Microsoft® Research

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
  - …

Related Works

Microsoft® Research

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
    - $P(C=1|q,u,i) = \underbrace{P(C=1|u,q,E=1)}_{r_{u,q}} \cdot \underbrace{P(E=1|i)}_{x_i}$
  - User Browsing Model
    - previous clicked position l
    - $P(C=1|q,u,i,l) = \underbrace{P(C=1|u,q,E=1)}_{r_{u,q}} \cdot \underbrace{P(E=1|i,l)}_{x_{i,l}}$

Related Works

Microsoft® Research

- Cascade Model
  - Model for each queries separately
  - E_i, C_i be the probabilistic events indicating whether the ith 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 ith url
- $\Rightarrow P(C_i = 1) = r_{u_i,q} \prod_{j=1}^{i-1} \left(1 - r_{u_j,q}\right)$

Related Works

Microsoft® Research

- 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

Microsoft® Research

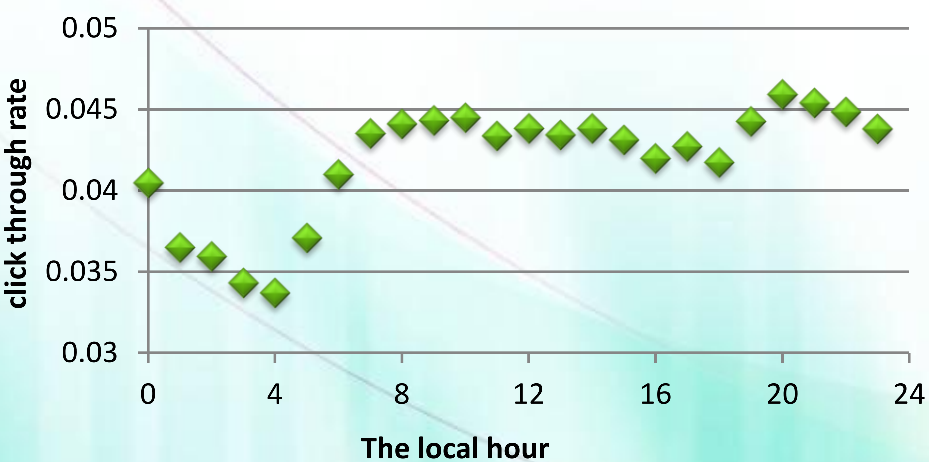
Transition probability only considers the relevance.

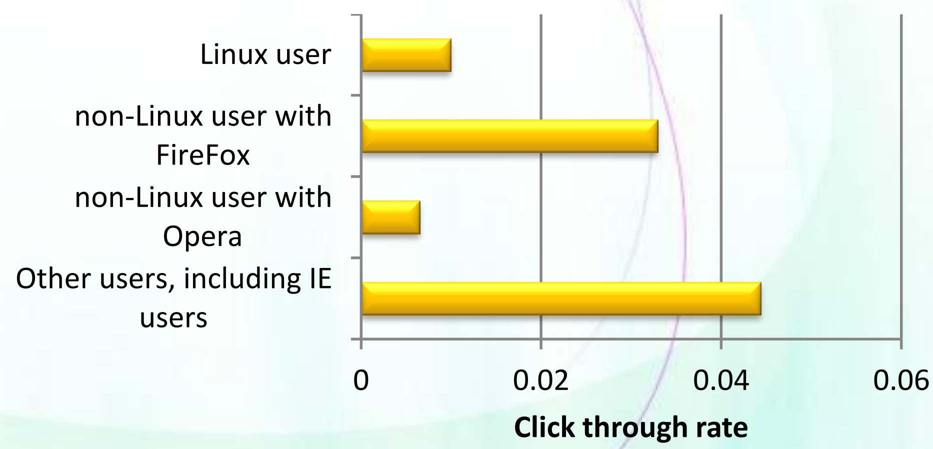
- Click Chain Model (CCM)
  - $P(E_{i+1} = 1 | E_i = 1, C_i = 0) = \alpha_1$
  - $P(E_{i+1} = 1 | E_i = 1, C_i = 1) = \alpha_2 (1 - r_{u_i,q}) + \alpha_3 r_{u_i,q}$
- Dynamic Bayesian Network (DBN)
  - $P(E_{i+1} = 1 | E_i = 1, C_i = 0) = \gamma$
  - $P(E_{i+1} = 1 | E_i = 1, C_i = 1) = \gamma (1 - s_{u_i,q})$

Observation

Microsoft® Research

- But a click is influenced by multiple bias:
  - local hour
  - user agent
  - …





