questionnaires
- User's View
  - Questionnaires
  - Automate verification
- Our Method
  - Predicting Power

Evaluation Methodology

- Incorporated into a recommender system
- Idea - pages that are close to each other are more similar to each other than pages that are farther apart
- Performance compared with ‘2, -1’ model
- Tested on www.cs.umn.edu

The Recommender System Architecture



Evaluation Measures

- *Hit Ratio (HR)*: Percentage of *hits*. If a recommended page is actually requested later in the session, we declare a hit.

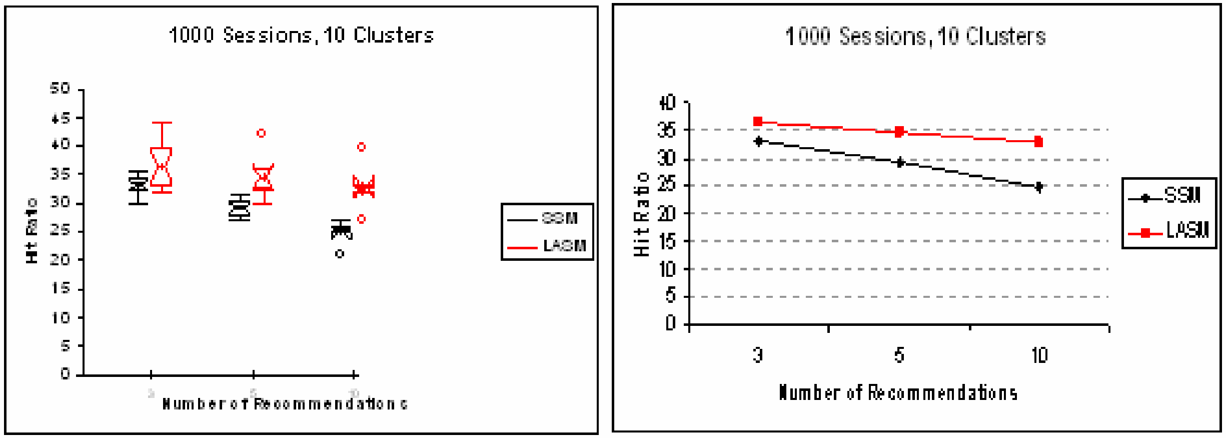
- *Click Reduction (CR)*: For a test session *(p1, p2,…, pi…, pj…, pn)*, if *pj* is recommended at page *pi*, and *pj* is subsequently accessed in the session, then the click reduction due to this recommendation is,

$$Click\ \ reduction\ = \frac{j-i}{i}$$

Experimental Set-up

- 1000 training sessions
- 3, 5, 10 recommendations
- 10, 15 and 20 ClickStream Clusters
- Different testing sessions
- Experiment repeated 5 times using different training set
- Results compared against the ‘2, -1’ model
- T-tests performed
- Same procedure for 3000 training sessions

