Lecture 5: Sponsored Search

ISM293
University of California, Santa Cruz
Spring 2009
Instructors: Ram Akella, Andrei Broder, and Vanja Josifovski
Questions about last lecture?

- We welcome questions & suggestions about all aspects of the course!
- E-mail to Ism293-group@soe.ucsc.edu
Sponsored Search
Disclaimers

- This talk presents the opinions of the authors. It does not necessarily reflect the views of Yahoo! Inc or any other entity.
- Algorithms, techniques, features, etc mentioned here might or might not be in use by Yahoo! Or any other company.
Lecture Content (6 hrs)

- Overview of Sponsored Search Advertising
- Anatomy of textual ads
- Web queries
- Ad Retrieval
  - Exact Match
  - Broad Match
- Broad Match
  - Query rewriting for Broad Match
  - Search-based Broad Match
- Using click data for Retrieval and Reordering
- Bid phrase suggestion
Sponsored Search Basics
Landing page
The general interaction picture: Publishers, Advertisers, Users, & “Ad agency”

- Each actor has its own goal (more later)
The simplified picture for sponsored search

- All major search engines (Google, MSN, Yahoo!) are simultaneously
  1. search results provider
  2. ad agency (Homework Lecture 3!!)

- Sometimes full picture: SE provides ad results to a different search engine: e.g. Google to Ask.
Interactions in Sponsored Search

- **Advertisers:**
  - Submit ads associated to certain *bid phrases*
  - Bid for position
  - Pay CPC

- **Users**
  - Make queries to search engine, expressing some *intent*

- **Search engine**
  1. Executes query against web corpus + other data sources
  2. Executes query against the ad corpus
  3. Displays a *Search Results Page (SERP)* = integration of web results, other data, and *ads*
Types of ads

For a given query the engine can display two types of ads:

1. “Exact match” (EM)  ➔ The advertiser bid on that specific query a certain amount

2. “Advanced match” (AM) or “Broad match”  ➔ The advertiser did not bid on that specific keyword, but the query is deemed of interest to the advertiser.
   - Needed to ensure volume + ads on new/rare queries (advertise on the tail queries)
   - Advertisers usually opt-in to subscribe to AM
   - From the SE point of view AM is much more challenging
Practical implementation
The two approaches

1. **The database approach** (original Overture approach)
   - Ads are records in a database
   - The bid phrase (BP) is an **attribute**
   - On query q
     - For EM consider all ads with BP=q
     - For AM translate q into “equivalent” queries q1, q2, … and consider all ads with BP=q1, BP = q2, …

2. **The IR approach** (modern view)
   - Ads are documents in an **ad corpus**
   - The bid phrase is a meta-datum
   - On query q run q against the ad corpus
     - Have a suitable ranking function (more later)
     - BP = q (exact match) has high weight
     - No distinction between AM and EM
Which ads?
The economic view

• The search engine wants to maximize long term revenue

• To this end it needs

  1. **Advertisers:** want ROI
     • Conversions = make money directly
     • Reach the right audience
     • Volume – there is a friction cost

  2. **Users:** want to reach their goals
     • Find relevant results
     • Have a good search experience

• Each of the SE, Advertisers, and Users has its own utility
Conflicts and synergies

- Some utilities are aligned, some are in conflict

- **Aligned:**
  - SE and advertisers want more clicks – revenue for SE + volume for advertisers
  - SE and users want good ads – ads offer information + users click and SE makes money

- **Conflicting utilities:**
  - Higher cost per click better for SE but worse for advertiser
  - Irrelevant ads sometimes clicked annoy many users but make money for SE, ROI for advertiser

- **How to balance?**
User useful ads

**Miele Appliances On Sale**
Free Shipping on Miele Appliances Fast, Reliable Nationwide Delivery.
www.us-appliance.com/miele

**Miele Range Hood DA3190 - krillion.com**
Find great products like the Miele Range Hood in-stock and on-sale at a store near you. Krillion provides relevant local search results.
www.krillion.com/xNOP-Miele-Range_Hoods-DA3190

**Da3190 Miele Built-in Wall Hood - Shop.com**
Shop for Da3190 Miele Built-in Wall Hood and Vent Hoods & Duct Covers at Shop.com.
DA3190 36.90 cm* Built-in hood with retractable canopy** Home Store|Large...