Big Challenge

Microsoft® Research

- 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

General Click Model

Microsoft® Research

- We need to consider multiple bias into transition probability

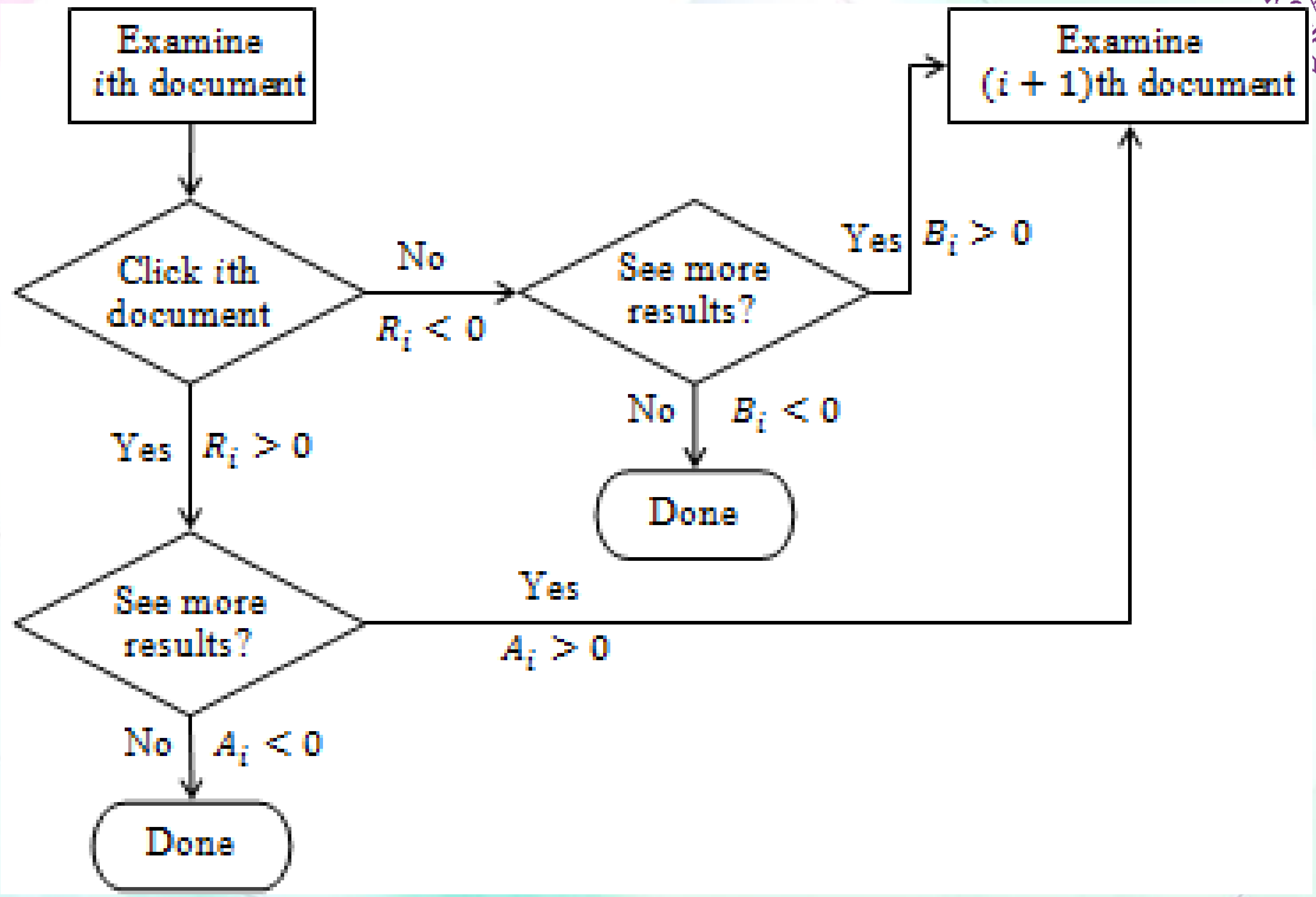
The Outer Model

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

The Inner Model

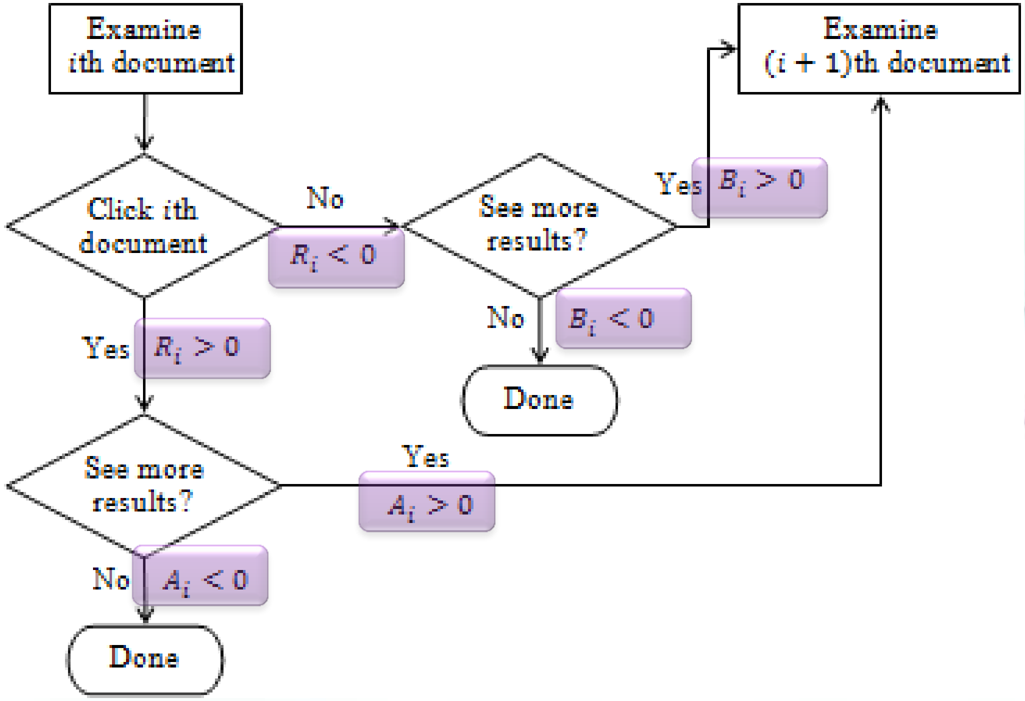
- define the transition probability in the network to be a summation of parameters, each corresponding to a single attribute value

GCM – The Outer Model



GCM – The Outer Model

Microsoft® Research



- $P(E_1) = 1$
- $P(E_{i+1} = 1 | E_i = 0) = 0$
- $P(E_{i+1} = 1 | E_i = 1, C_i = 0, B_i) = \mathbb{I}(B_i > 0)$
- $P(E_{i+1} = 1 | E_i = 1, C_i = 1, A_i) = \mathbb{I}(A_i > 0)$
- $P(C_i = 1 | E_i = 1, R_i) = \mathbb{I}(R_i > 0)$

Different with DBN/CCM

Microsoft® Research

- 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

GCM – The Inner Model

Microsoft® Research

- We assume each attribute value f is associated with three parameters θ_f^A, θ_f^B and θ_f^R, each of which is a continuous random variable
- $A_i = \sum_{j=1}^{s} \theta_{f_j^{user}}^A + \sum_{j=1}^{t} \theta_{f_{i,j}^{url}}^A + err$
- $B_i = \sum_{j=1}^{s} \theta_{f_j^{user}}^B + \sum_{j=1}^{t} \theta_{f_{i,j}^{url}}^B + err$
- $R_i = \sum_{j=1}^{s} \theta_{f_j^{user}}^R + \sum_{j=1}^{t} \theta_{f_{i,j}^{url}}^R + err$
- Let $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ be the parameter set.

