Lecture 5: 
Content match and IR ad selection
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.
- First part of the lecture based on the slides of Prabhakar Raghavan and Chris Manning.
Lecture overview

- Content match basics
- Database approach to content match
- IR approach to content match
- Bridging content match and sponsored search
Course Overview

1. 09/24 Introduction
2. 10/01 Textual Advertising and Sponsored Search part 1
3. 10/08 Marketplace
4. 10/15 Sponsored Search part 2 and IR intro
5. 10/22 Content Match and IR ad retrieval
6. 10/29 Using click data for textual ad selection
7. 11/05 Display advertising
8. 11/12 Targeting
9. 11/19 Mobile/Social/Rich formats
10. 12/03 Project presentations
Query rewriting is one of the core methods for advanced match
Rewriting can be used to save results of elaborate ad selection
Online rewrites – the only option for the tail queries
Query fragment rewriting (e.g. ‘cheap’ → ‘inexpensive’)
  • Single rewrites apply to many queries
  • Apply the same techniques as q→q’ (random walks, PMI) but over queries
  • Denser data set
  • Deletions

IR
  • Document model (boolean, tf-idf)
  • Comparison operations (boolean, ranked similarity retrieval)
  • Efficient implementation with inverted indexes
Content Match Basics
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
Baidu Workers Call Off Strike

BEIJING – Workers at Baidu Inc. in two southern Chinese cities said they called off their strike Monday, although they said hadn’t reached any immediate agreement with the Internet search company over complaints about compensation.

Representatives of several hundred employees at the two southern offices of Baidu, which runs China’s top search site, finished meetings on Monday with executives from the company’s Beijing headquarters, including the head of human resources and an assistant to Robin Li, the company’s chief executive. They plan to give the company two to three days to come up with a fair response, and will decide...

TO CONTINUE READING, SUBSCRIBE NOW
The general interaction picture: Publishers, Advertisers, Users, & “Ad agency”
Interactions in Content Match

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

- **Users**
  - Interact with the content of the publisher’s web page

- **Publisher**
  - Provides the content, determines the placement of the ads
  - Paid either CPC or CPM

- **Ad network**
  - Selects the ads based on the information supplied by the publisher (and sometimes separate knowledge of the user)
How does it all work: the front end

- Two main approaches:
  1. Page fully built by publisher using ads supplied by the ad network.
     - E.g.: XML feed (Usually done with large partners.)
  2. Dynamic loading of ads:
Ad Network Utility

- Revenue
  - Long term
  - Short Term
- Short term revenue
  - Clicks
  - CPC – Cost per Click
- Long term revenue depends on:
  - Traffic volume and quality
  - Depth of market (number of advertisers)
  - Cost of serving
User Utility

- Browsing pages: Not as clear intent as with search queries
- Activities differ by the level of intent:
  - **Related resources**: reading one blog about gardening, would like to read another
  - **Moving through the shopping funnel**: Research ➔ Review ➔ Buy
  - **Lateral move**: Reading about golf, check out Rolex watches
- Ads should not interfere with the users’ activity
  - Position
  - Size
  - Content
Publisher Utility

• Short term income:
  • Ads are visible enough for the user to interact

• Long term income:
  • Do not impede on the user utility

• If user visits guaranteed (or independent of content) can place most ads
  • Spam

• Ad placement the strongest factor in user response:
  • CTR can vary by 2 orders of magnitude depending on the placement
Types of publishers

• “Conventional”
  • Owns content – advertising is main/only source of revenue
  • From small blogs to New York Times

• “Mixed model”
  • Have source of revenue, advertising is extra
  • Examples:
    • Small: “Ben’s bargains” – bargain hunting site, makes money from referrals + ads
    • Big: Amazon, E-bay
Music review in NYT + ads

Music

Note by Note, Manhattan Acquires a New Glow
Bounce

Ads by Google

Learn Music in San Mateo
Fun and Rewarding Music Lessons in Piano, Choir, Guitar, & Preschool
www.myriadmusic.net

Avloni Piano School
Silicon Valley Russian Teachers Piano Lessons, Competitions, Fun
avlonimusic.com

McMahon Jazz Medicine
Jazz, Classical, Rock CDs & Art by healers & legends. Great Rates!
www.McMahonJazzMedicine.com
Bargain hunting site + ads

**Ben’s Bargains**

**Deal:** Prince of Persia for Xbox 360 or PS3 $21 at GoGamer.com

- Discuss (2) | History | Tell | Posted 9:40 PM PDT 05/20/09 by Ben

GoGamer.com has Prince of Persia for PC priced at $18 + $3 shipping - $21 shipped. This reboot uses Ubisoft’s Anvil engine, and features a whole new combat system that you use to choose the Prince’s path in an open-ended game world.

