A Dynamic Bayesian Network Click Model for Web Search Ranking

Olivier Chapelle and Anne Ya Zhang

YAHOO! RESEARCH

Apr 22, 2009
18th International World Wide Web Conference

Introduction

Motivation

- Clicks provide valuable implicit relevance feedback.
- More abundant and cheaper than editorial judgments.
- For learning a web search function, clicks have been used as a *target* [Joachims '02] or as a *feature* [Agichtein et al. '06].

Main difficulty: *Presentation bias*

Results at lower positions are less likely to be clicked even if they are relevant.

Introduction

Motivation

- Clicks provide valuable implicit relevance feedback.
- More abundant and cheaper than editorial judgments.
- For learning a web search function, clicks have been used as a *target* [Joachims '02] or as a *feature* [Agichtein et al. '06].

Main difficulty: *Presentation bias*

Results at lower positions are less likely to be clicked even if they are relevant.

Problem statement

Unbias the clicks = "What would have been the click-through rate (CTR) of a given url had it been shown in first position."

Goal is not click modeling per se.

Two main types of click models to unbias the click logs:

1. Position based models
2. *Cascade* model [Craswell et al. '08]

Position based models are widely used but cascade models turn out to be more accurate.

We propose an extension of the cascade model and apply it to the problem of learning a ranking function.

Outline

Outline

Position based model

Naive approach: unbias by computing the aggregated CTRs at various positions.

Examination model

User clicks $\Leftrightarrow$ examines the snippet and finds it attractive.
Probability of examination depends only on the position.

$$P(C = 1|u, p) = a_u b_p.$$

- Estimation through maximum likelihood.
- Leverages variations in the search results to isolate the position effect.

Variation: *logistic model*

$$P(C = 1|u, p) = (1 + \exp(-a_u - b_p))^{-1}.$$

Unconstrained and jointly convex problem: easier optimization.

Position based model

Naive approach: unbias by computing the aggregated CTRs at various positions.

Examination model

User clicks $\Leftrightarrow$ examines the snippet and finds it attractive.
Probability of examination depends only on the position.

$$P(C = 1|u, p) = a_u b_p.$$

- Estimation through maximum likelihood.
- Leverages variations in the search results to isolate the position effect.

Variation: *logistic model*

$$P(C = 1|u, p) = (1 + \exp(-a_u - b_p))^{-1}.$$

Unconstrained and jointly convex problem: easier optimization.

Position based model

Naive approach: unbias by computing the aggregated CTRs at various positions.

Examination model

User clicks $\Leftrightarrow$ examines the snippet and finds it attractive.
Probability of examination depends only on the position.

$$P(C = 1|u, p) = a_u b_p.$$

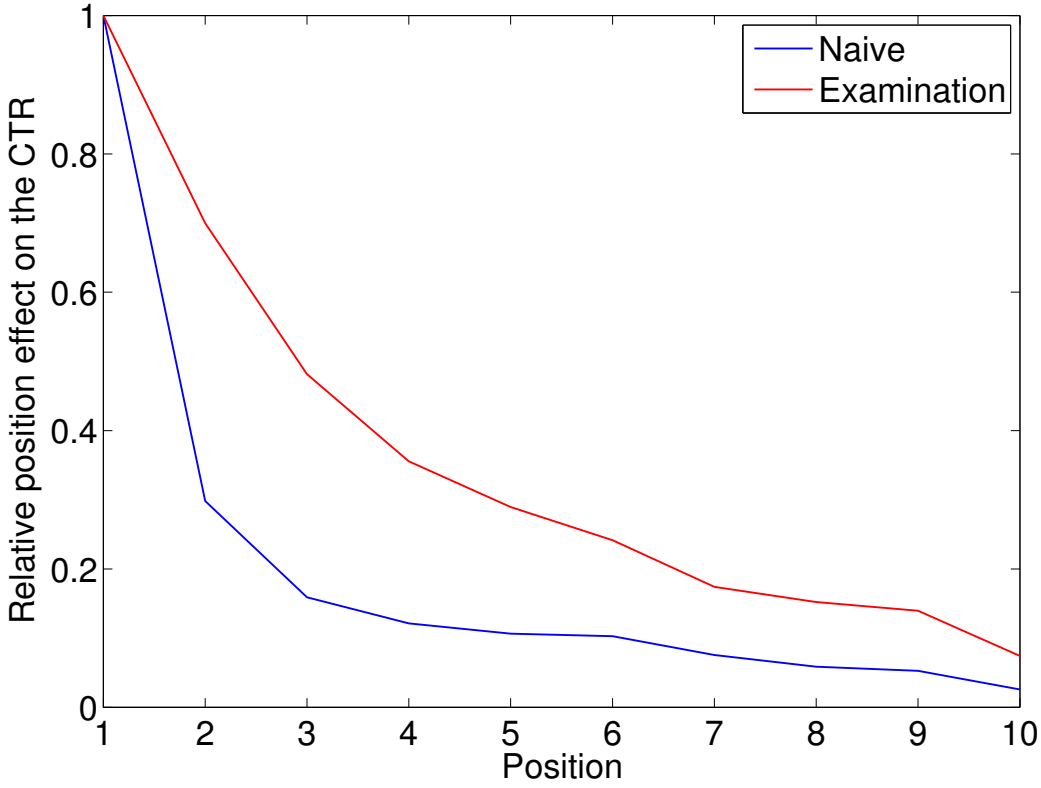
- Estimation through maximum likelihood.
- Leverages variations in the search results to isolate the position effect.