GCM – The Inner Model

Microsoft® Research

user-specific attributes

- the query
- the location
- the browser type
- the local hour
- the IP address
- the query length

$$f_1^{user}, f_2^{user}, \dots f_s^{user}$$

- the url
- the displayed position(=i)
- the classification of the url
- the matched keyword
- the length of the url

$$f_{i,1}^{url}, f_{i,2}^{url}, \dots f_{i,t}^{url}$$

url-specific attributes

GCM – The Inference Method

Microsoft® Research

- Assume parameters in Θ 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 Θ.

GCM – Algorithm

Microsoft® Research

Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in Θ satisfy a prior $N(0, 1/(s+t))$.
2. Construct a Bayesian inference calculator G using *Expectation Propagation*.
3. For each session s
4. $M \leftarrow$ number of urls in s
5. Obtain the attribute values
   $F = \{f_1^{user}, \dots f_s^{user}\} \cup \{f_{i,1}^{url}, \dots f_{i,t}^{url}\}_{i=1}^{M}$
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 Θ
9. End For

GCM – Algorithm

Microsoft® Research

Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R \mid \forall f\}$ and let each parameter in Θ satisfy a prior $N(0, 1/(s+t))$.
2. Construct a Bayesian inference calculator G using *Expectation Propagation*.
3. For each session s
4. $M \leftarrow$ number of urls in s
5. Obtain the attribute values $F = \{f_1^{user}, \dots f_s^{user}\} \cup \{f_{i,1}^{url}, \dots f_{i,t}^{url}\}_{i=1}^{M}$
6. Input $\{\theta_f^A, \theta_f^B, \theta_f^R \mid f \in F\} \subset \Theta$ 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 $\{\theta_f^A, \theta_f^B, \theta_f^R \mid f \in F\}$, and update them in Θ
9. End For

GCM – Algorithm

Microsoft® Research

Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in Θ satisfy a prior $N(0, 1/(s+t))$.
2. Construct a Bayesian inference calculator G using *Expectation Propagation*.
3. For each session s
4. $M \leftarrow$ number of urls in s
5. Obtain the attribute values $F = \{f_1^{user}, \dots f_s^{user}\} \cup \{f_{i,1}^{url}, \dots f_{i,t}^{url}\}_{i=1}^{M}$
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 Θ
9. End For

GCM – Algorithm

Microsoft® Research

Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in Θ satisfy a prior $N(0, 1/(s+t))$.
2. Construct a Bayesian inference calculator G using *Expectation Propagation*.
3. For each session s
4. $M \leftarrow$ number of urls in s
5. Obtain the attribute values $F = \{f_1^{user}, \ldots f_s^{user}\} \cup \{f_{i,1}^{url}, \ldots f_{i,t}^{url}\}_{i=1}^{M}$
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 Θ
9. End For

GCM – Algorithm

Microsoft® Research

Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in Θ satisfy a prior $N(0, 1/(s+t))$.
2. Construct a Bayesian inference calculator G using *Expectation Propagation*.
3. For each session s
4. $M \leftarrow$ number of urls in s
5. Obtain the attribute values $F = \{f_1^{user}, \dots f_s^{user}\} \cup \{f_{i,1}^{url}, \dots f_{i,t}^{url}\}_{i=1}^{M}$
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 Θ
9. End For

GCM – Algorithm

Microsoft® Research

Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in Θ satisfy a prior $N(0, 1/(s+t))$.
2. Construct a Bayesian inference calculator G using *Expectation Propagation*.
3. For each session s
4. $M \leftarrow$ number of urls in s
5. Obtain the attribute values
   $F = \{f_1^{user}, \dots f_s^{user}\} \cup \{f_{i,1}^{url}, \dots f_{i,t}^{url}\}_{i=1}^{M}$
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 Θ
9. End For

GCM – Algorithm

Microsoft® Research

Algorithm: The General Click Model

1. Initiate $\Theta = \{\theta_f^A, \theta_f^B, \theta_f^R | \forall f\}$ and let each parameter in Θ satisfy a prior $N(0, 1/(s+t))$.
2. Construct a Bayesian inference calculator G using *Expectation Propagation*.
3. For each session s
4. $M \leftarrow$ number of urls in s
5. Obtain the attribute values
   $F = \{f_1^{user}, \dots f_s^{user}\} \cup \{f_{i,1}^{url}, \dots f_{i,t}^{url}\}_{i=1}^{M}$
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 Θ
9. End For

GCM – Reductions

- **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 θ_f^R is a point mass Gaussian centered at $F^{-1}\left(r_{u_i,q}\right)$,where F is the cumulative distribution function of $N(0,1)$.
- Recall $R_i = \sum_{j=1}^{s} \theta_{f_j^{user}}^R + \sum_{j=1}^{t} \theta_{f_{i,j}^{url}}^R + err$

