Introduction to Computational Advertising

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
Lecture 9: Contextual Advertising 2
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.
Checkpoint

Contextual Advertising
Contextual Advertising (Content Match)

- Textual advertising on third-party web pages
- Complement the content of the web page with paid content
- Ubiquitous on the web
- Supports the diversity of the web
  - Sites small and big rely on CM revenue to cover for the cost of existence
- Players
  - Google: Adsense
  - Microsoft: ContentAds
  - Yahoo!: Content match
Checkpoint (cont.)

- Two main ways of selecting unseen ads in CM:
  - Single feature (phrase extraction):
    - Lecture 8: ML based approach to learn how to extract phrases from pages
  - Multiple features (similarity)
    - Lecture 8: Impedance coupling – expanding the page with similar features

- Ads that have been already in the system for a while: use click data for better placement
  - Abstract to higher-level features for click-based ad selection mechanisms
# Reading for Lecture 9

<table>
<thead>
<tr>
<th>Paper</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. A hidden class page-ad probability model for Contextual Advertising. A. Ratnaparkhi In Proc. of TROA 2008</td>
<td>click modeling using latent topics</td>
</tr>
<tr>
<td>4. Estimating Rates of Rare Events at Multiple Resolutions. Agarwal et al. In Proc of KDD 2007</td>
<td>click aggregation at multiple levels</td>
</tr>
<tr>
<td>5.</td>
<td></td>
</tr>
<tr>
<td>6. To Swing or not to Swing: Learning when (not) to Advertise. Broder et al. CIKM 2008</td>
<td>when to advertiser</td>
</tr>
</tbody>
</table>
Holistic view at the page in Contextual Advertising

Motivation

- Even with using multiple features there is still a risk that the subset used in matching does not represent the semantics of the page
- Can we somehow summarize the content of the whole page into a small number of features?
  - This work: supervised approach based on classification
- Use external knowledge: taxonomies
  - This work: a topical taxonomy
- What is a better signal: page class or page words? Or both?
Semantic-syntactic match

- Figure out the topic of the page
  - Classification of the page into a commercial oriented taxonomy

- Pre-classify all the ads into the same taxonomy

- Restrict the matching to ads that are in related categories

- Use word similarity to improve the match
Page and ad classification

- Use a large scale classification to relate pages and ads
  - Need a taxonomy with sufficient resolution

- We used a taxonomy of 6,000+ nodes, primarily built for classifying commercial interest queries
  - Each node is a collection of query terms

- Rocchio-style nearest neighbor classifier
  - Meta-document produced of the queries at each node
Taxonomy requirements: intuition

- Enough resolution to be useful
- Not too specific to make maintenance too costly:
  - Electronics - too broad
  - Electronics/Digital Camera/Canon - feasible
  - Electronics/Digital Camera/Canon/XT10i - hard to maintain
Taxonomy statistics
Scoring

- For a given page score every ad, select the top-k ads
- Linear combination of 2 scores:
  - Taxonomy score (semantic distance)
  - Word and phrase score (syntactic distance)

- Allow generalization in the taxonomy

\[
\text{Score}(p_i, a_i) = \alpha \cdot \text{TaxScore}(\text{Tax}(p_i), \text{Tax}(a_i)) + (1-\alpha) \cdot \text{KeywordScore}(p_i, a_i)
\]
Generalization paths

- winter sports
  - skiing
  - snowboarding
  - Atomic snowboard
  - Atomic pages

match

ads
Semantic and syntactic scores

- **Semantic component - class based**
  \[ Tax(d_j) = \{ d_{j1} \ldots d_{jv} \} \quad \sum_{d \in Tax(x_i)} cWeight(d) = 1 \quad idist(c, p) = \frac{n_c}{n_p} \]

- **Syntactic component - term vector cosine**
  \[ TaxScore(PC, AC) = \sum_{pc \in PC} \sum_{ac \in AC} idist(LCA(pc, ac), ac) \cdot cWeight(pc) \cdot cWeight(ac) \]

\[ tWeight(kw^{st}) = weightSection(S_i) \cdot tf_{idf}(kw) \]

\[ KeywordScore(p_i, a_i) = \frac{\sum_{i \in |K|} tWeight(pw_i) \cdot tWeight(kw_i)}{\sqrt{\sum_{i \in |K|} (tWeight(pw_i))^2} \sqrt{\sum_{i \in |K|} (tWeight(aw_i))^2}} \]
Searching the ad space