**Da3190**
A Giant Selection of da3190. Shop Here Now and Save.
www.become.com

**Da3190**
Create A Cooking Paradise. Save On Da3190.
RangeHoods.Shopzilla.com

See your message here...
Optimization

- Total utility of a Sponsored Search system is a balance of the individual utilities:
  \[ \text{Utility} = f(\text{UtilityAdvertiser}, \ \text{UtilityUser}, \ \text{UtilitySE}) \]

- Function \( f() \) combines the individual utilities

- How to choose an appropriate combination function?
  - Model the long-term goal of the system
  - Parameterized to allow changes in the business priorities
  - Simple – so that business decisions can be done by the business owners!

- Example: convex linear combination:
  \[ \text{Utility} = \alpha \ast \text{UtilityAdvertiser} + \beta \ast \text{UtilityUser} + \gamma \ast \text{UtilitySE} \]

  where  \( \alpha + \beta + \gamma = 1 \)
Utility – more pragmatic view

- Long term utilities are hard to capture/quantify
- Instead

Maximize per search revenue subject to
1. User utility per search > $\alpha$
2. Advertiser ROI per search > $\beta$

- Practically:
  1. Find all ads that have user utility above $\alpha$
  2. Optimize which ads to show based on an auction mechanism as discussed before (captures the $\beta$)
Why do it this way?

(As opposed to first find all ads with utility > $\beta$, etc)

- **Ad relevance**: is a simple proxy for total utility:
  - Users – better experience
  - Advertisers – better (more qualified) traffic but possible volume reduction
  - SE gets revenue gain through more clicks but possible revenue loss through lower coverage

- Ad relevance does not solve all problems
  - When to advertise: certain queries are more suitable for advertising than others
  - Interaction with the algorithmic side of the search

- **In the remaining of this lecture we will focus on retrieving relevant ads!**
Web Queries
Web User Needs

- Need [Brod02, RL04]
  - Informational – want to learn about something (~40% / 65%)  
    - Swine Flu prevention
  - Navigational – want to go to that page (~25% / 15%)  
    - Alitalia US
  - Transactional – want to do something (web-mediated) (~35% / 20%)
    - Access a service
    - Downloads
    - Shop
  - Gray areas
    - Find a good hub
    - Exploratory search “see what’s there”
    - Rome hotels

New York weather
Mars surface images
Nokia mp3 phone
Web Queries Statistics

- Queries are a (very) succinct representation of the user’s intent = The ultimate driver of the ad selection
- Before any grand design, let’s look at the data:
  - Data points from Steve Beitzels’ PhD thesis 2006:
    - Based on AOL queries
    - Dataset 1: a week of queries from Dec. 2003
    - Dataset 2: six months of queries Sept. 2005-Feb. 2005
### Table 2.1. Aggregate Query Log Statistics

<table>
<thead>
<tr>
<th>Property</th>
<th>One week</th>
<th>Six months</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Queries</td>
<td>Hundreds of Millions</td>
<td>Billions</td>
</tr>
<tr>
<td>Number of Users</td>
<td>Tens of Millions</td>
<td>Tens of Millions</td>
</tr>
<tr>
<td>Average Query Length</td>
<td>2.2 Terms</td>
<td>2.7 Terms</td>
</tr>
<tr>
<td>Average Popular Query Length</td>
<td>1.7 Terms</td>
<td>1.7 Terms</td>
</tr>
<tr>
<td>Portion of users viewing first results page</td>
<td>81%</td>
<td>79%</td>
</tr>
<tr>
<td>Portion of users viewing second results page</td>
<td>18%</td>
<td>15%</td>
</tr>
<tr>
<td>Portion of users viewing three or more pages</td>
<td>1%</td>
<td>6%</td>
</tr>
</tbody>
</table>
Frequency breakdown
The long tail

Note the large "tail" of rare queries. In any given hour, approximately 50% of queries occur 5 times or less.
Query Volume per Hour of the Day

Note the drop in query volume during off-peak time, and its subsequent rise throughout the remainder of the day.