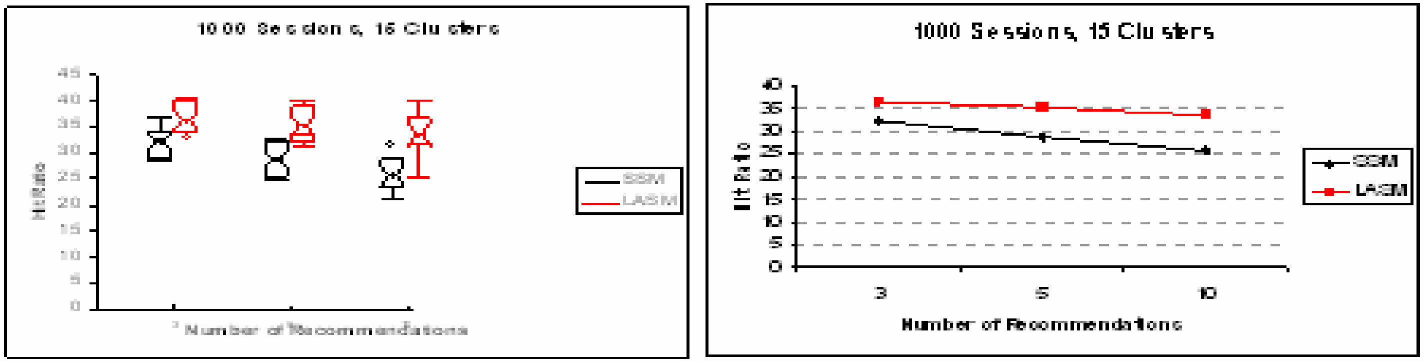
Results



Hit Ratio Vs No. of Recommendations for 1000 sessions, 10 clusters

Recommendations	3	5	10
p value	0.123242	0.030262	0.006292

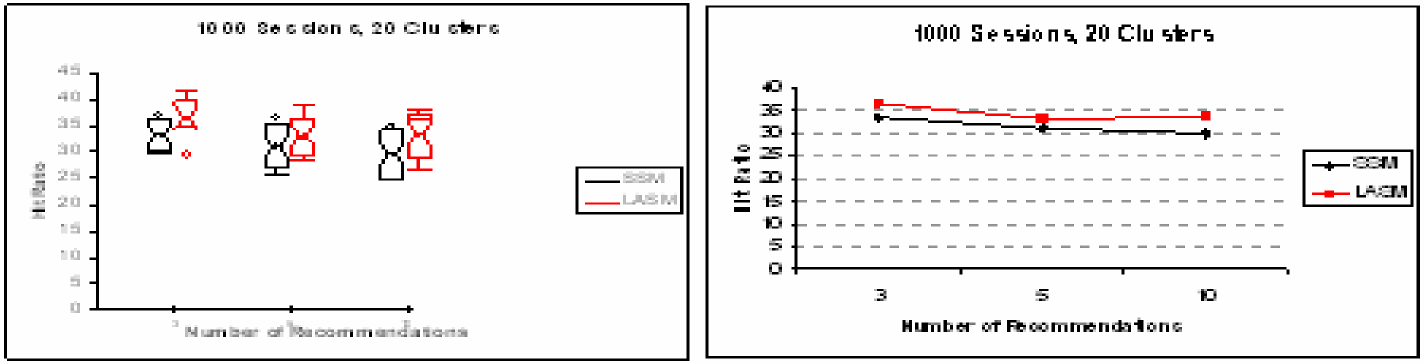
t-test scores for 1000 sessions, 10 clusters