- Ad search done in real time - how to make it fast enough?
- Index the ads using a inverted index
  - Use the page features as the query
- Find top-k ads with the highest score
- Monotonic scoring function that has the two sub-scores
- Evaluate the query using a variant of the WAND doc-at-a-time algorithm [Broder et al.]
Dataset

• Ad-page pairs manually evaluated 3 times by human editors as: (1) Relevant; (2) Somewhat relevant; and (3) Irrelevant

• Average judgments and round to the closest integer

• 3 x 3K judgments for a set of 105 pages

• The pages sampled from a set of over 20M pages that are enabled for contextual advertising

• Ads selected from a set of over 10M ads
## Dataset Statistics

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of pages</td>
<td>105</td>
</tr>
<tr>
<td>Words per page</td>
<td>868</td>
</tr>
<tr>
<td>Judgments</td>
<td>2946 x 3</td>
</tr>
<tr>
<td>Inter-editor agreement</td>
<td>84%</td>
</tr>
<tr>
<td>Unique ads shown</td>
<td>2680</td>
</tr>
<tr>
<td>Unique ads per page</td>
<td>25.5</td>
</tr>
<tr>
<td>Page class. precision</td>
<td>70%</td>
</tr>
<tr>
<td>Ad class. precision</td>
<td>80%</td>
</tr>
</tbody>
</table>
Methodology

- Data from prior experiments
- For each page consider only the judged ads
  - Did not have the exact ad set used in the original experiments
- Rank the ads by each method
- Precision/recall and K-tau to compare different orderings
- Precision at 1, 3, 5
- Evaluate relative performance of the methods
Some Results - using past editorial judgments

<table>
<thead>
<tr>
<th>alpha</th>
<th>K-tau</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0.086</td>
</tr>
<tr>
<td>0.25</td>
<td>0.155</td>
</tr>
<tr>
<td>0.5</td>
<td>0.158</td>
</tr>
<tr>
<td>0.75</td>
<td>0.166</td>
</tr>
<tr>
<td>1.00</td>
<td>0.136</td>
</tr>
</tbody>
</table>
Precision at position 1, 3, and 5

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>#1</th>
<th>#3</th>
<th>#5</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>80</td>
<td>70</td>
<td>68</td>
<td>218</td>
</tr>
<tr>
<td>0.25</td>
<td>89</td>
<td>76</td>
<td>73</td>
<td>238</td>
</tr>
<tr>
<td>0.50</td>
<td>89</td>
<td>74</td>
<td>73</td>
<td>236</td>
</tr>
<tr>
<td>0.75</td>
<td>89</td>
<td>78</td>
<td>73</td>
<td>240</td>
</tr>
<tr>
<td>1.00</td>
<td>86</td>
<td>79</td>
<td>74</td>
<td>239</td>
</tr>
</tbody>
</table>
Conclusions

- Contextual advertising is the economic engine for a large number of non-transactional sites
- Novel way to match ads to pages
- Topical (semantic) similarity is a major component of the relevance score (~80%)
- Evaluation showing results for different alpha values
Using topics for ad selection

A hidden class page-ad probability model for Contextual Advertising. A. Ratnaparkhi In Proc. of TROA 2008
Bridging the gap between the ads and the pages

- Text of related pages and **topics** (classes) are improving ad selection
- The ‘topics’ could be induced from the click data – no need to have a fixed structure and a training set
- Topics can be described in terms of features and weights – no need to specify instances
Inducing topics from click data

- Collect clicks of (page, ad) pairs
- Estimate class-based probability model on this data
- Premise: (page, ad) click share some topic
- Use topics probabilities to assign topics to pages and ads
  - Page and ad can be in same topic even with no term overlap
  - Same topic for page and ad implies clickability
“Generate” the click using a hidden topic

Page

b1

Class1

b2

a1

b3

a2
Probability model

\[
p(c, \text{ad}, \text{page}) = p(c) p(\text{ad}|c) p(\text{page}|c) = p(c) \cdot \prod_{i=1}^{\text{size(ad)}} l_{\text{ad}}(\text{size(ad)}|c) \prod_{i=1}^{\text{size(page)}} q_{\text{ad}}(a_i|c) \cdot l_{\text{page}}(\text{size(page)}|c) \prod_{i=1}^{\text{size(page)}} q_{\text{page}}(b_i|c)
\]

\[
p(\text{ad}, \text{page}) = \sum_c p(c, \text{ad}, \text{page})
\]