**Check Out Dell Laptops**

- Compare Inspiron, Studio, Latitude, XPS & More. All of Intel Technology

**Post your comments:** To post comments, please Sign In or Register
Category browse on e-bay ("inside the home") + ads
Advertiser Utility: The Value Funnel

- **Value** = Long Term Profit
- What is the value of each event?
- **Immediate value** – profit of the current event (conversion and below)
- **Future value** – increase of the future profit due to the user action:
  - Ad impression might bring future conversions
  - Revenue events (upon user satisfaction) bring repeat customers
- Approximation: value declines by a linear coefficient as we move upwards
- **Content match was invented to increase volume for advertisers satisfied with sponsored search**
Overall Utility

• Combination of all the utilities – we can use the same principles as in Sponsored Search (e.g. linear convex combination)

• Relevance a good proxy. Still cases where relevance does not align all interests:
  • revenue vs. relevance (user vs. publisher and ad network)
  • volume vs. relevance (advertiser vs. the rest)

• Practice: similar as sponsored search
  • Maximize revenue, subject to two constraints:
    • Minimum user utility (relevance)
    • Minimum advertiser utility (price)
Some differences with Sponsored Search

- Coverage at 100% usually – do not want to leave empty slots on the page
  - Trade-off with display advertising
- Lesser role of the ad network, increased role of the publisher
  - Ad Network: which ads
  - Publisher: how many/where/how
- Ad selection using the content of a web page
  - Much more text
  - Less focused
  - Less intentional
Content match ad selection
Ad selection methods: what information is provided from the page

- Publisher can supply different information to the ad network
- **Page Content**
  - Process the content of the page
  - Cannot be done on-line: crawl
  - Most flexible from the ad selection perspective
- **Page Snippet**
  - Part of the page
  - How much can we process online?
  - How much is enough?
- **Custom Keywords**
  - Sponsored Search – like mechanism
  - Least flexibility in ad selection
  - More control for the publisher
Two main implementation strategies

- **Phrase extraction (from the publisher page)**
  - Map CM to Sponsored Search
  - Extract phrases from the page
  - Use these phrases to select ads (exact match or advanced match in Sponsored Search)
  - Ads selected on a single feature (phrase) from the page and the ad
  - Historically first approach

- **IR approach**
  - Treat CM as a *document similarity* problem
  - Pages are compared to the ads in corpus in a common feature space
  - Bid phrase one of the features used in matching
  - Ads selected based on multiple (overlapping) features of the page and threads
Web page analysis

- Standard web page processing: parsing, stemming, ...
- Content match ads present on a diverse set of pages:
  - Transactional pages, blogs, news, aggregators, ...
- Challenge: How to understand the content of such diverse set of pages?
Reminder: the ad

**Title**

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

**Creative**

**Display URL**

**Bid phrase:** computational advertising
**Bid:** $0.5

**Landing URL:** http://research.yahoo.com/tutorials/acl08_compadv/
Contextual Advertising  Ad Selection:  
Case studies

<table>
<thead>
<tr>
<th>Paper</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Finding Advertising Keywords on Web Pages. Wen-tau Yih et al. In Proc. of WWW 2006</td>
<td>phrase extraction ad selection</td>
</tr>
<tr>
<td>5. To Swing or not to Swing: Predicting when (not) to Advertise. Broder et al, CIKM 2008</td>
<td>various</td>
</tr>
</tbody>
</table>
Phrase Extraction for Contextual Advertising

Contextual Advertising by single feature

- Goal: given a page find phrases that are good for placing ads
- **Reverse search problem**: given a page, find the queries that would match (summarize) the content of this page
- Select ads based on a single selected keyword:
  - Contextual Advertising translated into database approach of Sponsored Search
  - Reuse of the Sponsored Search infrastructure – lower cost
  - Ad Networks earn less per impression in CA
    - Lower click-through rates (high-variance)
    - Lower conversion (less clear intent)
    - Lower earning per click: 1) lower bids; 2) revenue share with the publisher
System architecture

web page

Preprocessor
process html text

Candidate Selector
generate candidates

Classifier
score the candidates

Postprocessor
score \rightarrow probability

bid phrases
Preprocessor

- Translate HTML into plain text
- Preserve the blocks in the original document
- Preserve info about outgoing anchor text, meta tags
- Open source HTML parser for scraping – BeautifulSoup
- Part-of-Speech (POS) tagger – record the type of the word
- Chunker – detecting noun phrases
Candidate Selection

- All phrases of length up to 5 (including single words)
  - Within a single page block (sentence)
- Two dimensions of candidate selection:
  - Individual occurrences extracted separately vs. combining all occurrences into entry per page (*separate vs. combined*)
  - Consider the phrase as a whole
- Label individual words with their relationship with a phrase:
  - **Beginning** of a phrase
  - **Inside** a phrase
  - **Last** word of a phrase
  - ...