Figure 2.1. Query Volume Over a Day
Query Volume: Day of Week

Figure 2.3. Average Volume of Days in the Week
Topical Distribution of Web Queries

Figure 2.9. Breakdown of Categorized Queries
Textual Ads
Anatomy of a Textual Ad: the Visible and Beyond

Title
Creative
Display URL

ACL-08: HLT Tutorial
Computational Advertising Tutorial
Columbus, OH - June 15, 2008
research.yahoo.com

Bid phrase: computational advertising
Bid: $0.5

Landing URL: http://research.yahoo.com/tutorials/acl08_compadv/
OVERVIEW

Web advertising is the primary driving force behind many Web activities, including Internet search as well as publishing of online content by third-party providers. A new discipline - Computational Advertising - has recently emerged, which studies the process of advertising on the Internet from a variety of angles. A successful advertising campaign should be relevant to the immediate user's information need as well as more generally to
Types of Landing Pages

[H. Becker, AB, E. Gabrilovich, VJ, B. Pang, SIGIR 2009]

- Classify landing page types for all the ads for 200 queries from the 2005 KDD Cup labeled query set. Four prevalent types:

**I. Category (37.5%)**: Landing page captures the broad category of the query

**II. Search Transfer (26%)**: Land on dynamically generated search results (same q) on the advertiser’s web page
  - a) Product List – search within advertiser’s web site
  - b) Search Aggregation – search over other web sites

**III. Home page (25%)**: Land on advertiser’s home page

**IV. Other (11.5%)**: Land on promotions and forms
Beyond a Single Ad

- Advertisers can sell multiple products
- Might have budgets for each product line and/or type of advertising (AM/EM) or bunch of keywords
- Traditionally a focused advertising effort is named a campaign
- Within a campaign there could be multiple ad creatives
- Financial reporting based on this hierarchy
Campaign organization

Advertiser

Account 1

Campaign 1

Ad group 1

Creatives

Bid phrases

Campaign 2

Ad group 2

...
Ad Selection
Exact Match Challenges

- What is an exact match?
  - Is “Miele dishwashers” the same as
    - Miele dishwasher (singular)
    - Meile dishwashers (misspelling)
    - Dishwashers by Miele (re-order, noise word)
  - Query normalization
- Which exact match to select among many?
  - Varying quality
    - Spam vs. Ham
    - Quality of landing page
  - Suitable location
  - More suitable ads (E.g. specific model vs. generic “Buy appliances here”)
  - Budget drain
    - Cannot show the same ad all the time
  - Economic considerations (bidding, etc)
Broad/Advanced match

- Significant portion of the traffic has no bids
- Advertisers do not care about bid phrases – they care about conversions = selling products

Main approaches
- The DB approach = Query rewriting
- The IR approach = Search based methods
- Syntactic approach = purely syntactic extensions, e.g. all queries containing the word “Seattle”
Not easy to capture all opportunities

- Advertiser can bid on “broad queries” and/or “concept queries”
  - Suppose your ad is:
    - “Good prices on Seattle hotels”
  - Can bid on any query that contains the word Seattle
- Problems
  - What about query “Alaska cruises start point”?
  - What about “Seattle's Best Coffee Chicago”
- Ideally
  - Bid on any query related to Seattle as a travel destination
  - We are not there yet …
- Market Question: Should these “broad matches” be priced the same?
  - Whole separate field of research
Implementation:
Ad Selection in Two Phases