Variation: *logistic model*

$$P(C = 1|u, p) = (1 + \exp(-a_u - b_p))^{-1}.$$

Unconstrained and jointly convex problem: easier optimization.

Typical results for position based model (normalized at 1 for position 1):



Example: under the examination model, moving a document from position 1 to 3 will reduce its CTR by about 50%.

Problem with position based models

- Example
  - Query = myspace
  - Url = www.myspace.com
  - Market = UK
- Ranking 1
  - Pos 1: uk.myspace.com, ctr = 0.97
  - Pos 2: www.myspace.com, ctr = 0.11
- Ranking 2
  - Pos 1: www.myspace.com, ctr = 0.97

Position based models cannot explain such a variation.

Underlying problem: the probability of examination depends more on the relevance of the documents above than on the position.

Cascade model

Introduced by Craswell et al. (WSDM 2008).

1: $i = 1$
2: User examines position i and sees url u.
3: **if** random(0,1) $\leq a_u$ **then**
4: User clicks in position i and stops.
5: **else**
6: $i \leftarrow i + 1$; go to 2
7: **end if**

- User does a top-dwon scan of the documents and stops when he finds a relevant one.
- Probability of a click in position 3: $(1 - a_{u_1})(1 - a_{u_2})a_{u_3}$.
- Limitation: exactly one click per session (by definition).
- But already outperforms position based models. (correctly handles the myspace example)

Cascade model

Introduced by Craswell et al. (WSDM 2008).

1: $i = 1$
2: User examines position i and sees url u.
3: **if** random(0,1) $\leq a_u$ **then**
4: User clicks in position i and stops.
5: **else**
6: $i \leftarrow i + 1$; go to 2
7: **end if**

- User does a top-dwon scan of the documents and stops when he finds a relevant one.
- Probability of a click in position 3: $(1 - a_{u_1})(1 - a_{u_2})a_{u_3}$.
- Limitation: exactly one click per session (by definition).
- But already outperforms position based models.
  (correctly handles the myspace example)