- Terms and lengths are dependent on class
- Ad and page have separate vocabularies
- Topics are not observed
  - Parameter estimation?
Use EM algorithm for parameter updates

- Each iteration has E-step and M-step
- E-step
  - Imagine that (page, ad) training instance occurs with topic $c$
  - Weight instance by $p(c \mid ad, page)$
    - Using current parameter estimates

$$p(c \mid ad, page, \theta_n) = \frac{p(c, ad, page \mid \theta_n)}{\sum_{c'} p(c', ad, page \mid \theta_n)}$$

- Collect counts
M-step

- Use counts collected in E-step for new estimates
- Maximize the conditional expectation of the complete data log-likelihood:

\[
Q(\theta|\theta_n) = E_{C|T,\theta_n} \left[ \log \prod_{ad,\text{page} \in T} p(c, ad, \text{page}|\theta) \right] \\
= E_{C|T,\theta_n} \left[ \sum_{ad,\text{page} \in T} \log p(c, ad, \text{page}|\theta) \right] \\
= \sum_{ad,\text{page} \in T} E_{c|ad,\text{page},\theta_n} \left[ \log p(c, ad, \text{page}|\theta) \right] \\
= \sum_{ad,\text{page} \in T} \sum_c p(c|ad, \text{page}, \theta_n) \log p(c, ad, \text{page}|\theta)
\]

- Example solution:

\[
g_{ad}(a|c) = \frac{\sum_{ad,\text{page} \in T} p(c|ad, \text{page}, \theta_n) \sum_{i=1}^{\text{size}(ad)} \delta(a, a_i)}{\sum_{ad,\text{page} \in T} p(c|ad, \text{page}, \theta_n) \text{size}(ad)}
\]
Synthetic example

- 2 underlying classes: \textit{flavors} and \textit{sports}
- Each line represents a click:

<table>
<thead>
<tr>
<th>Ad text</th>
<th>Page text</th>
</tr>
</thead>
<tbody>
<tr>
<td>vanilla</td>
<td>chocolate mint</td>
</tr>
<tr>
<td>vanilla</td>
<td>strawberry banana</td>
</tr>
<tr>
<td>football</td>
<td>golf shoes</td>
</tr>
<tr>
<td>soccer</td>
<td>tennis shoes</td>
</tr>
</tbody>
</table>
### Synthetic example (cont’d)

| Ad words  | $q_{ad}(w|c1)$ | $q_{ad}(w|c2)$ |
|-----------|----------------|----------------|
| vanilla   | 0              | 1              |
| soccer    | 0.5            | 0              |
| football  | 0.5            | 0              |

| Page words | $q_{page}(w|c1)$ | $q_{page}(w|c2)$ |
|------------|-----------------|-----------------|
| golf       | 0.25            | 0               |
| strawberry | 0               | 0.25            |
| banana     | 0               | 0.25            |
| tennis     | 0.25            | 0               |
| shoes      | 0.5             | 0               |
| mint       | 0               | 0.25            |
| chocolate  | 0               | 0.25            |

<table>
<thead>
<tr>
<th>Ad text</th>
<th>Page text</th>
</tr>
</thead>
<tbody>
<tr>
<td>vanilla</td>
<td>chocolate mint</td>
</tr>
<tr>
<td>vanilla</td>
<td>strawberry banana</td>
</tr>
<tr>
<td>football</td>
<td>golf shoes</td>
</tr>
<tr>
<td>soccer</td>
<td>tennis shoes</td>
</tr>
</tbody>
</table>
Real click data