Hit Ratio Vs No. of Recommendations for 1000 sessions, 15 clusters

Recommendations	3	5	10
p value	0.053543	0.014464	0.020082

t-test scores for 1000 sessions, 15 clusters

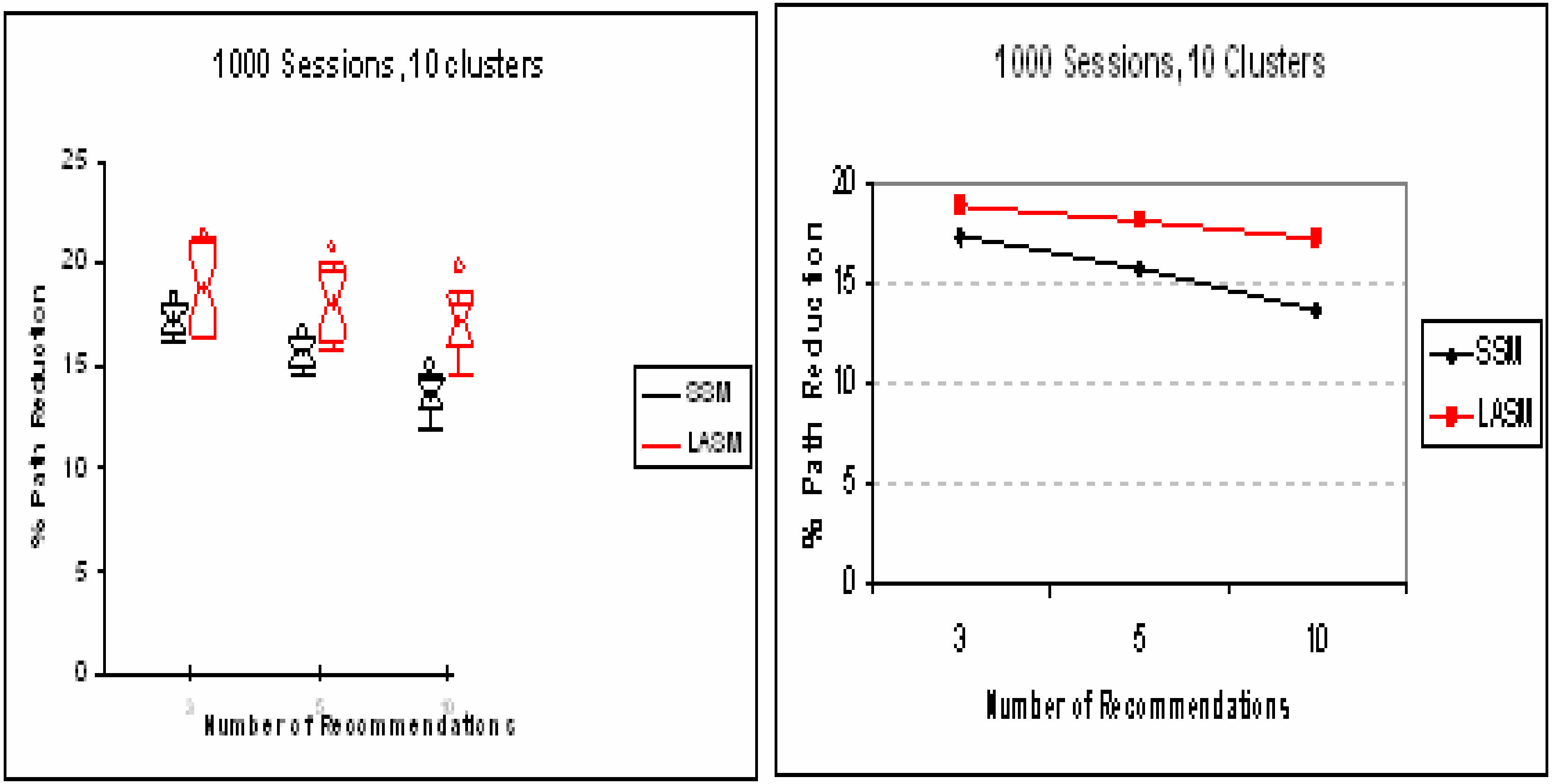


Hit Ratio Vs No. of Recommendations for 1000 sessions, 20 clusters

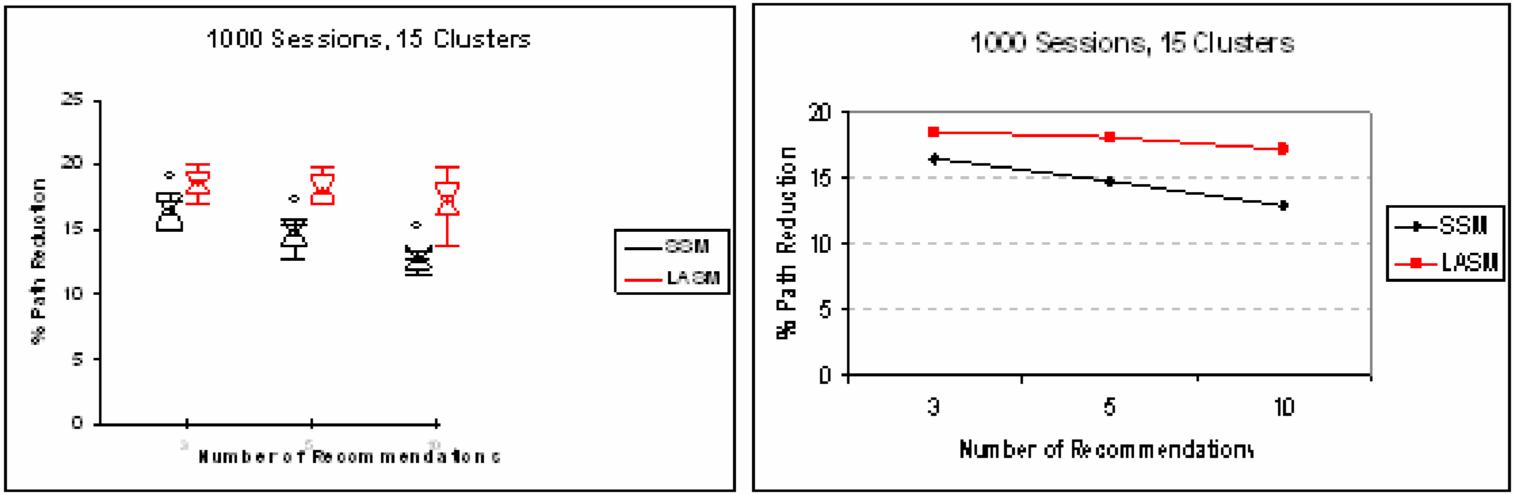
Recommendations	3	5	10
p value	0.04985	0.224891	0.125186