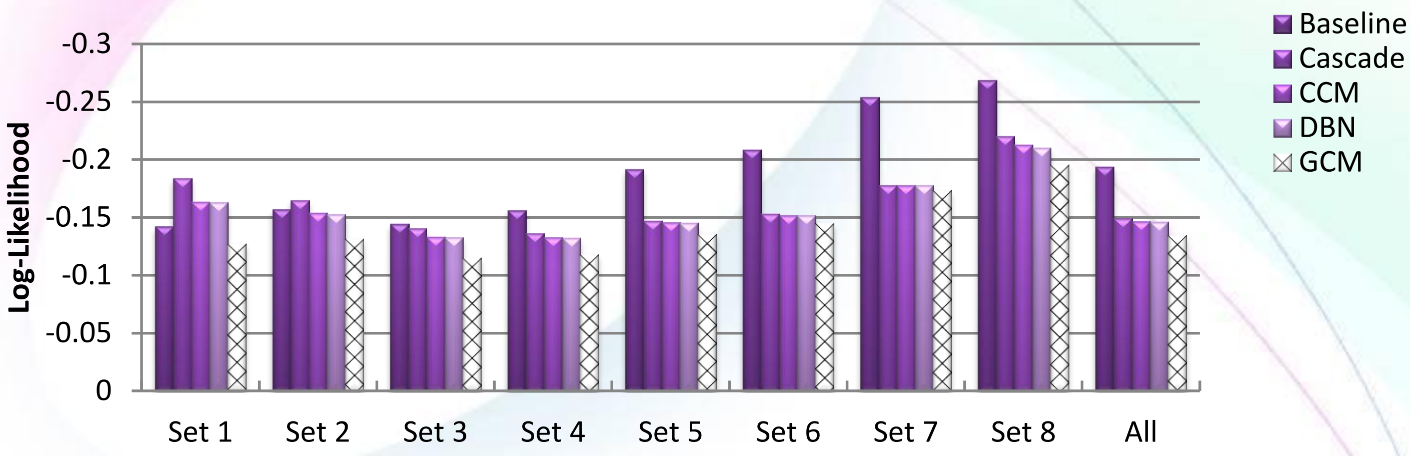
GCM – Reductions

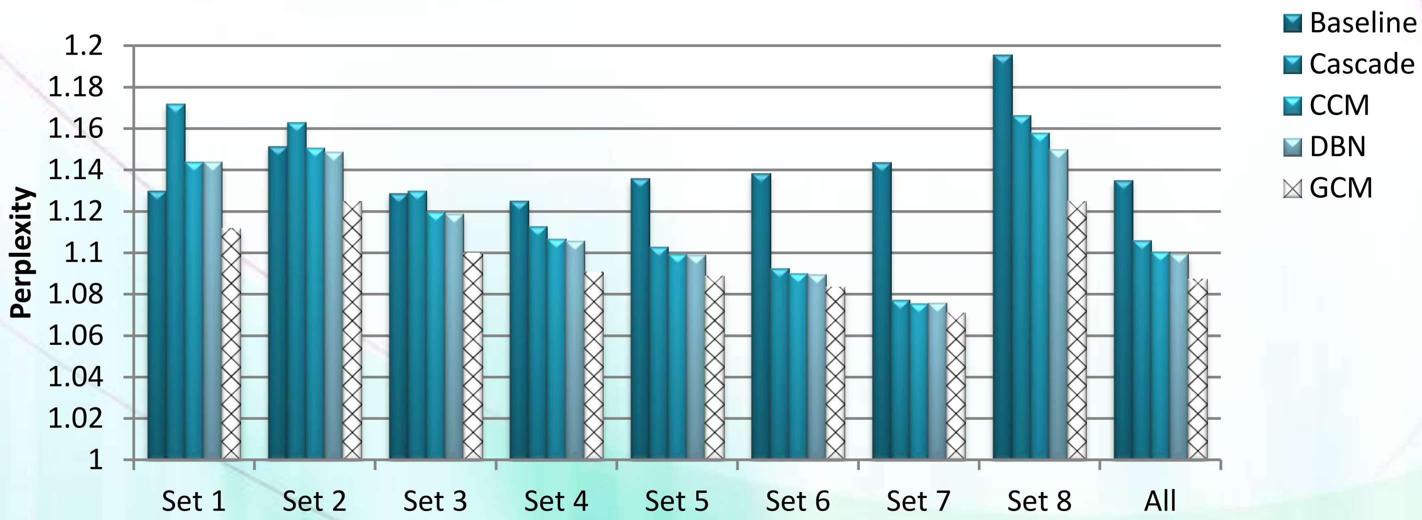
- *Examination Hypothesis*

- $P(B_i > 0) = P(A_\mathrm{i} > 0) = x_{i+1}$
- $P(R_i > 0) = r_{u_i,q}$
- define two attributes $f_1 = i + 1$ and $f_2 = (u_i, q)$
- $A_i = \theta^A_{f_1} + err;\ B_i = \theta^B_{f_1} + err;\ R_i = \theta^R_{f_2} + err$