Classifier

- Given a phrase predict if it is “keyword” (usable for selecting ads)
- For the whole phrase (monolithic) case binary classifier
  - Logistic regression model \( P(Y = 1| x = \bar{x}) = \frac{1}{1 + e^{-xw}} \)
  - \( x \) is vector of features of a given phrase
  - \( w \) is a vector of importance weights learned from the training set
- Decomposed – multi label classifier (B,I,L,...)

\[
P(Y = 1| x = \bar{x}_i) = \frac{e^{xw_i}}{\sum_j e^{xw_j}}
\]
Features

- **Linguistic features:** is a noun; is a proper name; is a noun phrase; are all words in the phrase of the same type
- **Capitalization:** any/all/first word capitalization
- **Section based features:**
  - Hypertext – is the feature extracted from anchor text
  - Title
  - Meta tags
  - URL
- **IR features:** tf, idf, log(tf), log(idf), sentence length, phrase length, relative location in the document
- **Query log features:** log(phrase frequency), log(first/second/interior word frequency)
Score and feature reconciliation from multiple occurrences

- Binary features – OR of all occurrences
- Real valued features – min
- Score reconciliation: instance with the highest score
- Separate words $\rightarrow$ phrase probability:
  - $p_1 =$ probability of a phrase: product of the confidence of the classification of each term
  - $p_0 =$ probability of all the words of the phrase being outside a keyword
  - $\text{score} = \frac{p_1}{(p_1+p_0)}$
Experiments: Data

- 828 pages
- Indexed by MSN
- Have ads
- In the Internet Archive
- One page per domain
- Eliminate foreign and adult pages
- Editors (8) instructed to seek highly prominent keywords with advertising potential
Measuring the extraction quality

- Editorial judgments
- Precision-recall – might be too difficult
  - Too long for the judges to find all the relevant phrases
  - Given a phrase – influence the judges
- A proxy for P-R
  - top-1 = top-1 results is in the list selected by the editor, count across the set of pages
  - top-10 = % of top-10 results in the editor set, averaged over the set of pages
Main result

<table>
<thead>
<tr>
<th>system</th>
<th>top-1</th>
<th>top-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>MoC (Monolithic, Combined), -Lin</td>
<td>30.06\textsuperscript{b}</td>
<td>46.97\textsuperscript{b}</td>
</tr>
<tr>
<td>MoC (Monolithic, Combined), All</td>
<td>29.94</td>
<td>46.45</td>
</tr>
<tr>
<td>MoS (Monolithic, Separate), All</td>
<td>27.95</td>
<td>44.13\textsuperscript{c}</td>
</tr>
<tr>
<td>DeS (Decomposed, Separate), All</td>
<td>24.25\textsuperscript{c}</td>
<td>39.11\textsuperscript{c}</td>
</tr>
<tr>
<td>KEA [7]</td>
<td>23.57\textsuperscript{c}</td>
<td>38.21\textsuperscript{c}</td>
</tr>
<tr>
<td>MoC (Monolithic, Combined), IR</td>
<td>13.63\textsuperscript{c}</td>
<td>25.67\textsuperscript{c}</td>
</tr>
<tr>
<td>MoC (Monolithic, Combined), TFIDF</td>
<td>13.01\textsuperscript{c}</td>
<td>19.03\textsuperscript{c}</td>
</tr>
</tbody>
</table>