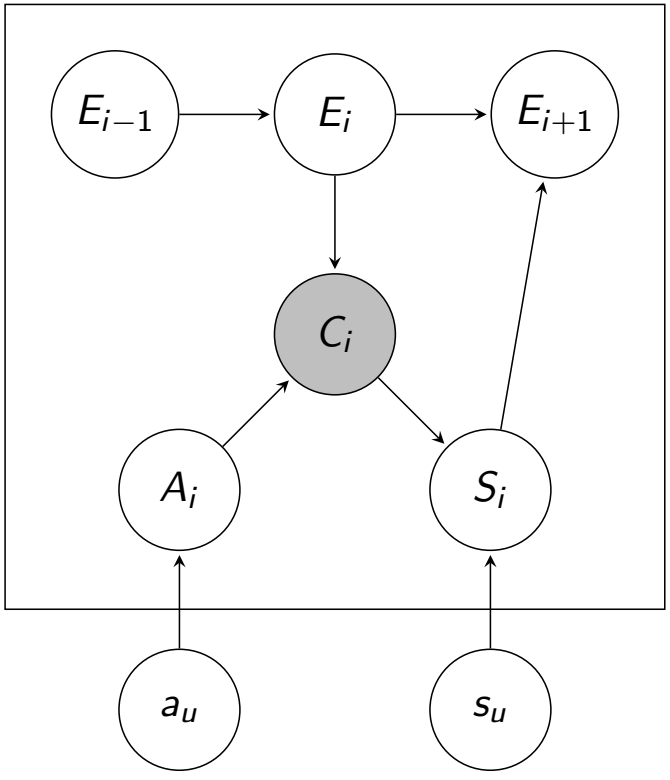
Proposed extension

Two main differences:

1. After clicking, there is some probability that the user is not *satisfied* and goes back to the search results.
2. The user may *abandon* his search before finding a relevant result.

$\longrightarrow$ Allows for 0 or more than 1 click per session.

Dynamic Bayesian Network for the cascade model



$$E_1 = 1$$
$$E_i = 0 \Rightarrow E_{i+1} = 0$$
$$A_i = 1, E_i = 1 \Leftrightarrow C_i = 1$$
$$P(A_i = 1) = a_u$$
$$P(S_i = 1 | C_i = 1) = s_u$$
$$C_i = 0 \Rightarrow S_i = 0$$
$$S_i = 1 \Rightarrow E_{i+1} = 0$$
$$P(E_{i+1} = 1 | E_i = 1, S_i = 0) = \gamma$$

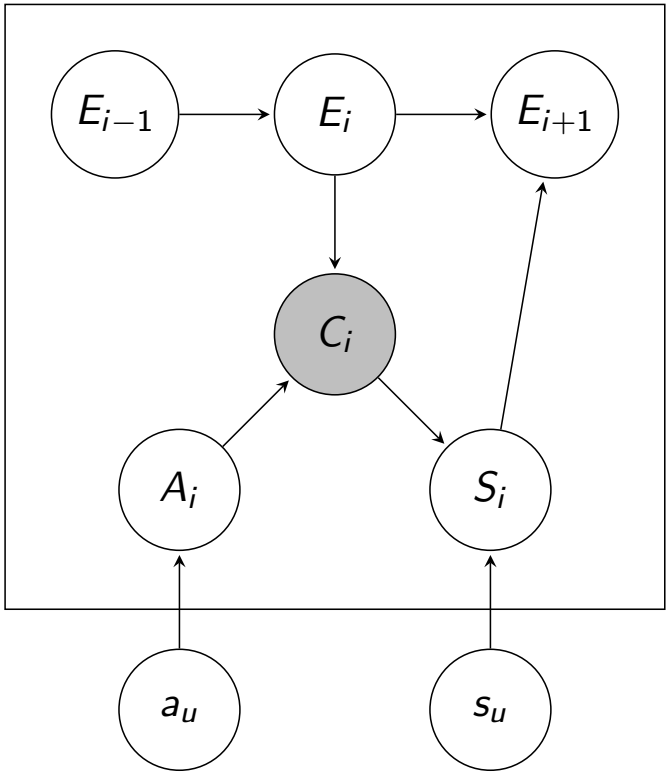
For each position i, 4 binary variables $E_i, A_i, S_i, C_i \in \{0, 1\}$

E_i: did the user *examine*?

A_i: was the user *attracted*?

S_i: was the user *satisfied*?

Dynamic Bayesian Network for the cascade model



$$E_1 = 1$$
$$E_i = 0 \Rightarrow E_{i+1} = 0$$
$$A_i = 1, E_i = 1 \Leftrightarrow C_i = 1$$
$$P(A_i = 1) = a_u$$
$$P(S_i = 1 | C_i = 1) = s_u$$
$$C_i = 0 \Rightarrow S_i = 0$$
$$S_i = 1 \Rightarrow E_{i+1} = 0$$
$$P(E_{i+1} = 1 | E_i = 1, S_i = 0) = \gamma$$