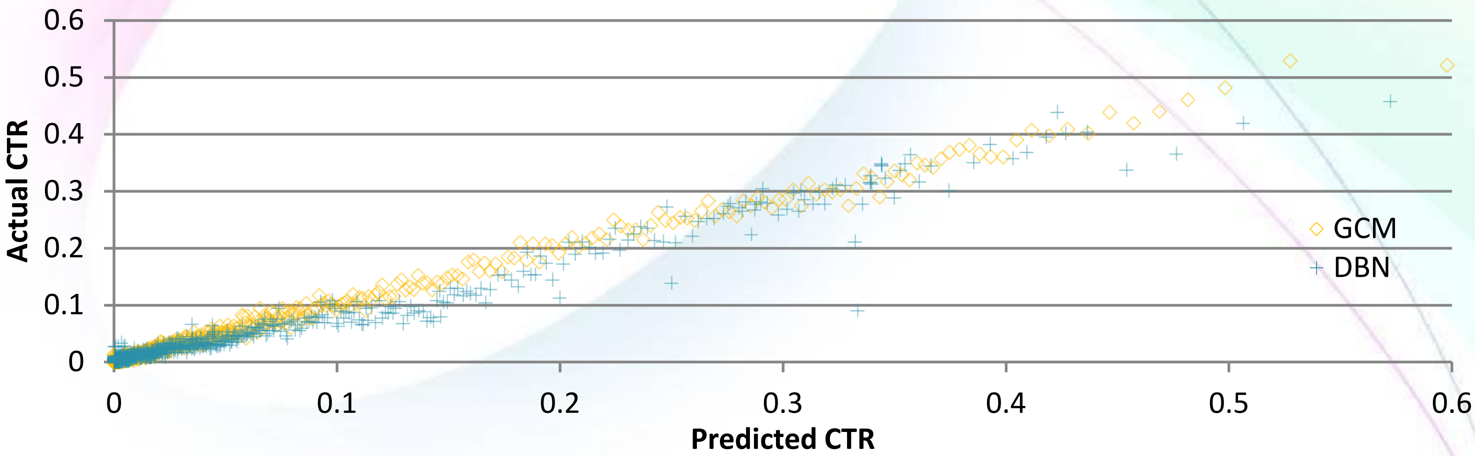
- Similar for other prior works

Experiment

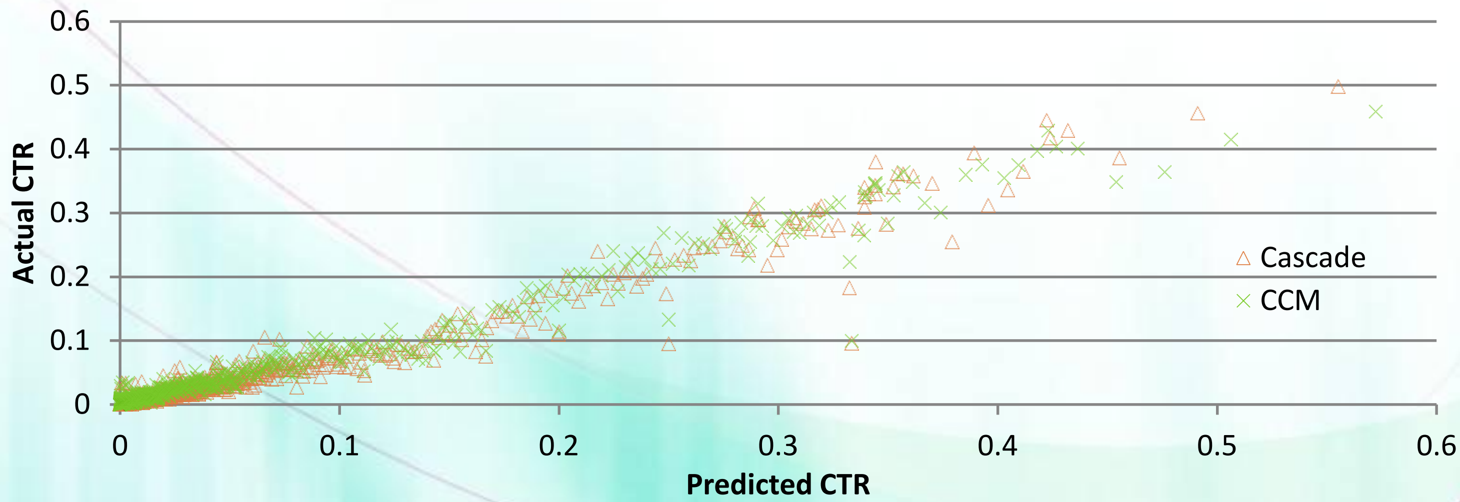

Log-Likelihood
-0.3
-0.25
-0.2
-0.15
-0.1
-0.05
0
Set 1
Set 2
Set 3
Set 4
Set 5
Set 6
Set 7
Set 8
All
Baseline
Cascade
CCM
DBN
GCM


Perplexity
1.2
1.18
1.16
1.14
1.12
1.1
1.08
1.06
1.04
1.02
1
Set 1
Set 2
Set 3
Set 4
Set 5
Set 6
Set 7
Set 8
All
Baseline
Cascade
CCM
DBN
GCM

Experiment



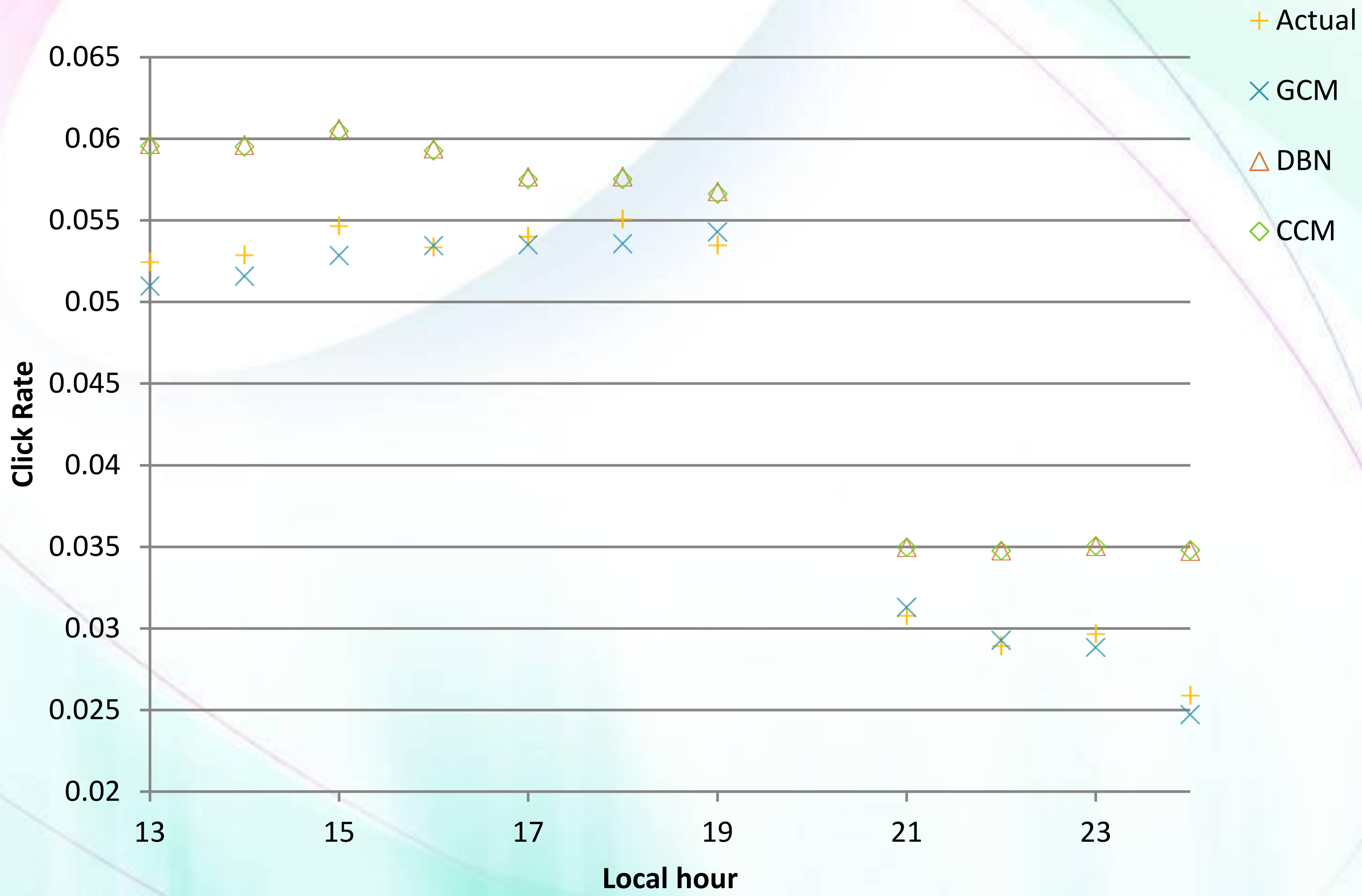
0.6
0.5
0.4
0.3
0.2
0.1
0
Actual CTR
0
0.1
0.2
0.3
0.4
0.5
0.6
Predicted CTR
GCM
DBN