- **Ad Retrieval**: Consider the whole ad corpus and select a set of most viable candidates (e.g. 100)
- **Ad Reordering**: Re-score the candidates using a more elaborate scoring function to produce the final ordering

Why do we need 2 phases:
- Ad Retrieval:
  - considers a larger set of ads, using only a subset of available information
  - might have a different objective function (e.g. relevance) than the final function
- Ad Reordering
  - Limited set of ads with more data and more complex calculations
  - Must use the bid in addition to the retrieval score (e.g. revenue as criteria for the ordering, implement the marketplace design())
Query Rewriting for Broad Match
Query Rewriting Flow

- Typical of the DB approach to AM
- Rewrite the user query $q$ into $Q' = (q_1, q_2, \ldots)$
- Use EM to select ads for $Q'$
- Two basic strategies:
  1. Static: $Q \rightarrow Q'$ table produced and saved for use at runtime
     - Rewrites can be generated off line: more time and more data can be used
  2. Dynamic: $Q$ mapped into $Q'$ at runtime using matching algorithms
     - Only a few milliseconds available
Data Sources

Queries

- contains
  - co-occurrence
    - search result
    - co-occurrence

Query Sessions

- issued

Users

- clicks

Web pages

- clicks
- bid phrases
- similarity

Ads

- clicks
## Query Rewrite Methods Discussed in this Lecture

<table>
<thead>
<tr>
<th>Query rewriting technique</th>
<th>Data source</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Generating Query Substitutions: Jones et al, in Proc of WWW 2006</td>
<td>query logs (query sessions)</td>
</tr>
<tr>
<td>2. Simrank++: Query Rewriting through Link Analysis of the Click Graph: Atoanellis et al., In proc of VLDB 2008</td>
<td>co-clicks on ads</td>
</tr>
<tr>
<td>3. Learning Query Substitutions for Online Advertising: Broder et al. in Proc of ACM SIGIR 2008</td>
<td>query-to-ad similarity</td>
</tr>
<tr>
<td>5. Query Word Deletion Prediction: Jones at al., in Proc of ACM SIGIR 2003</td>
<td>query logs</td>
</tr>
</tbody>
</table>
Query Rewriting using Web Search Query Logs

Data Source

Queries -> Web pages

Query Sessions -- contains -- Users

Web pages -- issued -- Queries

Queries -- co-occurrence -- Ads

Web pages -- search result -- Queries

Users -- clicks -- Ads

Queries -- similarity -- Ads

Queries -- bid phrases -- Ads

Data Source
Session: Trying to Find Nathan Welsh, who lives and works in Edinburgh

1. nathan welsh edinburg scotland
2. nathan welsh edinburgh scotland
3. financial consultants edinburg scotland
4. financial consultants edinburgh scotland
5. financial consultants
6. nathan welsh 16-18 pennwell place edinburgh
7. ...

Define: Query pair \(<q_1, q_2>\): Two queries issued by the same user in the same day
Types of query reformulations in the log

• Enhance meaning
  • Spell correction
  • Corpus-appropriate terminology
    • *Cat cancer* → *feline cancer*

• Change meaning
  • Narrow
    • [lexical entailment: *fruit* → *apple*]
  • Broaden
    • [alternatives, common interests]
    • Conference proceedings → textbooks
### Half of Query Pairs are Reformulations