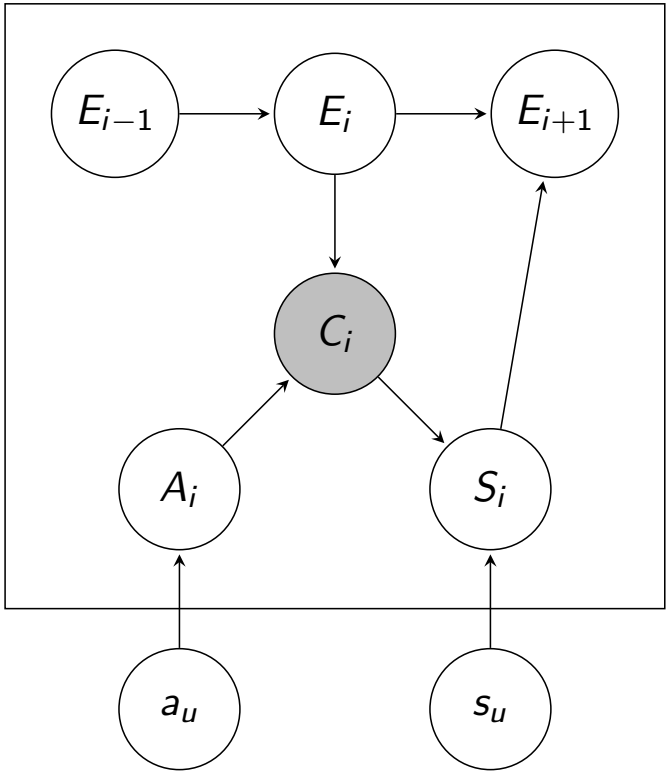
For each position i, 4 binary variables $E_i, A_i, S_i, C_i \in \{0, 1\}$

E_i: did the user *examine*?

A_i: was the user *attracted*?

S_i: was the user *satisfied*?

Dynamic Bayesian Network for the cascade model



$$E_1 = 1$$
$$E_i = 0 \Rightarrow E_{i+1} = 0$$
$$A_i = 1, E_i = 1 \Leftrightarrow C_i = 1$$
$$P(A_i = 1) = a_u$$
$$P(S_i = 1 | C_i = 1) = s_u$$
$$C_i = 0 \Rightarrow S_i = 0$$
$$S_i = 1 \Rightarrow E_{i+1} = 0$$
$$P(E_{i+1} = 1 | E_i = 1, S_i = 0) = \gamma$$

a_u measures the *perceived* relevance

s_u is the "ratio" between *actual* and perceived relevance

$P(S_i = 1 | E_i = 1) = a_u s_u$.

Inference

The model is similar to an HMM (Hidden Markov Model) and can be trained in the same way using the Expectation-Maximization algorithm (EM):

E-Step Given a_u, s_u and γ, compute the posterior probabilities on A_i, E_i and S_i. This involves the forward-backward algorithm.

M-Step Given the posterior probabilities, update a_u, s_u and γ (simple counting).

Note that the inference for different queries is decoulped.

Throughput: $\sim$ 10k sessions per second.
$\longrightarrow$ Processing several months of data on a Map-Reduce cluster typically takes several hours.

Extension to other clicks

Introduce two virtual nodes to capture clicks that are not on the search results:

- **Position 0** Anything at the top of the search result page (sponsored ads, spelling suggestion)
- **Position 11** Anything at the bottom (next page button)

Outline

Experimental setup

Session definition

Unique user and query, finished by 60 minutes idle time.

- Consider only first page of results.
- Ignore the sessions for which the clicks are not in the same order of the ranking.
- Kept only queries for which we have at least 10 sessions.
- One month of UK data.
- 682k unique queries and 58M sessions.

