A Novel Click Model and Its Applications to Online Advertising

Zeyuan Zhu
Weizhu Chen
Tom Minka
Chenguang Zhu
Zheng Chen
Introduction

Click Model - To model the user behavior

Application

- Predict CTR
  - Improve NDCG
  - AdPrediction
  - ...
- Document relevance estimation
  - Replace human judged data
  - As ranking features.
  - ...

Clicks are *biased*

- presenting order
  - ...

February 5, 2010
Related Works

**examination hypothesis (position model)**

- Observation: The relevance of a document at position $i$ should be further multiplied by a term $x_i$.

**cascade model**

- Observation: user scans from top to bottom – a Bayesian network.
Related Works

Examination Hypothesis
if a displayed url is clicked, it must be both examined and relevant
query $q$; url $u$; position $i$; binary click event $C$

$$P(C = 1|q, u, i) = P(C = 1|u, q, E = 1) \cdot P(E = 1|i)$$

User Browsing Model
previous clicked position $l$

$$P(C = 1|q, u, i, l) = P(C = 1|u, q, E = 1) \cdot P(E = 1|i, l)$$
Cascade Model

Model for each queries separately

$E_i, C_i$ be the probabilistic events indicating whether the $i$th url is examined and clicked resp.

$P(E_1) = 1$

$P(E_{i+1} = 1|E_i = 0) = 0$

$P(E_{i+1} = 1|E_i = 1, C_i) = 1 - C_i$

$P(C_i = 1|E_i = 1) = r_{u_i,q}$ where $u_i$ is the $i$th url

$\Rightarrow P(C_i = 1) = r_{u_i,q} \prod_{j=1}^{i-1} \left(1 - r_{u_j,q}\right)$
Cascade Model

\[ P(E_{i+1} = 1|E_i = 1, C_i) = 1 - C_i \]

Extension

Click Chain Model (CCM)

\[ P(E_{i+1} = 1|E_i = 1, C_i = 0) = \alpha_1 \]
\[ P(E_{i+1} = 1|E_i = 1, C_i = 1) = \alpha_2 (1 - r_{u_i,q}) + \alpha_3 r_{u_i,q} \]

Dynamic Bayesian Network (DBN)

\[ P(E_{i+1} = 1|E_i = 1, C_i = 0) = \gamma \]
\[ P(E_{i+1} = 1|E_i = 1, C_i = 1) = \gamma (1 - s_{u_i,q}) \]
Limitation

Click Chain Model (CCM)

\[ P(E_{i+1} = 1 | E_i = 1, C_i = 0) = \alpha_1 \]

\[ P(E_{i+1} = 1 | E_i = 1, C_i = 1) = \alpha_2 (1 - r_{u_i,q}) + \alpha_3 r_{u_i,q} \]

Dynamic Bayesian Network (DBN)

\[ P(E_{i+1} = 1 | E_i = 1, C_i = 0) = \gamma \]

\[ P(E_{i+1} = 1 | E_i = 1, C_i = 1) = \gamma (1 - s_{u_i,q}) \]

Transition probability only considers the relevance.
But a click is influenced by multiple bias:

- local hour
- user agent
- ...

The local hour

Click through rate

Linux user
non-Linux user with FireFox
non-Linux user with Opera
Other users, including IE users
How to tolerate multiple-bias in the click model?
General Click Model

We still need to keep E and C

They are good assumption

The Outer Model

- Bayesian network, in which we assume users scan urls from top to bottom

The Inner Model

- define the transition probability in the network to be a summation of parameters, each corresponding to a single attribute value

February 5, 2010
We need to consider multiple bias into transition probability

The Outer Model

• Bayesian network, in which we assume users scan urls from top to bottom

The Inner Model

• define the transition probability in the network to be a summation of parameters, each corresponding to a single attribute value
$$P(E_1) = 1$$
$$P(E_{i+1} = 1|E_i = 0) = 0$$
$$P(E_{i+1} = 1|E_i = 1, C_i = 0, B_i) = \mathbb{I}(B_i > 0)$$
$$P(E_{i+1} = 1|E_i = 1, C_i = 1, A_i) = \mathbb{I}(A_i > 0)$$
$$P(C_i = 1|E_i = 1, R_i) = \mathbb{I}(R_i > 0)$$
Similar Bayesian Network