- 5M page-ad clicks
  - Collected from existing contextual ad system
  - Similarity between pages and ads is expected
- Page side (first 25 terms)
  - Title
  - Meta-keywords
- Ad side (first 10 terms)
  - Title
  - Bid phrase (supplied by advertiser)
- Terms are phrases
  - In-house phrase dictionary
Examples from real click data

| Ad words                  | $q_{ad}(w|c1)$ | $q_{ad}(w|c2)$ |
|--------------------------|----------------|----------------|
| currency trading         | 0.0820891      |                |
| home loans               | 0.067927       |                |
| countrywide home loans   | 0.067313       |                |
| free demo                | 0.0417377      |                |
| forex currency           | 0.0365207      |                |
| forex currency trading   | 0.0364378      |                |
| invoice price            | 0.00946243     |                |
| used autos               | 0.00848382     |                |
| for sale                 | 0.00793812     |                |
| real estate              | 0.00680954     |                |
| birthday party           | 0.0501497      |                |
| birthday invitations     | 0.0338498      |                |
| free birthday            | 0.0336109      |                |
| free birthday invitations| 0.033425       |                |
| party idea               | 0.024849       |                |
| party invitation         | 0.0225438      |                |
| birthday party invitation| 0.0203071      |                |
| party ideas              | 0.0202639      |                |
| ice cream                | 0.012521       |                |
| alabama job              | 0.0121976      |                |

| Page words               | $q_{page}(w|c1)$ | $q_{page}(w|c2)$ |
|--------------------------|-----------------|-----------------|
| yahoo finance            | 0.0599609       |                |
| exchange rates           | 0.0369237       |                |
| view exchange            | 0.0366989       |                |
| exchange rate            | 0.0339454       |                |
| financial news           | 0.0209973       |                |
| news yahoo               | 0.0209596       |                |
| dollar exchange          | 0.0162197       |                |
| dollar exchange rate     | 0.0162182       |                |
| canadian dollar          | 0.0115819       |                |
| new york                 | 0.00555826      |                |
| birthday party           | 0.0751149       |                |
| party ideas              | 0.072431        |                |
| birthday party ideas     | 0.0662678       |                |
| party planning           | 0.0360878       |                |
| party planning ideas     | 0.0325496       |                |
| party party              | 0.020292        |                |
| for sale                 | 0.0113448       |                |
| nick jr                  | 0.0110672       |                |
| classifieds ads          | 0.00916032      |                |
| classified ads           | 0.00912854      |                |
Conclusion

- The vocabulary gap between the ads and the pages can be bridged by using hidden classes
- Topics (classes) can be discovered using click data
- Might be a better fit for the data
- Easier to bootstrap if you already have a running system
- Sometimes we can guess ‘labels’ for the classes based on the feature weights
Using click data to improve ad retrieval

Relevance vs. Click Data

**Relevance-based**
- Uses IR measures of match
  - cosine similarity
  - BM25
- Uses editorial data
- Matching score

**Click-based**
- Uses ML methods to learn a good matching function
  - Maximum Entropy
- Uses click data
- Probability of click

- No purely click-based systems deployed today (why)?
- Previous work: click based reordering for IR based selection
Goals and Approach

- **Goals**
  - Improve ad selection
  - Preserve the efficiency of the evaluation
    - IR scoring functions mapped to efficient algorithms
  - Transform the relevance score into a click probability \( P_{\text{click}}(\text{ad, page}) \)

- **Our approach:**
  - Start with the relevance scoring formula
  - Add parameters learned from click data
  - Logistic regression to fit score into a probability function
Combined Ad Scoring Function

$$\text{logit}(p_{ij}) = \sum_w \alpha_w M_{p,w} + \sum_w \beta_w M_{a,w} + \sum_w \delta_w I_{p,a,w}$$

- CTR
- Main effect for page (how good is the page)
- Main effect for ad (how good is the ad)
- Interaction effect (words shared by page and ad)

- I,M - IR scoring component
  - tf-idf based
  - Parameters learned from editorial training sets
Challenges

- Three sources of complexity:
  - Transform the IR score to a probability function
  - Learning with:
    - Many parameters
    - Sparseness
    - Correlation of features
  - Fast implementation for training and testing
IR Score – Probability Relationship

- How can IR scores fit into the model?
  - What is the relationship between logit(p_{ij}) and cosine score?
  - Quadratic relationship

  \[
  \text{logit}(p_{ij}) = -1.75 + 2.52 \times \text{cosine} - 1.56 \times \text{cosine}^2
  \]

- Add as a prior

  \[
  \text{logit}(p_{ij}) = \sum_w \alpha_w M_{p,w} + \sum_w \beta_w M_{a,w} + \sum_w \delta_w I_{p,a,w} - 1.75 + 2.52 \times \text{cos} - 1.56 \times \text{cos}^2
  \]
Feature Selection