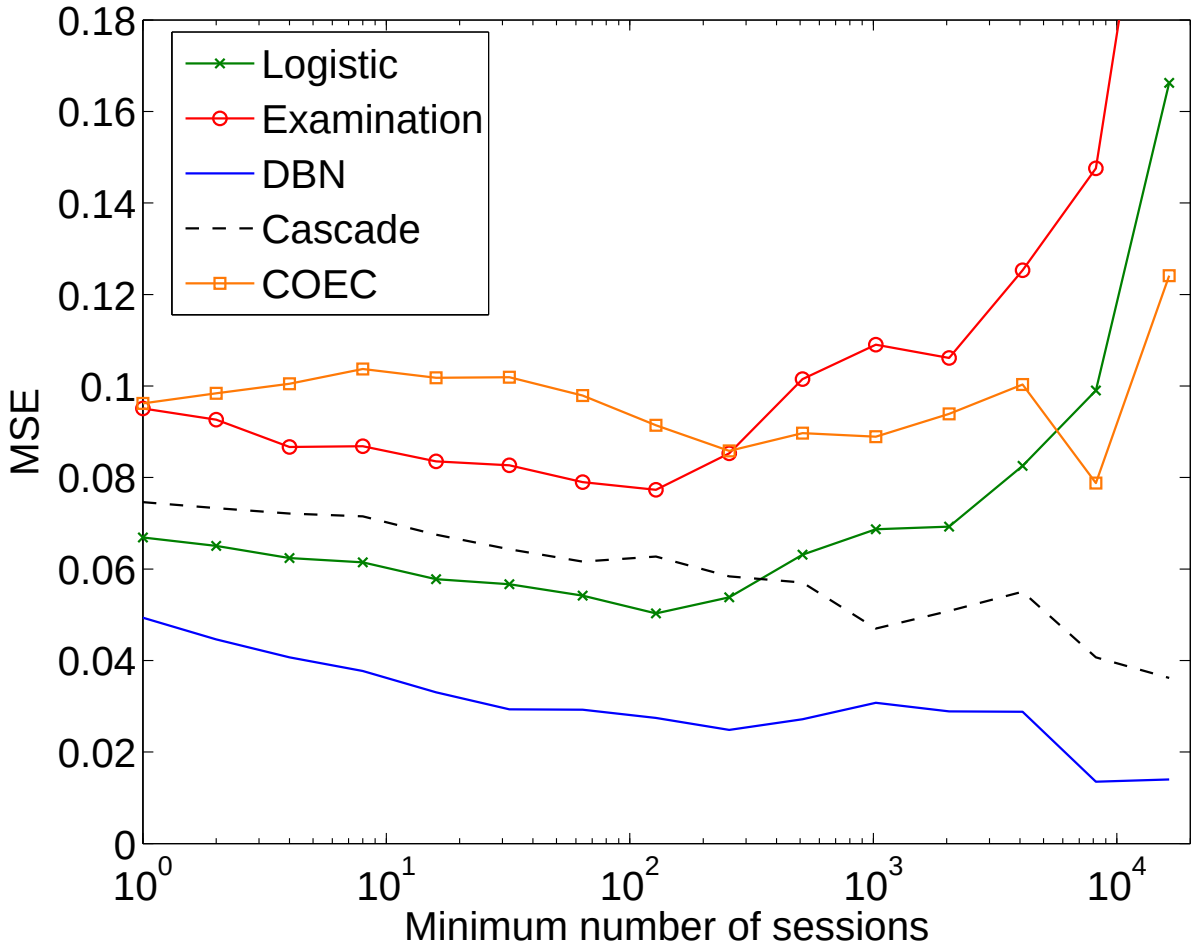
Experiment (I)