Table 1: Performance of different systems
Feature importance

<table>
<thead>
<tr>
<th>features</th>
<th>top-1</th>
<th>top-10</th>
<th>entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>A all</td>
<td>29.94(^b)</td>
<td>46.45(^b)</td>
<td>0.0113732(^b)</td>
</tr>
<tr>
<td>-C capitalization</td>
<td>30.11</td>
<td>46.27</td>
<td>0.0114219(^+)</td>
</tr>
<tr>
<td>-H hypertext</td>
<td>30.79</td>
<td>45.85(^+)</td>
<td>0.0114370</td>
</tr>
<tr>
<td>-IR IR</td>
<td>25.42(^+)</td>
<td>42.26(^+)</td>
<td>0.0119463(^\ddagger)</td>
</tr>
<tr>
<td>-Len length</td>
<td>30.49</td>
<td>44.74(^+)</td>
<td>0.0119803(^\ddagger)</td>
</tr>
<tr>
<td>-Lin linguistic</td>
<td>30.06</td>
<td>46.97</td>
<td>0.0114853(^\ddagger)</td>
</tr>
<tr>
<td>-Loc location</td>
<td>29.52</td>
<td>44.63(^+)</td>
<td>0.0116400(^\ddagger)</td>
</tr>
<tr>
<td>-M meta</td>
<td>30.10</td>
<td>46.78</td>
<td>0.0113633(^\ddagger)</td>
</tr>
<tr>
<td>-Ms meta section</td>
<td>29.33</td>
<td>46.33</td>
<td>0.0114031</td>
</tr>
<tr>
<td>-Q query log</td>
<td>24.82(^+)</td>
<td>42.30(^+)</td>
<td>0.0121417(^\ddagger)</td>
</tr>
<tr>
<td>-T title</td>
<td>28.83</td>
<td>46.94</td>
<td>0.0114020</td>
</tr>
<tr>
<td>-U URL</td>
<td>30.53</td>
<td>46.39</td>
<td>0.0114310</td>
</tr>
</tbody>
</table>

Table 3: The system performance by removing one set of features in the MoC framework
Conclusion

- Mapping Contextual Advertising to Sponsored Search
  - Extract phrases from the publisher’s web page
  - Select ads using exact or advanced match on this phrase
- Ad selection using a single feature
- Approach based on logistic regression trained on editorial judgments
  - Editors extracting salient terms from pages
- Combining the information from multiple occurrences and treating the phrases as single units yields best results
- IR and query log features account for almost all of the signal
- Low precision – difficult problem
IR methods for ad retrieval
In the beginning: The database approach

- **Thinking of SS as a data base problem**
  - `SELECT ads`  
    `FROM ad_table`  
    `WHERE bid_phrase = query`

- **Implementation**
  - Sponsored search
    - Match the query to the ad bid phrase (some normalization performed)
    - Advertisers cannot bid on all feasible queries (especially in the tail) \(\Rightarrow\) Need advanced match
  - Advanced match \(\Rightarrow\) translate the query into bid phrases
    - Very difficult to capture context, relevance, etc.
    - Pricing is misleading – bid on original phrase has little to do with value of AM
  - Content match \(\Rightarrow\) bid phrases from pages
    - very difficult to capture context, semantics, relevance, etc.
Finding the “best ad” as an Information Retrieval (IR) problem

• Treat the ads as documents in IR
  [Ribeiro-Neto et al. SIGIR 2005] [Broder et al. SIGIR2007] [Broder et al. CIKM2008]
  • Retrieve the ads by evaluating the query over the ad corpus
  • Use multiple features of the query and the ad

1. Analyze the “query” and extract query-features
  • Query = full context (content, user profile, environment, etc)
2. Analyze the documents (= ads) and extract doc-features
3. Devise a scoring function = predicates on q-features and d-features + weights
4. Build a search engine that produces quickly the ads that maximize the scoring function
Similarity to web search

- Ads have textual content and similar text processing pipeline can be used:
  - tokenization, stemming, stop words, entity extraction, etc.
- Inverted indexes can be used for scalable search with low latency
- The IR models of retrieval can be used to compare the query to ads
- Some of the properties of the landing pages can also be used in the search:
  - Anchor text
  - Static score (rank)
Differences from Web search

- **Web search:**
  - Large corpus
  - Reorder the pages that contain all the query terms

- **Ad retrieval:**
  - Smaller corpus
  - Similarity search rather than conjunction of the query terms: recall in the first phase important

- **What needs to be reevaluated:**
  1. **Query features (terms) and weighting?**
  2. **Document (unit of search) and cost of serving considerations?**
  3. **Balancing the visible and invisible parts of the ad?**
  4. **When (not) to advertise? Deciding when to show ads**
  5. **Economic considerations**
1. Query features and weighting

- Sponsored search: starting point is the web query
  - Limited recall

- Query expansion by using user’s past activities:
  - Browsing history
  - Search history
  - How far back? How to combine with current context

- Query expansion using web results [Broder et al. SIGIR2008]

- Content match – craft query from the page
  - Long queries – many features
  - What are query features: bid phrases only or unigrams/phrases?
2. Ads as documents – unit of search

- Given the ad schema, what should be the unit of search (document)?
- Where is the useful information in the ads?
  - Creatives are crafted as marketing message
  - Bid phrases are not visible to the user—fidelity issues
- Some options:
  1. Cross product of the creatives and bid phrases (the ad as the user sees it)
  2. Ad group as a document
  3. Subset of the ad group (cluster by content, classes, using display or landing page urls)
- Grouping more information might overcome the shortness of the ads
- However for heterogeneous ad groups this could increase the noise to signal ratio
2a. Document features and weighting