0.6
0.5
0.4
0.3
0.2
0.1
0
Actual CTR
0
0.1
0.2
0.3
0.4
0.5
0.6
Predicted CTR
Cascade
CCM

Experiment

Microsoft® Research



Main Contribution

Microsoft® Research

- **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…

Definitions

Microsoft® Research

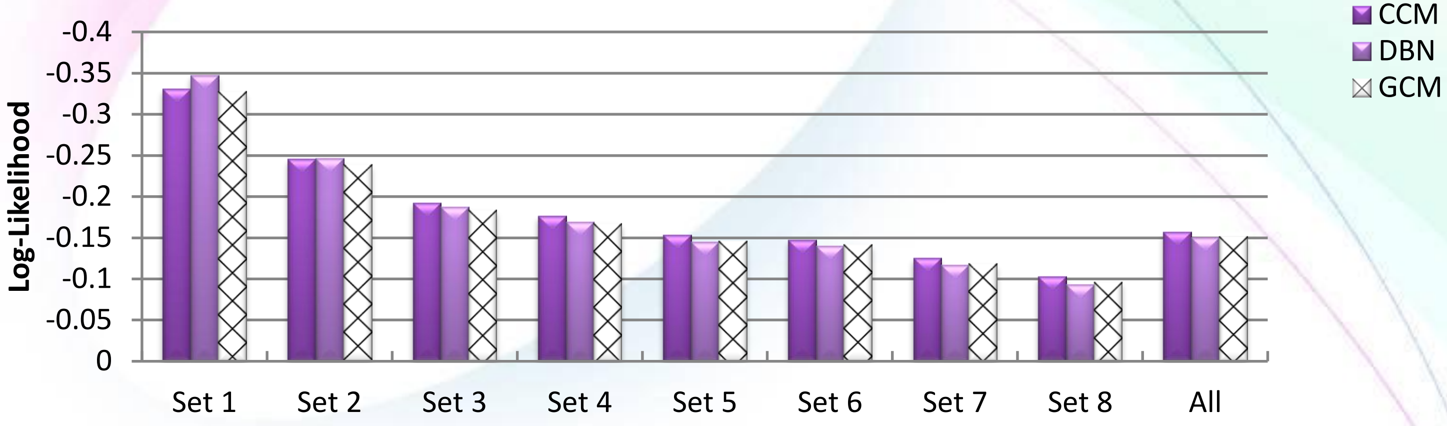
Query	"Microsoft Research"
Query session	$U = \{u_1, u_2, \ldots u_M\}$
Urls impressions	u_2 ="research.microsoft.com"
Attribute	192.168.0.1
	IE
	7am local time

Experiment

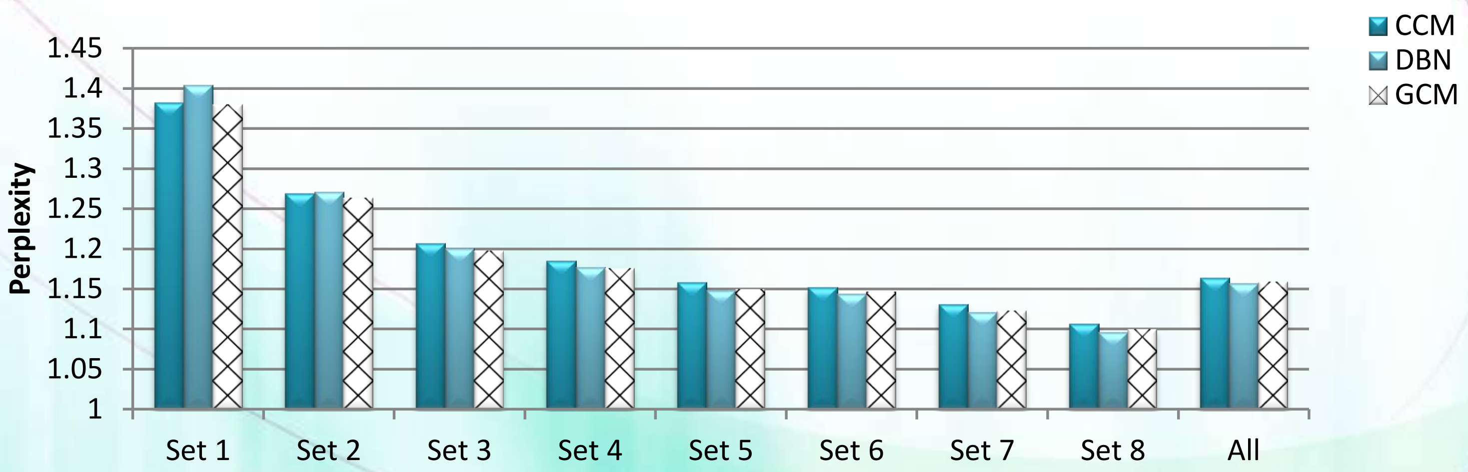
Microsoft® Research

Set	Query Freq	#Queries	Train set		Test set	
			#Sessions	#Urls	#Sessions	#Urls
1	1~10	141	866	5,698	177	1,057
2	10~30	1,211	24,928	1,664,403	2,122	13,664
3	30~100	5,058	308,203	1,810,009	18,629	105,716
4	100~300	3,988	674,654	3,148,826	40,304	180,532
5	300~1000	1,651	847,722	3,011,482	54,098	184,606
6	1,000~3,000	481	792,422	2,470,665	48,449	147,561
7	3,000~10,000	132	660,645	1,508,985	42,067	92,122
8	10,000~30,000	22	315,832	769,786	19,338	48,808
9	30,000+	7	642,835	1,046,948	37,796	64,236
All	All of above	12,691	4,267,241	15,431,104	262,803	837,245