Predict *attractiveness*, i.e. CTR@1

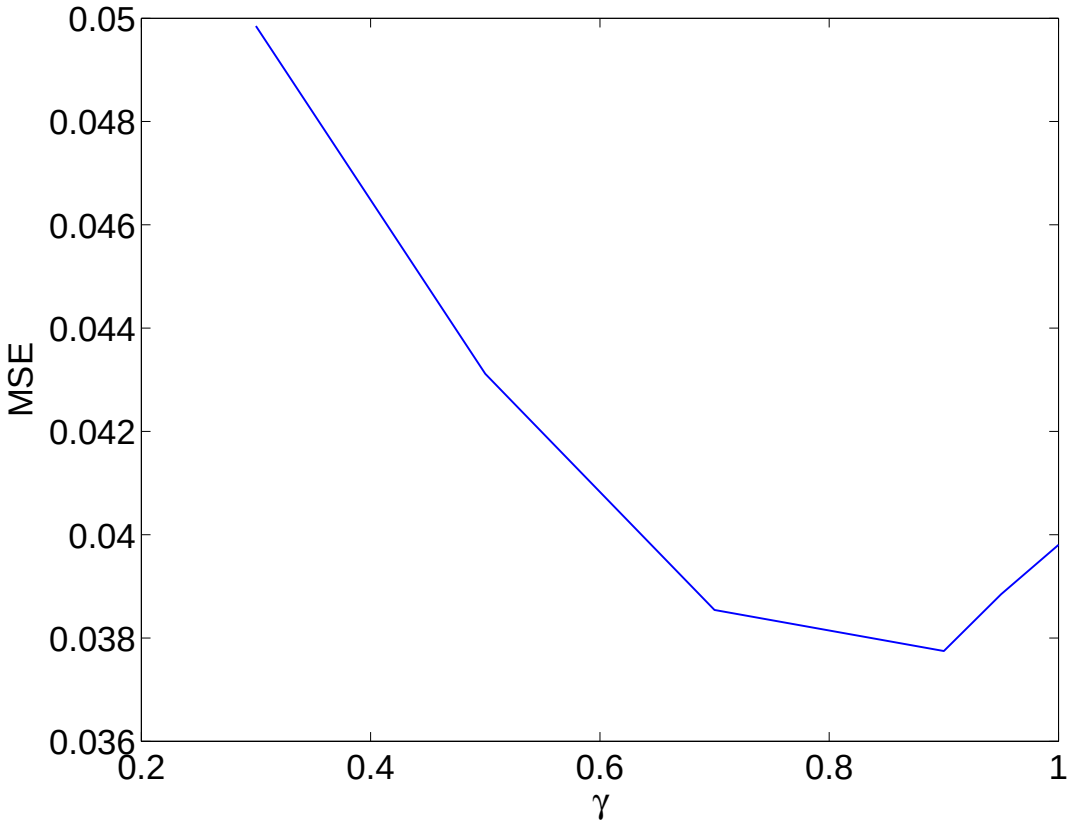
Leave-first-position-out

1. Retrieve all the sessions related to a given query
2. Consider an url that appeared both in position 1 and other positions
3. Hold out as test the sessions in which that url appeared in position 1
4. Train the model on the remaining sessions and predict a_u
5. Compute the test CTR in position 1 on the held-out sessions
6. Compute an error between these two quantities
7. Average the error on all such urls and queries, weighted by the number of test sessions.

MSE as a function of the minimum number of training sessions: left = all queries; right = head queries.



Influence of γ



$\longrightarrow$ Users are persistent.

Experiments (II)

Use the predicted relevance $a_u s_u$ for ranking:

1. Simply rank according to the predicted relevance. NDCG comparison computed for urls with at least 10 sessions and queries with at least 10 urls:

Logistic	0.705	-7.8%
Cascade	0.73	-4.6%
DBN – 10 nodes	0.748	-2.2%
DBN – 12 nodes	**0.765**	–
DBN – 12 nodes (a_u only)	0.756	-1.2%

   Machine learned ranking function trained with hundreds of ranking features: 0.795 (+3.9%)

2. Add predicted relevance as a *feature* in the machine learned ranking function: +0.8%. Feature is one of the top 10 most important ones.

Experiments (II)

Use the predicted relevance $a_u s_u$ for ranking:

1. Simply rank according to the predicted relevance. NDCG comparison computed for urls with at least 10 sessions and queries with at least 10 urls:

Logistic	0.705	-7.8%
Cascade	0.73	-4.6%
DBN – 10 nodes	0.748	-2.2%
DBN – 12 nodes	**0.765**	–
DBN – 12 nodes (a_u only)	0.756	-1.2%

   Machine learned ranking function trained with hundreds of ranking features: 0.795 (+3.9%)

2. Add predicted relevance as a *feature* in the machine learned ranking function: +0.8%. Feature is one of the top 10 most important ones.