GCM has a general notation of $A_i$, $B_i$ and $R_i$

Our main contribution comes next:

- The inner model – how to build $A_i$, $B_i$ and $R_i$
We assume each attribute value $f$ is associated with three parameters $\theta^A_f, \theta^B_f$ and $\theta^R_f$, each of which is a continuous random variable.

$$A_i = \sum_{j=1}^{s} \theta^A_{f_j} \text{user} + \sum_{j=1}^{t} \theta^A_{f_i,j} \text{url} + err$$

$$B_i = \sum_{j=1}^{s} \theta^B_{f_j} \text{user} + \sum_{j=1}^{t} \theta^B_{f_i,j} \text{url} + err$$

$$R_i = \sum_{j=1}^{s} \theta^R_{f_j} \text{user} + \sum_{j=1}^{t} \theta^R_{f_i,j} \text{url} + err$$

Let $\Theta = \{\theta^A_f, \theta^B_f, \theta^R_f \mid \forall f\}$ be the parameter set.
GCM – The Inner Model

- the query
- the location
- the browser type
- the local hour
- the IP address
- the query length
  \[ f_{1user}, f_{2user}, \ldots f_{suser} \]

- the url
- the displayed position (=i)
- the classification of the url
- the matched keyword
- the length of the url
  \[ f_{i1url}, f_{i2url}, \ldots f_{iturl} \]
Assume parameters in $\Theta$ are independent Gaussians.

Bayesian Inference

*Expectation Propagation* method by Tom Minka

Given the structure of a Bayesian network with hidden variables, EP takes the observation values as input, and is capable of calculating the inference of any variable.

For each training session, we use the current Gaussians as prior, do the EP, and then calculate the posterior Gaussians and update them in $\Theta$. 

Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta^A_f, \theta^B_f, \theta^R_f | \forall f\}$ and let each parameter in $\Theta$ satisfy a prior $N(0, 1/(s + t))$.


3. For each session $s$:
   
4. $M \leftarrow$ number of urls in $s$:
   
5. Obtain the attribute values $F = \{f^\text{user}_1, ..., f^\text{user}_s\} \cup \{f^\text{url}_{i,1}, ..., f^\text{url}_{i,t}\}_{i=1}^{M}$
   
6. Input $\{\theta^A_f, \theta^B_f, \theta^R_f | f \in F\} \subset \Theta$ to $G$ as the prior Gaussian distributions.

7. Input the user’s clicks to $G$ as observations.

8. Execute the $G$, measure the posterior distributions for $\{\theta^A_f, \theta^B_f, \theta^R_f | f \in F\}$, and update them in $\Theta$.

9. **End For**
Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f \}$ and let each parameter in $\Theta$ satisfy a prior $N(0, 1/(s + t))$.


3. For each session $s$

4. $M \leftarrow$ number of urls in $s$

5. Obtain the attribute values

   $F = \{f_1^{user}, \ldots, f_s^{user}\} \cup \{f_i^{url}, \ldots, f_{i,t}^{url}\}^M_{i=1}$

6. Input $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\} \subset \Theta$ to $G$ as the prior Gaussian distributions.

7. Input the user’s clicks to $G$ as observations.

8. Execute the $G$, measure the posterior distributions for $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\}$, and update them in $\Theta$