• First level: consider only features with enough occurrences in the click data
• Further reduction needed to make the learning tractable
  • Click data based:
  • Unsupervised: tf-idf intensities
  • Click data based approach yields better
  \[ i_w = \frac{CTR_w^{both}}{CTR_w^{page} \cdot CTR_w^{ad}} \]
• How many is enough?
  • 1K features give the same result with 3K features
Efficient Training

- Fast Implementation
- Training: Hadoop implementation of Logistic Regression

Data → Random data splits → Iterative Newton-Raphson → Mean and Variance estimates → Combine estimates → Learned model params
Summary

• Combine the IR-style and click-based ranking in the first phase of the ad selection
  • Inverted index friendly scoring formula
• Fit the score into a probability function
• Improve over tf-idf by using click data
When to advertise

To Swing or not to Swing: Predicting when (not) to Advertise. Broder et al, CIKM 2008
The Swing Problem

- Repeatedly showing non-relevant ads can have detrimental long-term effects
- Want to be able to predict when (not) to show individual ads or a set of ads (“swing”)
- Modeling actual short and long term costs of showing non-relevant ads is very difficult
Two Approaches

- Thresholding Approach
  - Decision made on individual ads
  - Only based on ad scores

- Machine Learning Approach
  - Decision made on sets of ads
  - Based on a variety of features

- Applies to both Sponsored Search and Contextual Advertising
Thresholding Approach

- Set a global score threshold
- Only retrieve ads with scores above threshold
- If none of the ad scores are above the threshold, then no ads are retrieved ("no swing")
Machine Learning Approach

- Learn a binary prediction model ("swing" or "no swing") for an entire set of ads
- If we swing, then all ads are retrieved
- If we do not swing, then no ads are retrieved
- Must extract features defined over sets of ads, rather than individual ads
- Use support vector machines (SVMs)
Features

• Relevance features
  • Word overlap
  • Cosine similarity
• Vocabulary mismatch features
  • Translation models
  • Point-wise mutual information
  • Chi-squared
• Ad-based features
  • Bid price
  • Coefficient of variation of ad scores
• Result set cohesiveness features
  • Result set clarity
  • Entropy
Language model for ad cohesiveness

\[ \theta_w = \sum_{A \in A} P(w|A)P(A|Q) \]  \hspace{1cm} (10)

\( P(w|A) \) is the likelihood of item \( w \) given ad \( A \),
\( P(A|Q) \) is the likelihood of ad \( A \) given query \( Q \).
\( \theta_w \) is shorthand for \( P(w|Q) \), which is a multinomial distribution over items \( w \).

\[ P(w|A) = \frac{tf_{w,A}}{|A|} \]  \hspace{1cm} (11)

where \( tf_{w,A} \) is the number of times that term \( w \) occurs in \( A \) and \( |A| \) is the total number of terms in ad \( A \).

\[ P(A|Q) = \frac{SCORE(Q,A)}{\sum_{A' \in ASCORE(Q,A')} \text{SCORE}(Q,A')} \]  \hspace{1cm} (12)

\( \text{SCORE}(Q,A) \) is the score returned by the ad scoring system.
Clarity and Entropy of the Language Models as features

\[
CLARITY(\theta) \overset{\text{def}}{=} KL(\theta \| \hat{\theta}) = \sum_{w \in \mathcal{V}} \theta_w \log \frac{\theta_w}{\hat{\theta}_w}
\]  

(14)

\[
H(\theta) = -\sum_{w \in \mathcal{V}} \theta_w \log \theta_w
\]

(15)
Experimental Results

- **Data sets**
  - Content match (CM1, CM2)
  - Sponsored search (SS)
- “Swing” / “no swing” target
  - Average all human judgments for a given query
  - If below threshold (tau), then target = “swing”

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Size</th>
<th># Judgments</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM1</td>
<td>199 pages</td>
<td>5554</td>
</tr>
<tr>
<td>CM2</td>
<td>1103 pages</td>
<td>13789</td>
</tr>
<tr>
<td>SS</td>
<td>642 queries</td>
<td>8923</td>
</tr>
</tbody>
</table>
**Thresholding vs. SVM Approach**