<table>
<thead>
<tr>
<th>Type</th>
<th>Example</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-rewrite</td>
<td>mic amps -&gt; create taxi</td>
<td>53.2%</td>
</tr>
<tr>
<td>insertions</td>
<td>game codes -&gt; video game codes</td>
<td>9.1%</td>
</tr>
<tr>
<td>substitutions</td>
<td>john wayne bust -&gt; john wayne statue</td>
<td>8.7%</td>
</tr>
<tr>
<td>deletions</td>
<td>skateboarding pics -&gt; skateboarding</td>
<td>5.0%</td>
</tr>
<tr>
<td>spell correction</td>
<td>real estate -&gt; real estate</td>
<td>7.0%</td>
</tr>
<tr>
<td>mixture</td>
<td>huston's restaurant -&gt; houston's</td>
<td>6.2%</td>
</tr>
<tr>
<td>specialization</td>
<td>jobs -&gt; marine employment</td>
<td>4.6%</td>
</tr>
<tr>
<td>generalization</td>
<td>gm reabtes -&gt; show me all the current auto rebates</td>
<td>3.2%</td>
</tr>
<tr>
<td>other</td>
<td>thanksgiving -&gt; dia de acconde gracias</td>
<td>2.4%</td>
</tr>
</tbody>
</table>

[Jones & Fain SIGIR 2003]
Many substitutions are repeated

- car insurance → auto insurance
  - 5086 times in a sample
- car insurance → car insurance quotes
  - 4826 times
- car insurance → geico [brand of car insurance]
  - 2613 times
- car insurance → progressive auto insurance
  - 1677 times
- car insurance → carinsurance
  - 428 times

Different Users, Different Days
Statistical Test to Find Significant Rewrites

Test whether

\[ p(q_2 \mid q_1) \gg p(q_2) \]

P(britney spears|brittney spears) >> P(britney spears)

8% >> 0.01%

Log likelihood ratio test (LLR) gives $\chi^2$ distributed score

About 90% of query pairs are related after filtering with LLR > 100
Tail Coverage with Query Segmentation

- What about rare queries?
- Query segmented using high mutual information terms: \( p(a,b) > k \cdot p(a)p(b) \)
- Learn segment replacement
- Most frequent queries: replace whole query
- Infrequent queries: replace constituent phrases

Diagram:
- Castles in Edinburgh
- Medieval castles near Glasgow

Legend:
- Orange: Represents initial query
- Blue: Represent rewrite query
Second example

• “catholic baby names” →
  \{christian baby names, 
  christian baby boy names, 
  catholic names, ...\}
More General Representation

\[ Q \rightarrow Q' \rightarrow Q'' \rightarrow \cdots \]

\[ p_1 p_2 \rightarrow p_1 p_2 \rightarrow p_1 p_2 \rightarrow \cdots \]

\textit{segmentation}
Learning to distinguish good pairs

- Sample 1000 queries (q1)
- Select a single substitution for each (q2)
- Manually label the \(<q1,q2>\) pairs from 1 to 4
- Learn to score \(<q1,q2>\) pairs based on the manually labeled set
- Order by score
- Assess Precision/Recall
  - Precise task \{1,2\} vs \{3,4\}
  - Broad task \{1,2,3\} vs \{4\}
Evaluation

- Two baselines:
  - Sample one random rewrite from top-10 whole query candidates and limited phrase substitution rewrites with LLR score > 50 – 55% of pairs
  - Take suggestions with LLR > 100, favoring whole query suggestions and suggestions with less phrase substitutions, cap per type – 66% of pairs
- Tested methods
  - Decision Trees
  - Linear Regression of Editorial Scores
  - 2-class classification with SVM (Editorial score threshold to distinguish the classes)
- Average precision / recall with 10-fold cross validation
Features Used in Scoring Rewrites

- Total of 37 features from 3 general groups:
  - Lexical features
    - Character edit distance
    - Prefix overlap
    - Porter-stem
    - Jaccard score on words
  - Statistical features
    - Probability of rewrite
    - Frequency of rewrite
  - Other
    - Number of substitutions (numSubst)
      - Whole query = 0
      - Replace one phrase = 1
      - Replace two phrases = 2
  - Query length,
  - Bid phrase of an ad
Simple Decision Tree

wordsInCommon > 0

Yes

Class={1,2}

No

prefixOverlap>0

Yes

Class={1,2}

No

Class={3,4}

Interpretation of the decision tree:
- substitution must have at least 1 word in common with initial query
- the beginning of the query should stay unchanged
Linear Regression using Editorial Scores

\[ EditorialScore = \text{intercept} + \sum_{f=\text{features}} w_f \cdot f \]