- There is no inherent document length as in classic documents
  - How to use the number of bid phrases/creatives per ad group
  - What is the signal in TF (local importance)?
  - DF (global importance)?
- Many ads are templates
  - Double counting
2b. Cost of serving considerations

- Document unit choice impacts the index size
  - Creative data can be duplicated
- Scoring combinations of creatives and bid phrases
  - One pass: score each creative-term combination
  - Two pass: score the unit in the first pass; score the creatives and terms in the second pass
- Efficient long query (similarity) algorithms
- Caching with long queries
  - Similarity for cache probes [Pandey et al. WWW2009]
  - Adapt the drift based on system workload
3. Balancing the visible and invisible parts of the ad

- Bid phrase ad creative relationship
  - Responsive, competitive, pragmatic, …
  - Determine weights for bid phrase and features derived from it

- Ad-landing pages relationship
  - In about 20-30% of cases landing page not related to the ad
  - Possible uses for the landing page:
    - Static score of the ad (or advertiser?)
    - Reweight existing ad features
    - Expand the ads with new features
4. When (not) to advertise

• In sponsored search ads not always shown
  • 30%-40% coverage today with most search engines

• How to determine when to advertise?
  • Commercial intent [Dai et al. WWW 2006]
  • Query difficulty
  • Quality of the available ads [Broder et al. CIKM 2008]
  • Quality of the web results
  • Interaction between ads and web results
  • …
5. Economic considerations

- Advertising is an economic activity with multiple participants
- Need to balance the utilities of all the participants
  - User utility - relevance
  - Advertiser utility – bid
- Balance the utilities by ordering ads by expected revenue: \( p(\text{click}) \times \text{bid} \)
  - \( p(\text{click}) \) proxy for relevance
- Recall very important: need enough depth (precision at lower rank) to preserve relevance through revenue reordering
IR methods for content match ad retrieval

Impedance coupling in content-targeted advertising. Ribeiro-Neto et al. SIGIR 2005
Using more than one feature in ad matching

- The phrase extraction approach uses one feature of the page (phrase) to select the ads.
- Risk with ambiguous phrases: ‘Tahoe’ is a destination as well as a truck model.
- Can we select ads based on multiple features from the page?
  - What are the features of the ad?
  - How to weight the features?
  - What metrics to use to relate the ads to the pages?
Formalism for comparing ads and pages: Vector Space Model

- Represent each ad $a$ as a vector: $a = \{w_{1a}, w_{2a}, \ldots, w_{na}\}$
  - In this study: $a$ is the visible part of the ad (title and abstract)
- Represent the page $p$ as a vector in the same space $p = \{w_{1p}, w_{2p}, \ldots, w_{np}\}$
- Weights using tf-idf method (last lecture)
- Use cosine of the angle between the vectors to rank the ads for a given page – denoted by $\text{sim}()$
Basic set of measures

- \( AD(p, a) = \text{sim}(p,a) \) – based on the visible parts of the ad
- \( KW(p,a) = \text{sim}(p, \text{kw}(a)) \) – based on the keyword of a
- \( AD\_KW(p, a) = \text{sim}(p, a \cup \text{kw}(a)) \) - using both the visible parts and the keyword

Assuming that \( \text{kw}(a) \) summarizes well the essence of \( a \), assure the presence of \( \text{kw}(a) \) in \( p \)

\[
ANDKW(p,a) = \begin{cases} 
\text{sim}(p,a) & \text{if } \text{kw}(a) \subseteq p \\
0 & \text{otherwise}
\end{cases}
\]

\[
AAK(p,a) = \begin{cases} 
\text{sim}(p,a \cup \text{kw}(a)) & \text{if } \text{kw}(a) \subseteq p \\
0 & \text{otherwise}
\end{cases}
\]
The Vocabulary impedance Problem

- Language and the topic of the page and the ad can differ substantially:
  - Publisher page belongs to a broader/narrower contextual scope
  - Ads concise in nature
  - ‘Hidden topic’ – not mentioned in the ad and/or the page
- Intersection of the vocabularies of related pages and ads can be low: vocabulary impedance problem
Solution: Impedance Coupling
Bayesian network model for page expansion using similar pages

Figure 3: Bayesian network model for our impedance coupling technique.