<table>
<thead>
<tr>
<th>Data</th>
<th>$\tau$</th>
<th>SVM Accuracy</th>
<th>Query Coverage</th>
<th>BPREF</th>
<th>Thresh.</th>
<th>SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM1</td>
<td>1.4</td>
<td>87.94</td>
<td>14.1</td>
<td>.1478</td>
<td>.2219†</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.8</td>
<td>87.90</td>
<td>33.7</td>
<td>.1497</td>
<td>.2019†</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.2</td>
<td>87.94</td>
<td>49.7</td>
<td>.1653</td>
<td>.1917†</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.6</td>
<td>86.43</td>
<td>58.3</td>
<td>.1677</td>
<td>.1929†</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\infty$</td>
<td>100</td>
<td>100</td>
<td>.1488</td>
<td>.1488</td>
<td></td>
</tr>
<tr>
<td>CM2</td>
<td>1.4</td>
<td>94.02</td>
<td>1.1</td>
<td>.1589</td>
<td>.1068</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.8</td>
<td>81.69</td>
<td>2.9</td>
<td>.1182</td>
<td>.1382</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.2</td>
<td>70.53</td>
<td>22.5</td>
<td>.1330</td>
<td>.1804†</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.6</td>
<td>72.35</td>
<td>86.7</td>
<td>.1059</td>
<td>.0943</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\infty$</td>
<td>100</td>
<td>100</td>
<td>.0863</td>
<td>.0863</td>
<td></td>
</tr>
<tr>
<td>SS</td>
<td>3.0</td>
<td>82.71</td>
<td>4.1</td>
<td>.2500†</td>
<td>.3889†</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.5</td>
<td>71.96</td>
<td>13.6</td>
<td>.2339†</td>
<td>.2802†</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.0</td>
<td>69.63</td>
<td>54.9</td>
<td>.1650†</td>
<td>.1539†</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.5</td>
<td>73.99</td>
<td>86.57</td>
<td>.1272</td>
<td>.1254</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\infty$</td>
<td>100</td>
<td>100</td>
<td>.1148</td>
<td>.1148</td>
<td></td>
</tr>
</tbody>
</table>

Bold BPREF = significant increase over other approach

Dagger = significant increase over “always swing” (tau = infinity)

**Summary:**
SVM approach tends to outperform simple thresholding, especially for small values of tau
Conclusion

- Two approaches to determine when to show ads
- Thresholding approach
  - Only shows ads above some global score threshold
  - Most effective for sponsored search
- Machine learning approach
  - Predicts over entire set of ads
  - Semantic class features important for prediction
  - Effective for both sponsored search and content match
- In practice we can combine both approaches
Aggregation of CTR

Estimating Rates of Rare Events at Multiple Resolutions.
Using click data for CTR estimation

- Can we estimate the CTR from the click data that we already have?
- If an ad-page pair has been seen significant (thousands) significant amount of times, we could estimate precisely

Issues:
- Infrequent pairs
- New pages/ads

Figure 1: Distribution of (page, ad) impressions: Plots are on log-log scale but ticks are on the original scale.
Estimation in the “tail”

- Use an existing, well-understood hierarchy
  - Categorize ads and webpages to leaves of the hierarchy
  - CTR estimates of siblings are correlated
    ➔ The hierarchy allows us to aggregate data

- Coarser resolutions
  - provide *reliable* estimates for rare events
  - which then influences estimation at finer resolutions
Units of estimation: region defined by the class hierarchy
System overview

Retrospective data
[URL, ad, isClicked]

Crawl
a sample of URLs

Classify pages and ads

Rare event estimation using hierarchy

Impute impressions, fix sampling bias
Sampling of webpages

• Naïve strategy: sample at random from the set of URLs
  ➔ Sampling errors in impression volume AND click volume

• Instead, we propose:
  • Crawling all URLs with at least one click, and
  • a sample of the remaining URLs
  ➔ Variability is only in impression volume
Imputation of impression volume

\[ \text{#impressions} = n_{ij} + m_{ij} + x_{ij} \]

sums to \( \sum n_{ij} + K \cdot \sum m_{ij} \)

[row constraint]

sums to

Total impressions

(known)

sums to #impressions on
ads of this ad class

[column constraint]

sums to #impressions on
ads of this ad class

[Ad classes]