9. End For
Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in $\Theta$ satisfy a prior $N(0, 1/(s + t))$.
3. For each session $s$
4. $M \leftarrow$ number of urls in $s$
5. Obtain the attribute values
   $$F = \{f_1^{user}, ..., f_s^{user}\} \cup \{f_i^{url}, ..., f_i^{url}\}_{i=1}^M$$
6. Input $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\} \subset \Theta$ to $G$ as the prior Gaussian distributions.
7. Input the user’s clicks to $G$ as observations.
8. Execute the $G$, measure the posterior distributions for $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\}$, and update them in $\Theta$.
9. End For
Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in $\Theta$ satisfy a prior $N(0, 1/(s + t))$.
3. For each session $s$
4. $M \leftarrow$ number of urls in $s$
5. Obtain the attribute values
   $$F = \{f_1^{user}, \ldots, f_s^{user}\} \cup \{f_1^{url}, \ldots, f_M^{url}\}^{M}_{i=1}$$
6. Input $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\} \subset \Theta$ to $G$ as the prior Gaussian distributions.
7. Input the user’s clicks to $G$ as observations.
8. Execute the $G$, measure the posterior distributions for $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\}$, and update them in $\Theta$
9. End For
Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in $\Theta$ satisfy a prior $N(0, 1/(s + t))$.
3. For each session $s$:
   4. $M \leftarrow$ number of urls in $s$
   5. Obtain the attribute values $F = \{f_{i, \text{user}}^1, \ldots, f_{s, \text{user}}^s\} \cup \{f_{i, \text{url}}^1, \ldots, f_{i, \text{url}}^M\}_{i=1}^M$
6. Input $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\} \subset \Theta$ to $G$ as the prior Gaussian distributions.
7. Input the user’s clicks to $G$ as observations.
8. Execute the $G$, measure the posterior distributions for $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\}$, and update them in $\Theta$.
9. End For
Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in $\Theta$ satisfy a prior $N(0, 1/(s + t))$.
2. Construct a Bayesian inference calculator $G$ using *Expectation Propagation*.
3. For each session $s$
4. \[ M \leftarrow \text{number of urls in } s \]
5. Obtain the attribute values
\[ F = \{f_1^{\text{user}}, ..., f_s^{\text{user}}\} \cup \{f_{i,1}^{\text{url}}, ..., f_{i,t}^{\text{url}}\}_{i=1}^M \]
6. Input $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\} \subset \Theta$ to $G$ as the prior Gaussian distributions.
7. Input the user’s clicks to $G$ as observations.
8. Execute the $G$, measure the posterior distributions for $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\}$, and update them in $\Theta$.
9. End For.
Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in $\Theta$ satisfy a prior $N(0, 1/(s + t))$.
3. For each session $s$
   4. $M \leftarrow$ number of urls in $s$
   5. Obtain the attribute values
      $$F = \{f_1^{user}, ..., f_s^{user}\} \cup \{f_i^{url}, ..., f_i^{url}\}^{M}_{i=1}$$
   6. Input $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\} \subset \Theta$ to $G$ as the prior Gaussian distributions.
   7. Input the user’s clicks to $G$ as observations.
   8. Execute the $G$, measure the posterior distributions for $\{\theta_f^A, \theta_f^B, \theta_f^R | f \in F\}$, and update them in $\Theta$.
9. End For
Lemma: If we define an attribute value \( f \) to be the pair of query and url \( f = (u_i, q) \), the traditional transition probability

\[
P(C_i = 1|E_i = 1) = r_{u_i,q}
\]

can reduce to

\[
P(C_i = 1|E_i = 1, R_i) = \mathbb{I}(R_i > 0)
\]

if we set \( R_i = \theta_f^R + err \) and \( \theta_f^R \) is a point mass Gaussian centered at \( F^{-1}(r_{u_i,q}) \), where \( F \) is the cumulative distribution function of \( N(0,1) \).

Recall \( R_i = \sum_{j=1}^{S} \theta_{f_j}^R_{\text{user}} + \sum_{j=1}^{t} \theta_{f_{i,j}}^R_{\text{url}} + err \)
Examination Hypothesis

\[ P(B_i > 0) = P(A_i > 0) = x_{i+1} \]
\[ P(R_i > 0) = r_{u_i,q} \]

define two attributes \( f_1 = i + 1 \) and \( f_2 = (u_i, q) \)

\[ A_i = \theta_{f_1}^A + \text{err} \]
\[ B_i = \theta_{f_1}^B + \text{err} \]
\[ R_i = \theta_{f_2}^R + \text{err} \]

Similar for other prior works
Experiment
Main Contribution

**Multi-bias aware.** The transition probabilities between variables depend jointly on a list of attributes. This enables our model to explain bias terms other than the position-bias.

**Learning across queries.** The model learns queries altogether and thus can predict clicks for one query – even a new query – using the learned data from other queries.

**Extensible:** The user may actively add or remove attributes applied in our GCM model. In fact, all the prior works mentioned above can reduce to our GCM as special cases when only one or two attributes are incorporated.