\[
P(T_i|R) = \frac{1}{P(R)} \sum_d P(T_i|d)P(R|d)P(d)
\]

\[
P(T_i|R) = \frac{\nu}{P(R)} \sum_{j=0}^{k} P(T_i|d_j)P(R|d_j)
\]
Page expansion, continued

\[ P(T_i|d_j) = \eta \ w_{ij} \]

\[ P(R|d_j) = \begin{cases} 
(1 - \alpha) & j = 0 \\
\alpha \ \text{sim}(r, d_j) & 1 \leq j \leq k 
\end{cases} \]

\[ P(T_i|R) = \rho \ ((1 - \alpha) \ w_{i0} + \alpha \ \sum_{j=1}^{k} \ w_{ij} \ \text{sim}(r, d_j)) \]

\[ \text{AAK}_T(p, a) = \text{sim}(r, a) \]
\[ \text{AAK}_\text{EXP}(p,a) = \text{AAK}(p \cup r, a) \]

feature selection for r: \( P(T_i|R)/P(T_{top}|R) > 0.05 \)
The choice of DF source

- Different level of granularity for a ‘document’ in the DF calculation:
  - ad
  - advertiser
  - campaign

Figure 4: Precision-recall curves obtained for the AD strategy using ad, advertiser, and campaign idf factors.
Page expansion results: it works

Figure 6: Impact of using a new representation for the triggering page, one that includes expansion terms.

Figure 8: Comparison among our ad placement strategies.
Ad expansion: landing page content is useful

Figure 7: Impact of using the contents of the page pointed by the ad (the hyperlink).
Summary

- Using IR techniques to match ads and pages
- Both the ad and the page are mapped to a common vector space
- Cosine of the angle between the ad and the page as the basic similarity measure
  - Bid phrase as a required feature – projection of the space
- Expanding pages using terms from similar pages improves results
- Landing page contains useful data for ad selection
- Some practical considerations:
  - How long are the queries?
  - How much is the cost of this method?
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

\[
C = \alpha \frac{1}{|D_r|} \sum_{\tilde{d} \in D_r} \tilde{d} + (1 - \alpha) \frac{1}{|D_{nr}|} \sum_{\tilde{d} \in D_{nr}} \tilde{d}
\]
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
  - pages

- Atomic snowboard
  - ads

match
Semantic and syntactic scores

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

- Syntactic component - term vector cosine
  \[ TaxScore(PC, AC) = \sum_{pc \in PC} \sum_{ac \in AC} \text{idist}(LCA(pc, ac), ac) \cdot cWeight(pc) \cdot cWeight(ac) \]
  \[ tWeight(kw^{si}) = weightSection(S_i) \cdot tf\_idf(kw) \]
  \[ KeywordScore(p_i, a_i) = \frac{\sum_{i \in |K|} tWeight(pw_i) \cdot tWeight(kw_i)}{\sqrt{\sum_{i \in |K|} (tWeight(pw_i))^2 \sqrt{\sum_{i \in |K|} (tWeight(aw_i))^2}}} \]
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
Pooling: using data from previous evaluation

- Faster turnaround, lower cost
- Essentially reordering of the prior results
- Could be off if the new method would select substantially different ads
- 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>
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 click data to improve IR ad retrieval

Relevance vs. Click Data

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

Click-based
- ML loss function
  - Maximum Entropy
  - (Semi)-Supervised
- Uses click data
- Probability of click

• No purely click-based systems deployed today (why)?
• Traditional combo: IR method for first phases retrieval, click data for reordering
Is TF-IDF the best we can do?

- Effective but crude method to establish word/phrase importance
- Efficient algorithms for evaluation of vector space model
- Vector space model does not give back a probability of click estimate
  - Needed for revenue estimation
- Can we improve quality by using click data while preserving the benefits?

This approach:
- Start with the relevance scoring formula
- Add parameters learned from click data
- Logistic regression to fit score into a probability function
Score vs. probability of click

- What is the relationship between the vector space score and the probability of click
  - ranking: $s_1 > s_2$
  - probability $s_1/s_2 \sim p(\text{click} \mid a_1, p)/p(\text{click} \mid q_2, p)$
- Can we translate the score to $p(\text{click})$?
  \[ p_{PA}(\text{click}) = \frac{1}{1 + e^{-f(\text{score})}} \]
  \[ f(\text{score}) = -1.75 + 2.52 \text{score} - 1.56 \text{score}^2 \]
- What if we use the click data to improve the score?
IR Score – Probability Relationship