[Page classes]

[Clicked pool]

[Sampled Non-clicked pool]

[Excess impressions (to be imputed)]
Imputation of impression volume

- **Region**
  = (page node, ad node)

- **Region Hierarchy**
  ➔ A cross-product of the page hierarchy and the ad hierarchy
Imputation of impression volume

Level i

sums to

Level i +1

[block constraint]
Imputing $x_{ij}$

Iterative Proportional Fitting [Darroch + / 1972]

Initialize $x_{ij} = n_{ij} + m_{ij}$

**Top-down:**
- Scale all $x_{ij}$ in every block in $Z^{(i+1)}$ to sum to its parent in $Z^{(i)}$
- Scale all $x_{ij}$ in $Z^{(i+1)}$ to sum to the row totals
- Scale all $x_{ij}$ in $Z^{(i+1)}$ to sum to the column totals

*Repeat for every level $Z^{(i)}*

**Bottom-up:** Similar
Imputation: Summary

• Given
  • \( n_{ij} \) (impressions in clicked pool)
  • \( m_{ij} \) (impressions in sampled non-clicked pool)
  • \# impressions on ads of each ad class in the ad hierarchy

• We get
  • Estimated impression volume
    \[ \tilde{N}_{ij} = n_{ij} + m_{ij} + x_{ij} \]
    in each region \( ij \) of every level
System overview

Retrospective data
[page, ad, isclicked]

Crawl
a sample of pages

Classify pages and ads

Rare event estimation using hierarchy

Impute impressions, fix sampling bias
Rare rate modeling

1. Freeman-Tukey transform

\[ y_r = \frac{1}{2} \left( \sqrt{\frac{c_r}{\tilde{N}_r}} + \sqrt{\frac{c_r + 1}{\tilde{N}_r}} \right) \]

- Distinguishes between regions with zero clicks based on the number of impressions
- **Variance stabilizing transformation**: \( \text{Var}(y) \) is independent of \( \text{E}[y] \) \( \Rightarrow \) needed in further modeling
- \( y \mid S_r, \beta^{(d(r))} \approx N(\beta^{(d(r))} + S_r, V_r) \)
- Simplify: assume that \( V_r = V/N_r \)
- Main part of the model: \( S_r = \text{parent}(S_r) + w_r; \; w_r \approx N(0, W_{d(r)}) \)
Rare rate modeling

2. Generative Model (Tree-structured Markov Model)
Rare rate modeling

- Fitting using a Kalman filtering algorithm
  - Filtering: Recursively aggregate data from leaves to root
  - Smoothing: Propagates information from root to leaves

- Kalman filter requires knowledge of $\beta$, $V$, and $W$
  $\Rightarrow$ EM wrapped around the Kalman filter
Experiments

- 503M impressions
- 7-level hierarchy of which the top 3 levels were used
- Zero clicks in
  - 76% regions in level 2
  - 95% regions in level 3
- Full dataset DFULL, and a 2/3 sample DSAMPLE
Experiments

- Estimate CTRs for all regions $R$ in level 3 with zero clicks in $DSAMPLE$
- Some of these regions $R_{>0}$ get clicks in $DFULL$
- A good model should predict higher CTRs for $R_{>0}$ as against the other regions in $R$
Experiments

- We compared 4 models
  - **TS**: our tree-structured model
  - **LM (level-mean)**: each level smoothed independently
  - **NS (no smoothing)**: CTR proportional to $1/\tilde{N}$
  - **Random**: Assuming $|R_{>0}|$ is given, randomly predict the membership of $R_{>0}$ out of $R$
Experiments

![ROC Curve]

- True positive rate vs False positive rate
- Lines represent different methods: TS, LM, NS, Random
Experiments

Few impressions ➔ Estimates depend more on siblings

Enough impressions ➔ little “borrowing” from siblings
Conclusions

• We presented a method to estimate
  • rates of extremely rare events
  • at multiple resolutions
  • under severe sparsity constraints

• Our method has two parts
  • Imputation $\Rightarrow$ incorporates hierarchy, fixes sampling bias
  • Tree-structured generative model $\Rightarrow$ extremely fast parameter fitting
Issue with click aggregation