**One-pass.** Our click model is an on-line algorithm. The posterior distributions will be regarded as the prior knowledge for the next query session.

**Applicable to ads.** We have demonstrated our click model in the CTR prediction of advertisements. Experimental results show that our click model outperforms the prior works.
Future work

- To learn CTR@1
- Continuous attribute values
- Make use of the page structure
- Running time
Thanks!

Questions:  
zhuzeyuan@hotmail.com  
wzchen@microsoft.com

Thanks to:  
Haixun Wang  
Gang Wang  
Dakan Wang
Introduction

Implicit feedback

Attributes
- Query text
- Timestamps
- Localities
- The click-or-not flag
- Etc...
<table>
<thead>
<tr>
<th><strong>Definitions</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query</strong></td>
</tr>
<tr>
<td><strong>Query session</strong></td>
</tr>
<tr>
<td><strong>Urls impressions</strong></td>
</tr>
<tr>
<td><strong>Attribute</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Set</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>All</td>
</tr>
</tbody>
</table>
Experiment

![Graph showing log-likelihood and perplexity for different sets and models.]

**Log-Likelihood**
- CCM
- DBN
- GCM

**Perplexity**
- CCM
- DBN
- GCM

Set 1, Set 2, Set 3, Set 4, Set 5, Set 6, Set 7, Set 8, All
Related Works

CCM

![Flowchart diagram showing decision-making process with mathematical expression for CCM calculation.]

\[ \alpha_2 (1 - R_i) + \alpha_3 R_i \]
Related Works

CCM

\[ p(R_i|C_{1:U}) \approx \text{(constant)} \times p(R_i) \prod_{u=1}^{U} P(C^u|R_i). \]

<table>
<thead>
<tr>
<th>Case</th>
<th>Conditions</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( i &lt; l, C_i = 0 )</td>
<td>( 1 - R_i )</td>
</tr>
<tr>
<td>2</td>
<td>( i &lt; l, C_i = 1 )</td>
<td>( R_i \left(1 - \frac{1 - \alpha_3}{\alpha_2} R_i\right))</td>
</tr>
<tr>
<td>3</td>
<td>( i = l )</td>
<td>( R_i \left(1 + \frac{\alpha_2}{2 - \alpha_1^2} R_i\right))</td>
</tr>
<tr>
<td>4</td>
<td>( i &gt; l )</td>
<td>( 1 - \frac{2}{1 + (2/\alpha_1)^{i-1}} R_i )</td>
</tr>
<tr>
<td>5</td>
<td>No Click</td>
<td>( 1 - \frac{2}{1 + (2/\alpha_1)^{i-1}} R_i )</td>
</tr>
</tbody>
</table>

**Figure 4:** Different cases for computing \( P(C|R_i) \) up to a constant where \( l \) is the last clicked position. Darker nodes in the figure above indicate clicks.
Related Works

CCM

![Graph showing Log-Likelihood vs Query Frequency for UBM, DCM, and CCM]
DBN

\[ P(E_{i+1} = 1|E_i = 1, C_i = 0) = \gamma \]
\[ P(E_{i+1} = 1|E_i = 1, C_i = 1) = \gamma (1 - s_{u,i,q}) \]

\[ r_u := P(S_i = 1|E_i = 1) \]
\[ = P(S_i = 1|C_i = 1)P(C_i = 1|E_i = 1) \]
\[ = a_u s_u \]
Related Works

DBN

\[ a_u = \arg \max_a \sum_{j=1}^{N} \sum_{i=1}^{10} I(d^j_i = u) \]

\[ s_u = \arg \max_g \sum_{j=1}^{N} \sum_{i=1}^{10} I(d^j_i = u, C^j_i = 1) \]

\[ Q(A^j_i = 0) \log(1 - a) + Q(A^j_i = 1) \log(a) \right) + \log P(a). \]

\[ Q(S^j_i = 0) \log(1 - s) + Q(S^j_i = 1) \log(s) \right) + \log P(s). \]

\[ Q(A^j_i) := P(A^j_i | C^j, a_u, s_u, \gamma) \]

\[ Q(S^j_i) := P(S^j_i | C^j, a_u, s_u, \gamma). \]
Related Works

DBN