Experiment



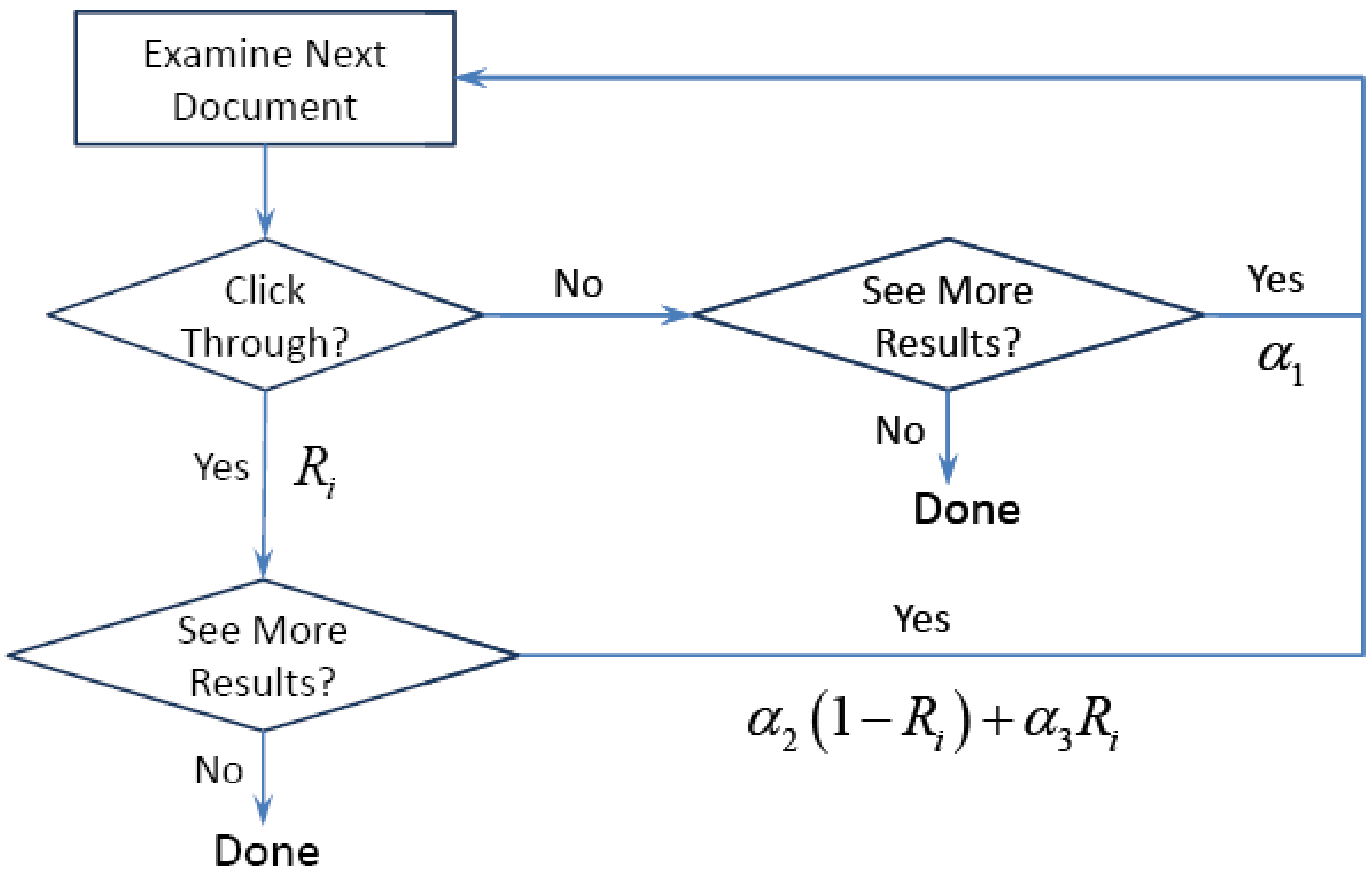
-0.4
-0.35
-0.3
-0.25
-0.2
-0.15
-0.1
-0.05
0
Log-Likelihood
Set 1
Set 2
Set 3
Set 4
Set 5
Set 6
Set 7
Set 8
All
CCM
DBN
GCM



1.45
1.4
1.35
1.3
1.25
1.2
1.15
1.1
1.05
1
Perplexity
Set 1
Set 2
Set 3
Set 4
Set 5
Set 6
Set 7
Set 8
All
CCM
DBN
GCM