Experiments (III)

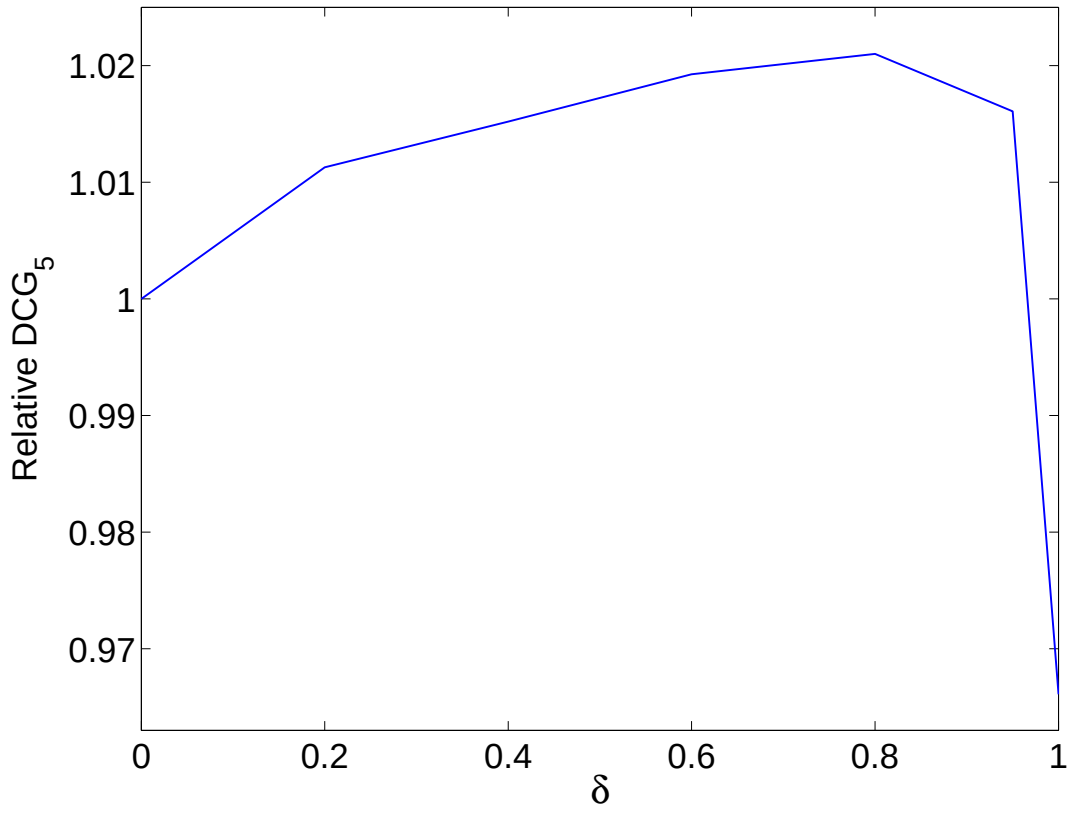
Use predicted relevance as a *target*: useful for markets with few editorial judgments.

Technique: boosting decision trees trained on pairwise preferences. 2 sets of preferences: $\mathcal{P}_E$ from editorial judgments and $\mathcal{P}_C$ coming from the click modeling. Minimize:

$$\frac{1-\delta}{|\mathcal{P}_E|} \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in\mathcal{P}_E} \phi(f(\mathbf{x}_i) - f(\mathbf{x}_j)) + \frac{\delta}{|\mathcal{P}_C|} \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in\mathcal{P}_C} \phi(f(\mathbf{x}_i) - f(\mathbf{x}_j)).$$

with $\phi(t) = \max(0, 1-t)^2$.

DCG relative to $\delta = 0$.
Left: only editorial judgments; right: only clicks.



About 2% gain by combining clicks and editorial judgments.

Outline

A simplified model

Optimal $\gamma = 0.9$, but $\gamma = 1$ is almost as good.
In that case, inference is much easier because we know that the urls after the last click have not been examined.

Simple counting algorithm for estimating a_u and s_u:

$$a_u = \frac{\#\text{ clicks}}{\#\text{ views}}, \quad \text{view} := \text{clicked or } \exists \text{ click below.}$$

$$s_u = \frac{\#\text{ last clicks}}{\#\text{ clicks}}.$$

Extremely simple and efficient to alleviate the position bias
$\longrightarrow$ Never use standard CTRs anymore!

A simplified model

Optimal $\gamma = 0.9$, but $\gamma = 1$ is almost as good.
In that case, inference is much easier because we know that the urls after the last click have not been examined.