- How to bootstrap the process?
- Learn only based on what the system show
- Are there any other ads that could yield good performance?
- Try all possible page-ad pairs enough times to decide
  - It has been done for display advertising!
  - Trying out costs money!
- Not feasible for even modest ad/page counts
- How to **explore** the space of ad placements in efficient way?
Explore-Exploit for CTR aggregation

Bandits for Taxonomies: A Model-based Approach S. Panday et al.
SDM 2007
Online Learning

Maximizing clicks requires:
- Dimensionality reduction
- Exploration
- Exploitation

Both must occur together

Online learning is needed, since the system must continuously generate revenue
Taxonomies for dimensionality reduction

- Already exist
- Actively maintained
- Existing classifiers to map pages and ads to taxonomy nodes

Learn the matching from page nodes to ad nodes \(\Rightarrow\) dimensionality reduction
Online Learning

Maximizing clicks requires:
- Dimensionality reduction
- Exploration
- Exploitation

Can taxonomies help in explore/exploit as well?
Outline

- Problem
- Background: Multi-armed bandits
  - Proposed Multi-level Policy
  - Experiments
  - Related Work
  - Conclusions
Background: Bandits

- Bandit “arms”
- $p_1$, $p_2$, $p_3$ (unknown payoff probabilities)

Pull arms sequentially so as to maximize the total expected reward

- Estimate payoff probabilities $p_i$
- Bias the estimation process towards better arms
Background: Bandits

Webpage 1

Webpage 2

Webpage 3

Bandit “arms” = ads

≈ 10⁹ pages

≈ 10⁶ ads
Background: Bandits

Content Match = A matrix
  • Each row is a bandit
  • Each cell has an unknown CTR
Bandit Policy
1. Assign priority to each arm
2. “Pull” arm with max priority, and observe reward
3. Update priorities
Why not simply apply a bandit policy directly to our problem?

- Convergence is too slow
  - \( \sim 10^9 \) bandits, with \( \sim 10^6 \) arms per bandit
- Additional structure is available, that can help
  ➤ Taxonomies
Outline

- Problem
- Background: Multi-armed bandits
- Proposed Multi-level Policy
  - Experiments
  - Related Work
  - Conclusions
Multi-level Policy

Consider only two levels
Multi-level Policy

Consider only two levels

Ad parent classes
Ad child classes
Block
One bandit
Multi-level Policy

Key idea: CTRs in a block are homogeneous
Multi-level Policy

- CTRs in a block are homogeneous
  - Used in allocation (picking ad for each new page)
  - Used in estimation (updating priorities after each observation)
Multi-level Policy

- CTRs in a block are homogeneous
  - Used in allocation (picking ad for each new page)
- Used in estimation (updating priorities after each observation)
Multi-level Policy (Allocation)

- Classify webpage ➔ page class, parent page class
- Run bandit on ad parent classes ➔ pick one ad parent class
Multi-level Policy (Allocation)

- Classify webpage ➔ page class, parent page class
- Run bandit on ad parent classes ➔ pick one ad parent class
- Run bandit among cells ➔ pick one ad class
- In general, continue from root to leaf ➔ final ad
Bandits at higher levels

• use **aggregated information**
• have **fewer bandit arms**

→ **Quickly figure out the best ad parent class**
Multi-level Policy

- CTRs in a block are homogeneous
  - Used in allocation (picking ad for each new page)
  - Used in estimation (updating priorities after each observation)
Multi-level Policy (Estimation)

- CTRs in a block are homogeneous
  - Observations from one cell also give information about others in the block
  - How can we model this dependence?
Multi-level Policy (Estimation)

- Shrinkage Model

\[ S_{\text{cell}} \mid \text{CTR}_{\text{cell}} \sim \text{Bin} (N_{\text{cell}}, \text{CTR}_{\text{cell}}) \]
\[ \text{CTR}_{\text{cell}} \sim \text{Beta} (\text{Params}_{\text{block}}) \]

All cells in a block come from the same distribution
Multi-level Policy (Estimation)

- Intuitively, this leads to *shrinkage* of cell CTRs towards block CTRs

\[
E[\text{CTR}] = \alpha \cdot \text{Prior}_{\text{block}} + (1-\alpha) \cdot S_{\text{cell}} / N_{\text{cell}}
\]

- Estimated CTR
- Beta prior ("block CTR")
- Observed CTR