Related Works

- CCM



Related Works

Microsoft® Research

- CCM

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

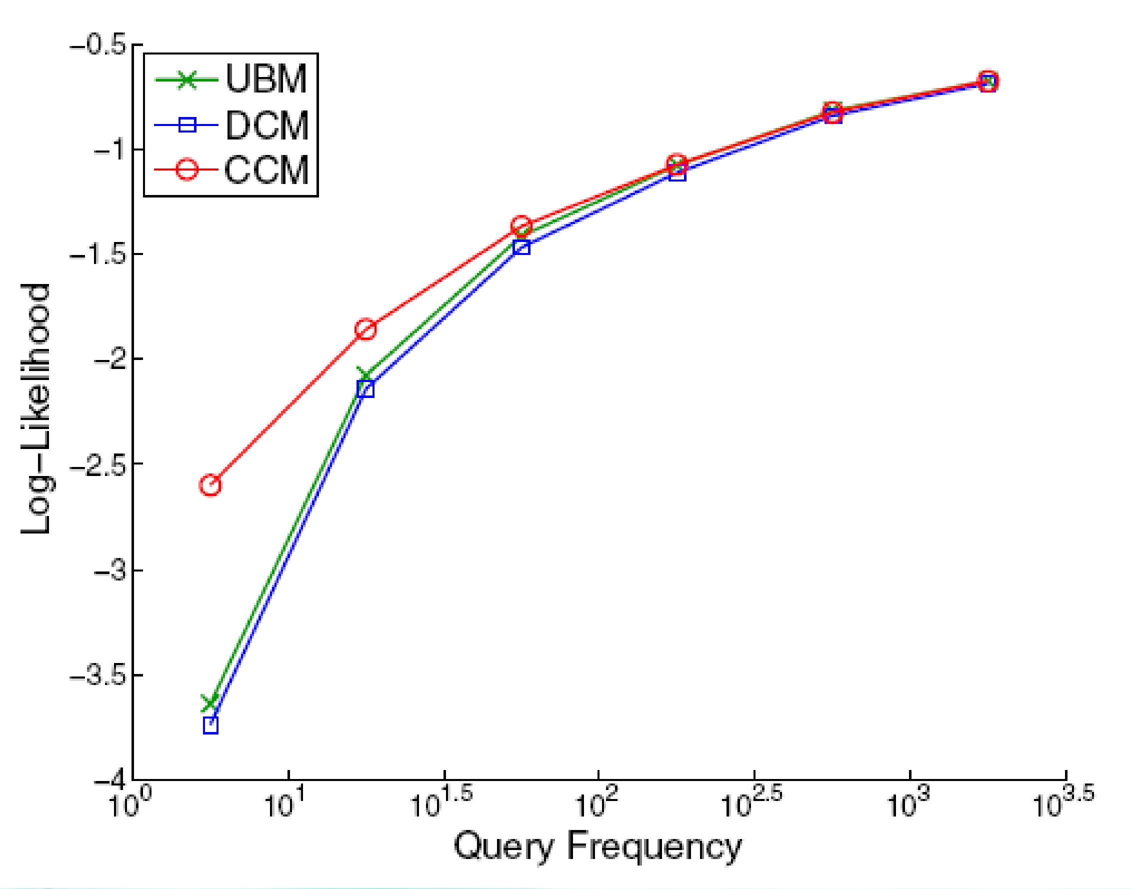
Case	Conditions	Results
1	$i < l, C_i = 0$	$1 - R_i$
2	$i < l, C_i = 1$	$R_i(1 - (1 - \alpha_3/\alpha_2)R_i)$
3	$i = l$	$R_i\left(1 + \frac{\alpha_2 - \alpha_3}{2 - \alpha_1 - \alpha_2} R_i\right)$
4	$i > l$	$1 - \frac{2}{1 + \frac{6 - 3\alpha_1 - \alpha_2 - 2\alpha_3}{(1-\alpha_1)(\alpha_2 + 2\alpha_3)} (2/\alpha_1)^{(i-l)-1}} R_i$
5	No Click	$1 - \frac{2}{1 + (2/\alpha_1)^{i-1}} R_i$

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

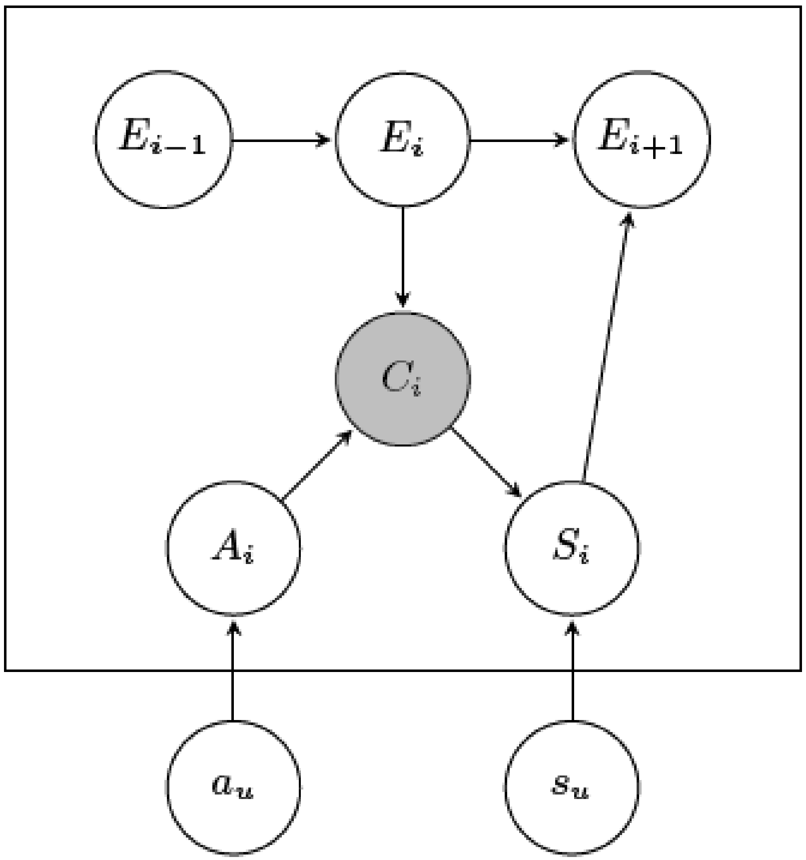
Microsoft® Research

- CCM



Related Works

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



$$
\begin{aligned}
r_u :=& P(S_i = 1 | E_i = 1) \\
=& P(S_i = 1 | C_i = 1) P(C_i = 1 | E_i = 1) \\
=& a_u s_u
\end{aligned}
$$

Related Works

- DBN

$$a_u = \arg\max_a \sum_{j=1} \sum_{i=1} I(d_i^j = u)$$

$$\Big(Q(A_i^j = 0)\log(1-a) + Q(A_i^j = 1)\log(a)\Big) + \log P(a).$$

$$s_u = \arg\max_s \sum_{j=1}^{N} \sum_{i=1}^{10} I(d_i^j = u, C_i^j = 1)$$

$$\Big(Q(S_i^j = 0)\log(1-s) + Q(S_i^j = 1)\log(s)\Big) + \log P(s).$$

$$Q(A_i^j) := P(A_i^j | C^j, a_u, s_u, \gamma)$$

$$Q(S_i^j) := P(S_i^j | C^j, a_u, s_u, \gamma).$$

Related Works

- DBN