t-test scores for 1000 sessions, 20 clusters

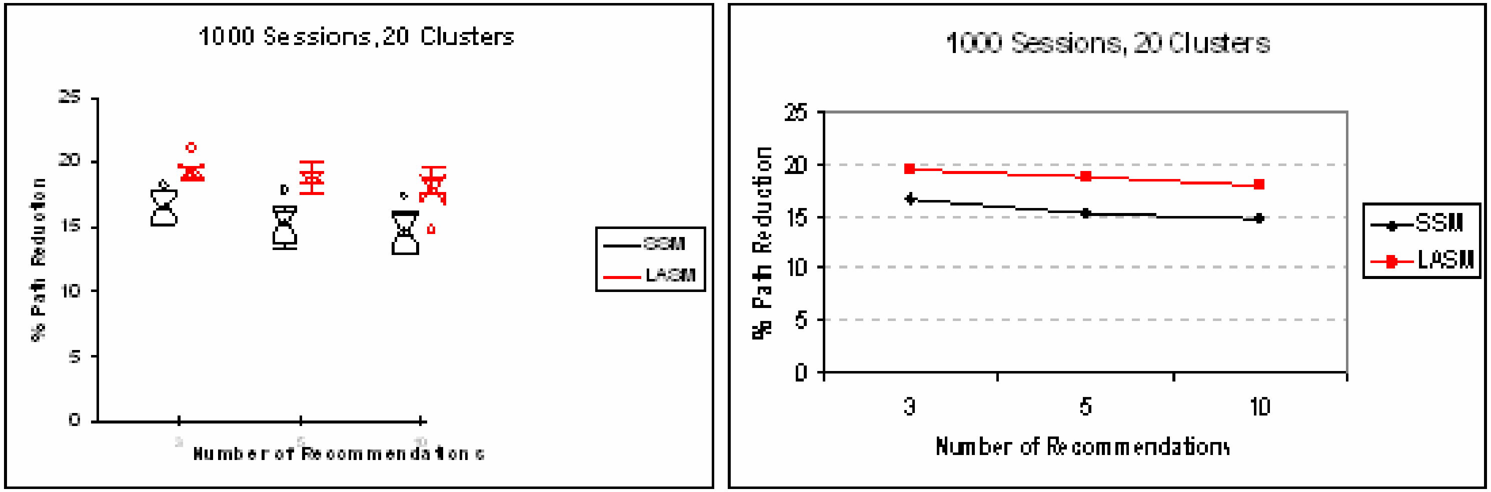
% Path Reduction



% Path Reduction Vs No. of Recommendations for 1000 sessions, 10 clusters



% Path Reduction Vs No. of Recommendations for 1000 sessions, 15 clusters



% Path Reduction Vs No. of Recommendations for 1000 sessions, 20 clusters

Conclusion

- Incorporated usage data into Average Clicks algorithm.
- Proposed a distance model using usage data and link graph
- Used this method to calculate the similarity between the pages in an intranet domain
- Showed that using a combination of web graph and link graph will provide better recommendations

Future Work

- Validate the algorithm using various testing methods like
  - Domain expert testing
  - User's perspective
- Compare the algorithm against other usage based link analysis algorithms
- Compare the quality of recommendations with those obtained by using other kinds of domain information

Questions