- How can IR scores fit into the model?
  - What is the relationship between $\text{logit}(p_{ij})$ 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

$$\text{logit}(p_{ij}) = \sum_w \alpha_w M_{p,w} + \sum_w \beta_w M_{a,w} + \sum_w \delta_w I_{p,a,w} - 1.75 + 2.52 \times \text{cos} - 1.56 \times \text{cos}^2$$
Going further: improve the match

\[ p_{pA}(\text{click}) = \frac{1}{1 + e^{-f(p,A)}} \]

\[ f(p,A) = \sum_{w \in p} \text{tfidf}_{pw} \alpha_w + \sum_{w \in A} \text{tfidf}_{Aw} \beta_w + \sum_{w \in p \cap A} \text{tfidf}_{pw} \text{tfidf}_{Aw} \delta_w \]

- Insert learned parameters in the dot product
- Minimize Mean square error over training click data
- Evaluate over test data
- Correct the weights (log idf) of the words.
  - Example: “ball”
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
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:
    \[ i_w = \frac{CTR_w^{both}}{CTR_w^{page} \cdot CTR_w^{ad}} \]
  • Unsupervised: tf-idf intensities

• Click data based approach yields better results

• How many is enough?
  • 1K features give the same result with 3K features
Efficient Training

- Fast Implementation
- Training: Parallel implementation of Logistic Regression

Diagram:
- 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
Ad set language model

- Language model: relative frequency of words conditioned on a given query:

\[ \theta_w = \sum_{A \in Ads} P(w \mid A)P(A \mid Q) \]

\[ P(w \mid A) = \frac{tf_{w,A}}{|A|} \]

\[ P(A \mid Q) = \frac{\text{score}(q,A)}{\sum_{A' \in ads(q)} \text{score}(q,A')} \]
Clarity and entropy of the language model as features

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

\[ D_{KL}(P,Q) = H(P,Q) - H(P) = \sum_j p(j) \log(q(j)) - \sum_j p(j) \log(p(j)) \]

\[ CLARITY(\theta) = D_{KL}(\theta,\hat{\theta}) \]

\[ \hat{\theta}_w = \frac{tf_w}{|Corpus|} \]

- Intuition: how much is the distribution of words different from noise (aggregate over all ads)
- Entropy: in every domain there is a set of core words that describe the domain
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
When to advertise – act 2

Estimating Advertisability of Tail Queries for Sponsored Search, SIGIR 2010, Panday et al
Swing in Sponsored Search

Query Selection → Ad Selection

Matching ads

Query

next few slides

unadvertisable queries

ad index
Sponsored Search

- Two steps of ad matching:
  
  - **Step 1 (Query selection):** Select queries on which to advertise
    
    e.g., queries with commercial intent [Dai et. al. WWW’06, Ashkan et. al. CIKM’09 and SIGIR’09]
  
  - **Step 2 (Ad selection):** Find matching ads for the selected queries
    
    e.g., ads offering products/services directly related to the query [Ribeiro-Neto SIGIR’05, Richardson et. al. WWW’07]
Query Selection Task

- Definition of Advertisability:
  - Ad-clickability: likelihood of the query to result in one or more ad-clicks
  - many definitions, e.g., commercial intent
    - subjective definition, needs editorial judgments

- Induces a straight-forward evaluation metric:
  - fraction of clicks obtained from a given budget of queries; the higher the better
Query Selection Task

- Relatively easy for head queries:
  - compute clickability estimate from historical data
  - perform complex offline analysis, for example:
    - find matching ads and gauge their quality [Broder et. al. CIKM’08]

- Our focus: tail queries (constitute 40% search traffic)
  - the selection mechanism must be online (i.e., decide at runtime)
  - must be simple, in terms of running time, model size and training requirements

We estimate query clickability using its keywords only
Word-Based Advertisability Model

- **Problem definition:** find query clickability using query keywords
  - \( c(q) \) denotes the clickability of query \( q \), i.e., likelihood of attracting an ad click
  - \( c(w) \) denotes the clickability of word \( w \)

- How does a word contribute towards query clickability?

- **Desired Properties:**
  - **P1:** Any one word in query can make \( c(q) \) go high
    - e.g., cheap, download, rent, music
  - **P2:** No one word can make the query unclickable
    - e.g., weather, university
Word-Based Advertisability Model

• **Model:** each word draws an ad-click independently based on its clickability score $c()$

  \[ c(q) = 1 - \prod_{i=1}^{n} (1 - c(w_i)) \]

  - **P1:** Any one word in query can make $(q)$ go high
  - **P2:** No one word can make the query unclickable

• To avoid favoring long queries:

  \[ c(q) = \max_S \left( 1 - \prod_{w \in S} (1 - c(w)) \right) \]

  where $S$ is any $k$-size subset of the query
Estimating word scores $c(w)$

- Using Maximum-Likelihood approach, for every $c(w)$ we get:

$$\frac{\sum_{q \ni w} n(q)}{(1 - c(w))} = \sum_{q \ni w} \left( s(q) \cdot \frac{\prod_{w' \in q, w' \neq w} (1 - c(w'))}{1 - \prod_{w' \in q} (1 - c(w'))} \right)$$