Simple counting algorithm for estimating a_u and s_u:

$$a_u = \frac{\#\text{ clicks}}{\#\text{ views}}, \quad \text{view} := \text{clicked or } \exists \text{ click below.}$$

$$s_u = \frac{\#\text{ last clicks}}{\#\text{ clicks}}.$$

Extremely simple and efficient to alleviate the position bias
$\longrightarrow$ Never use standard CTRs anymore!

Editorial judgments vs clicks

About 20% disagreement rate between the preferences based on clicks and on editorial judgments.

We asked editors to look at these contradicting pairs and assign of the following 6 reasons:

1. Errors in editorial judgment
2. Imperfect editorial guidelines
3. Popularity is not necessarily aligned with relevance
4. Clicks measure mostly perceived relevance, while editors judge the relevance of the landing page
5. Click model is not accurate
6. Other

Editorial judgments vs clicks

About 20% disagreement rate between the preferences based on clicks and on editorial judgments.

We asked editors to look at these contradicting pairs and assign of the following 6 reasons:

1. Errors in editorial judgment 33.3%
2. Imperfect editorial guidelines 24.6%
3. Popularity is not necessarily aligned with relevance 15.2%
4. Clicks measure mostly perceived relevance, while editors judge the relevance of the landing page 19.4%
5. Click model is not accurate 4.7%
6. Other 2.8%

Examples: Popularity $\neq$ "relevance"

First url preferred by the clicks, second one by the editors.

Adobe

`www.adobe.com/products/acrobat/readstep2.html`

`www.adobe.com`

Hsbc

`www.hsbc.co.uk/1/2/personal/internet-banking`

`www.hsbc.co.uk`

Paris Hilton

`www.maximonline.com/girls_of_maxim/`

`en.wikipedia.org/wiki/Paris_Hilton`

Rush Hour 3

`www.imdb.com/title/tt0293564`

`www.rushhourmovie.com`

Examples: perceived vs actual relevance

- Spam pages
- Quality of the url / abstract
- .com preferred to .co.uk
- Domain trust

Cricket scores

news.bbc.co.uk/sport2/hi/cricket/default.stm

www.cricinfo.com

Hotmail

www.hotmail.com

newhotmail.co.uk

Travel insurance

www.directline.com

www.travelandinsurance.com

Examples: perceived vs actual relevance

- Spam pages
- Quality of the url / abstract
- .com preferred to .co.uk
- Domain trust

Cricket scores

news.bbc.co.uk/sport2/hi/cricket/default.stm

www.cricinfo.com

Hotmail

www.hotmail.com

newhotmail.co.uk

Travel insurance

www.directline.com

www.travelandinsurance.com

Examples: perceived vs actual relevance

- Spam pages
- Quality of the url / abstract
- .com preferred to .co.uk
- Domain trust

Cricket scores

news.bbc.co.uk/sport2/hi/cricket/default.stm

www.cricinfo.com

Hotmail

www.hotmail.com

newhotmail.co.uk

Travel insurance

www.directline.com

www.travelandinsurance.com

Examples: perceived vs actual relevance

- Spam pages
- Quality of the url / abstract
- .com preferred to .co.uk
- Domain trust

Cricket scores

`news.bbc.co.uk/sport2/hi/cricket/default.stm`

`www.cricinfo.com`

Hotmail

`www.hotmail.com`

`newhotmail.co.uk`

Travel insurance

`www.directline.com`

`www.travelandinsurance.com`

Conclusion

- Cascade models are better than position based models to handle the position bias problem.
- Extension of the cascade model by allowing the user to 1) give up and 2) continue searching after a click (by introducing the notion of satisfaction).
- Simplified version of this model turns out to be almost as good and only involves simple counting.
- Estimated relevance is helpful both as a target and as a feature. But if test metric is editorial, improvements can be limited due to intrinsic differences between clicks and editorial judgments.
- Current work: use the click duration to model the satisfaction.

Differences with the next talk

Next talk, *Click Chain Model in Web Search*, is strikingly similar to our work: the authors also construct a graphical model to extend the basic model and allow for any number of clicks.

Some key differences:

1. Notion of satisfaction
2. Approximate vs exact inference
3. Evaluation