- Outputs continuous score [1..4]
- Like decision tree
  - Prefer few edits
  - Prefer few word changes
  - Prefer whole-query or few phrase changes
- Normalize output to a probability of correctness using sigmoid fit: \( p(f) = \frac{1}{1+e^{-f}} \)
- Result:
  \[ f(q_1,q_2) = 0.74 + 1.88 \times \text{editDist}(q_1,q_2) \]
  \[ + 0.71 \times \text{wordDist}(q_1,q_2) \]
  \[ + 0.36 \times \text{numSubst}(q_1,q_2) \]
SVM, Bags of Trees, Linear Model Give Similar Trade-offs
## Example Query Substitutions

<table>
<thead>
<tr>
<th>Initial Query</th>
<th>Substitution</th>
<th>Hand-label</th>
<th>Alg. Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>anne klien watches</td>
<td>anne klein watches</td>
<td>1</td>
<td>92%</td>
</tr>
<tr>
<td>sea world san diego</td>
<td>sea world san diego tickets</td>
<td>2</td>
<td>90%</td>
</tr>
<tr>
<td>restaurants in washington dc</td>
<td>restaurants in washington</td>
<td>2</td>
<td>89%</td>
</tr>
<tr>
<td>nash county</td>
<td>wilson county</td>
<td>3</td>
<td>66%</td>
</tr>
<tr>
<td>frank sinatra birth certificate</td>
<td>elvis presley birth</td>
<td>4</td>
<td>17%</td>
</tr>
</tbody>
</table>
Using Clicks on Ads for Query Rewriting

Simrank++: Query Rewriting through Link Analysis of the Click Graph
by Ioannis Antonellis, Hecor Garcia-Molina and Chi-Chao Chang
VLDB 2008
Slides based on the VLDB 2008 presentation by authors
Data source

- Query Sessions
- Users
- Queries
- Web pages
- Ads

- issued
- contains
- clicks
- search result
- co-occurrence
- similarities
- bid phrases

Data source: Contains queries and clicks, search result, co-occurrence, similarities, bid phrases.
Click Graph from sponsored search

Queries
- pc
- camera
- Digital camera
- tv
- flower

Ads
- Hp.com
- Bestbuy.com
- Teleflora.com
- Orchids.com

Clicks
- 20
- 5
- 7
- 15
- 16
- 15

Similar Queries
- camera
- Digital camera
- pc
- camera
- pc
- Digital camera
- tv
- camera
- tv
- Digital camera
- pc
- tv
Simrank Intuition

- Intuition:
  - “Two queries are similar if they are connected to similar ads”
  - “Two ads are similar if they are connected to similar queries”
- Iterative procedure: at each iteration similarity propagates in the graph
Simrank algorithm

- \( E(q) \): set of ads connected to \( q \)
- \( N(q) \): # of ads connected to \( q \)
- \( \text{sim}_k(q,q') \): \( q-q' \) similarity at \( k \)-th iteration
- Initially \( \text{sim}(q,q) = 1 \), \( \text{sim}(q,q') = 0 \), \( \text{sim}(a,a) = 1 \), \( \text{sim}(a,a') = 0 \)

\[
\text{sim}_k(q,q') = \frac{C}{N(q)N(q')} \sum_{i \in E(q)} \sum_{j \in E(q')} \text{sim}_{k-1}(i,j)
\]

\[
\text{sim}_k(a,a') = \frac{C}{N(a)N(a')} \sum_{i \in E(a)} \sum_{j \in E(a')} \text{sim}_{k-1}(i,j)
\]