- Under the assumption that:
  - each query is a referendum on each word in the query independently

$$c(w) = \frac{\sum_{q \ni w} s(q)}{\sum_{q \ni w} (s(q) + n(q))}$$

# instances of q that were clicked

# instances of q that were not clicked
Experiments

• **Data**
  - Sample of real search engine queries with < 2 impressions per day (i.e., tail set)
  - 2 million unique such queries (avg length 3.3 words)

• **Task:** rank these queries by clickability

• **Metric:** evaluate the ranked list by
  • fraction of clicks obtained in a given budget of query impressions
  • the higher, the better
Word-Based Advertisability Model

The diagram illustrates the fraction of clicks as a function of the fraction of impressions, comparing different models:
- word-based model (k=1)
- word-based model (k=2)
- word-based model (k=3)
- word-based model (k=100)
- random
Summary

• Define query advertisability in terms of ad-clickability
  • avoids editorial judgments

• Estimate clickability using query keywords only
  • applies on tail queries
  • proposed model has certain desired properties

• Evaluation on real data
  • compare against regression (with regularization and clustering)
  • our model performs significantly better
Search-based ad selection for sponsored search

An alternative view of Search Advertising

• A lesson from Content Match
  • View Search Advertising as CA on the web search result page
  • More general: use the web search results as a basis for ad selection

• What are the benefits?
  • Uniform look of the result page – improved user experience
  • Re-use of the web search technology
  • Circumstantial evidence for Search Advertising

• The approach
  • Web search results as (pseudo) feedback for the web search query
  • Expanded web search query used as a long ad query
  • Evaluate the ad query to select the ads
System Overview

Search query → Web Search Engine

- Page Classifier
  - Class selection
    - Query Categories
- Unigram extraction
  - Unigram selection
    - Query unigrams
- Phrase extraction
  - Phrase selection
    - Query phrases

Feature X extraction
- Feature selection
  - Query feature X

Query Generation

- Feature extraction
  - Index build
    - Ad Search Engine

Ads

Ad
Where to look for features?

- Snippets or full pages?
- Aggregation: bundling or voting?
- Number of search results to obtain
- Number of features per search result
Feature construction basics

- Use top 40 search results to extract features
- Unigrams
  - Bag of words
  - Porter stemming
- Phrase extraction
  - Corpus of 10M phrases, based on co-occurrence in web page
- Weighting
  - tf-idf based
  - Document frequency from an ad corpus
  - Ad side tf: weighted by ad sections (title, body, bid phrase, …)
  - Page side tf: weighted by page sections (title, meta, body, …)
- Feature selection
  - result set df (r-df)
  - sum of tf-idf scores from individual pages
Ad search

- Scalable inverted index building
  - Use grid computing framework
  - What is a the document unit?
- Similarity search with long queries (100-200 features)
  - WAND algorithm
- Scoring function: linear combination of phrase, class and unigram scores
  
  $$\text{score}(q,a) = \alpha \sum uq_i u_a + \beta \sum cq_i c_a + \gamma \sum pq_k p_a$$
Experimental setup: sampling queries

- How to make a query sample that represents the query stream?
  1. Sample from query occurrences
  2. Sample from unique queries

- To see how well a system performs, we need a representative sample of the traffic:
  - Sample from occurrences, dedup, keep counts

- However, we need to avoid over-fitting for a small number of queries

- This example: stratified sampling
  - 50 queries per decile
  - 200 tail queries (beyond the first 10M)

- Relevance judgments for the query-ad pairs
Precision-Recall

![Precision-Recall Graph]

Score vs. Recall levels for Onyx P@1, Onyx P@3, Baseline P@1, and Baseline P@3.
Conclusion

- Cross-corpora query expansion
  - Query + web results = ad query
- Ad selection using the web search results (vs. only the query)
  - Content Advertising and Search based approach
  - Improved user experience by better synchronization between the web and ad search results
- Beyond bid phrase based selection
  - Use the complete ad content for ad selection
- Promising experimental results
Contextual Advertising - summary

- One of the two textual advertising channels on the web
- Supports a large swath of the web eco system
- Three types of ad placement mechanisms:
  - Phrase extraction form the publisher pages
  - IR-style matching of the page content to the ads
  - Next class: click-data methods for improving the selection of ads
    - Exploration of unseen combinations
- Industrial systems likely using a combination of technologies
- Space for improvement in today’s state-of-the-art
Thank you!

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

http://research.yahoo.com