- \( C \) – constant smaller than 0, ensures diminishing impact with increased number of steps (small \( k \) \( \text{sim} \) goes to 0)
- Theorem: Solution always exist and is unique
Simrank in matrix notation

- **Input:**
  - transition matrix \( P \),
  - scalar decay factor \( C \),
  - scalar number of iterations \( k \)

- **Output:** similarity matrix \( S \)

For \( i = 1:k \), do

\[
S = C P^T S P
\]

Set diagonal entries of \( S \) to 1
### Simrank

**1st Iteration**

<table>
<thead>
<tr>
<th></th>
<th>pc</th>
<th>camera</th>
<th>digital camera</th>
<th>tv</th>
<th>flower</th>
</tr>
</thead>
<tbody>
<tr>
<td>pc</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>camera</td>
<td>0.0889</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>digital camera</td>
<td>0.0889</td>
<td>0.1778</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tv</td>
<td>0</td>
<td>0.0889</td>
<td>0.0889</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>flower</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

\[
s_k(q, q') = \frac{C}{N(q)N(q')} \sum_{i \in E(q)} \sum_{j \in E(q')} s_{k-1}(i, j)
\]

\[
s_k(a, a') = \frac{C}{N(a)N(a')} \sum_{i \in E(a)} \sum_{j \in E(a')} s_{k-1}(i, j)
\]

\[C = 0.8\]
### Simrank

#### 2nd Iteration

<table>
<thead>
<tr>
<th></th>
<th>pc</th>
<th>camera</th>
<th>digital camera</th>
<th>tv</th>
<th>flower</th>
</tr>
</thead>
<tbody>
<tr>
<td>pc</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>camera</td>
<td>0.1244</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>digital camera</td>
<td>0.1244</td>
<td>0.2489</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tv</td>
<td>0.0356</td>
<td>0.1244</td>
<td>0.1244</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>flower</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ s_k(q, q') = \frac{C}{N(q)N(q')} \sum_{i \in E(q)} \sum_{j \in E(q')} s_{k-1}(i, j) \]

\[ s_k(a, a') = \frac{C}{N(a)N(a')} \sum_{i \in E(a)} \sum_{j \in E(a')} s_{k-1}(i, j) \]

\( C = 0.8 \)
## Simrank

### 12th Iteration

<table>
<thead>
<tr>
<th></th>
<th>pc</th>
<th>camera</th>
<th>digital camera</th>
<th>tv</th>
<th>flower</th>
</tr>
</thead>
<tbody>
<tr>
<td>pc</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>camera</td>
<td>0.1650</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>digital camera</td>
<td>0.1650</td>
<td>0.33</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tv</td>
<td>0.0761</td>
<td>0.1650</td>
<td>0.1650</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>flower</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\[
s_k(q, q') = \frac{C}{N(q)N(q')} \sum_{i \in E(q)} \sum_{j \in E(q')} s_{k-1}(i, j)
\]

\[
s_k(a, a') = \frac{C}{N(a)N(a')} \sum_{i \in E(a)} \sum_{j \in E(a')} s_{k-1}(i, j)
\]

\[C = 0.8\]
Problems with Simrank

• Complete bipartite graphs: every node on the left links to every node on the right
• Why complete graphs?
  • Allows for simplified setting to examine the algorithm performance
• Finding: Simrank scores in complete bipartite graphs are sometimes counter-intuitive
• See Theorems in paper, here examples for intuition
Example: Similarity depends on the size of one of the graph partition

Theorem: Scores converge for $C=1$

<table>
<thead>
<tr>
<th>iteration</th>
<th>Camera – digital camera</th>
<th>Pc - camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.4</td>
<td>0.8</td>
</tr>
<tr>
<td>2</td>
<td>0.62</td>
<td>0.8</td>
</tr>
<tr>
<td>3</td>
<td>0.65</td>
<td>0.8</td>
</tr>
<tr>
<td>4</td>
<td>0.66</td>
<td>0.8</td>
</tr>
<tr>
<td>5</td>
<td>0.664</td>
<td>0.8</td>
</tr>
<tr>
<td>6</td>
<td>0.665</td>
<td>0.8</td>
</tr>
</tbody>
</table>

$C = 0.8$
Improvement 1: “Evidence”-based Simrank

\[
evidence(q,q') = \sum_{i=1}^{E(q) \cap E(q')} \frac{1}{2^i}
\]

\[
sim^k_{\text{evidence}}(q,q') = \evidence(q,q') \cdot \sim_k(q,q')
\]

<table>
<thead>
<tr>
<th>iteration</th>
<th>Camera – digital camera</th>
<th>Pc – camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.3</td>
<td>0.4</td>
</tr>
<tr>
<td>2</td>
<td>0.42</td>
<td>0.4</td>
</tr>
<tr>
<td>3</td>
<td>0.468</td>
<td>0.4</td>
</tr>
<tr>
<td>4</td>
<td>0.4872</td>
<td>0.4</td>
</tr>
<tr>
<td>5</td>
<td>0.49488</td>
<td>0.4</td>
</tr>
<tr>
<td>6</td>
<td>0.497952</td>
<td>0.4</td>
</tr>
</tbody>
</table>

C = 0.8
Link Weights

- So far the assumption is that all edges are equal
- Can we differentiate edges?
- Candidate strength indicators for the links:
  - Impressions
  - Clicks/CTR
- As Sponsored Search is mostly click based, CTR is more appropriate for weighting the edges
Position Bias

- Need to resolve position bias issue:
  - Ads shown on position 1 are more likely to get clicks even if they are less relevant
- Option 1: aggregate the CTR at each position
  - Issue: Ads at position 1 are more clicked because of their position as well as due to relevance ordering
    \[ p(\text{click}) = p(\text{seen})p(\text{relevant}) \]
- We need to separate the positional and relevancy effects
- Field of active study in the search community
Back to Simrank: Weighted Simrank

Absolute value of weights matters
Weighted Simrank: Variance

Why would variance matter?
Weighted Simrank

\[ p(a,i) = \text{spread}(i) \cdot \text{normalized_weight}(a,i), \forall i \in E(a) \]

\[ p(a,a) = 1 - \sum_{i \in E(a)} p(a,i) \]

\[ \text{spread}(i) = \frac{1}{\text{variance}(i)} \]

\[ \text{normalized_weight}(a,i) = \frac{w(a,i)}{\sum_{j \in E(a)} w(a,j)} \]
Evaluation: Dataset

- Dataset:
  - 2 weeks Yahoo! click graph, 15 million queries, 14 million ads, 28 million edges
  - Extracted largest connected component and further decomposed it into 5 subgraphs (details in the paper)
  - Edge weights: adjusted clicks over impressions rate (to account for position bias)

- Evaluation set:
  - 120 queries sampled from search engine traffic
Evaluation: Baseline and Method

- Baseline: Pearson similarity (also tried Jaccard and Cosine):

\[
sim(q, q') = \frac{\sum_{a \in E(q) \cap E(q')} (w(q, a) - \bar{w}_q)(w(q', a) - \bar{w}_{q'})}{\sqrt{\sum_{a \in E(q) \cap E(q')} (w(q, a) - \bar{w}_q)^2 (w(q', a) - \bar{w}_{q'})^2}}
\]

- Metrics: Precision/recall (editorial evaluation)
  - Precision(q) = relevant rewrites of q / number of rewrites for q (among all methods)
  - Recall(q) = relevant rewrites of q / number of relevant rewrites for q (among all methods)
Summary of the Results
Thank You

broder@yahoo-inc.com
vanjaj@yahoo-inc.com

http://research.yahoo.com
This talk is Copyright 2009. Authors retain all rights, including copyrights and distribution rights. No publication or further distribution in full or in part permitted without explicit written permission.