Evaluating Search Engines by Modeling the Relationship Between Relevance and Clicks

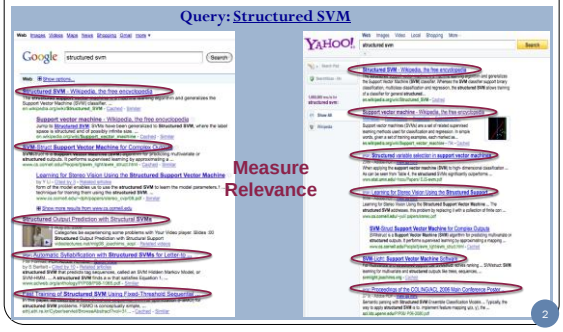
Ben Carterette & Rosie Jone

Presented by Congcong Li
04 / 01 / 2010

1

Introduction

- Goal: Evaluate different search engines



2

Introduction

- How to evaluate search engines?
  - Use relevance judgments
  - Complete relevance judgments not available
- How to predict relevance?
  - Use clicks
  - General problem: Clicks are biased!
- This work:

  Model the relationship between relevance & clicks

3

Evaluating Search Engine

- Discount Cumulative Gain

$$DCG_l = \sum_{i=1}^{l} gain_i \cdot discount_i = rel_1 + \sum_{i=2}^{l} rel_i \cdot \frac{1}{\log_2 i}$$

$rel_i \in \{a1, a2, a3, a4, a5\}$

{Perfect, Excellent, Good, Fair, Bad}

4

Evaluating DCG from Incomplete Information

- Express DCG as a random variable:

$$DCG_l = X_1 + \sum_{i=2}^{l} X_i \cdot \frac{1}{\log_2 i} \qquad X_i \in \{a_1, a_2, a_3, a_4, a_5\}$$

- Expectation and variance (to estimate confidence interval):

$$E[DCG_l] = E[X_1] + \sum_{i=2}^{l} \frac{E[X_i]}{\log_2 i}$$

$$Var[DCG_l] = Var[X_1] + \sum_{i=2}^{l} \frac{E[X_i]}{(\log_2 i)^2} + \underbrace{2\sum_{i=2}^{l} \frac{Cov(X_1, X_i)}{\log_2 i} + 2\sum_{k<i<j} \frac{Cov(X_i, X_j)}{\log_2 i \cdot \log_2 j}}_{= 0 \text{ under independent assumption}}$$

How to get $p(X_i = a_j)$? **Predict from clicks!**

5

Modeling Clicks & Relevance

- We want to predict $p(X_i = a_j)$

Joint probability $p(q, \mathbf{X}, \mathbf{c})$

$p(q, X_1, X_2, \ldots, X_l, c_1, c_2, \ldots, c_l)$

query relevance clickthrough_rates

Conditional probability $p(\mathbf{X} \mid q, \mathbf{c})$

Independent assumption

$$p(\mathbf{X} \mid q, \mathbf{c}) = \prod_{i=1}^{l} p(X_i \mid q, \mathbf{c})$$

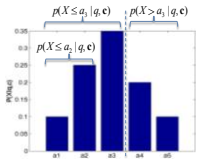
6

Modeling Clicks & Relevance

- Model $p(\mathrm{X} \mid q, \mathbf{c}) = \prod_{i=1}^{l} p(X_i \mid q, \mathbf{c})$
- Ordinal Regression

$$\log \frac{p(X > a_j \mid q, \mathbf{c})}{p(X \le a_j \mid q, \mathbf{c})} = a_j + \beta q + \sum_{i=1}^{l} \beta_i c_i + \sum_{i<k}^{l} \beta_{ik} c_i c_k$$

Scalars



a_j : one of the relevance levels
q : the aggregate click rate over all results
c_i : click times over # of times the list was shown

- Vector Generalized additive model (VGAM)

7

Comparative Evaluation

- If we only care about whether one ranking function outperforms another?

  we care only about the **sign** of ΔDCG_l

$$\Delta DCG_l = DCG_{l1} - DCG_{l2} = \sum_{i=1}^{N} g_{i1} - g_{i2}$$

$$g_{ij} = \begin{cases} rel_i & r_j(i) = 1 \\ \dfrac{rel_i}{\log_2 r_j(i)} & 1 < r_j(i) \le l \\ 0 & r_j(i) > l \end{cases}$$

$r_j(i)$ is the rank of document i from system j

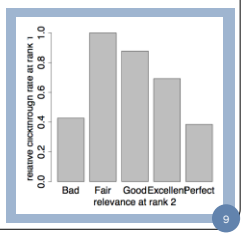
- Compute $P(\Delta DCG_l < 0)$
  - Monte Carlo simulation
  - Draw samples according to $p(X_1, X_2, \ldots X_l \mid q, \mathbf{c})$

8

Experiments

- Data
  - From Yahoo!
  - 28,961 relevance judgments for 2021 queries
  - A distinct list includes:
    - a query
    - the ranks of retrieved advertisements
    - The clickthrough rate at each rank
- Dependency of Clicks on Entire Result List

  e.g. rel_1 = Excellent



9

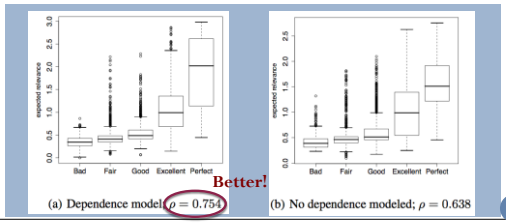
Experiments 1 -- relevance prediction model

Compare two models

dependence model: $p(\mathbf{X} \mid q, \mathbf{c}) = \prod p(X_i \mid q, \mathbf{c})$

independence model: $p(\mathbf{X} \mid q, \mathbf{c}) = \prod p(X_i \mid q, c_i)$

Predicted relevance: $\mathrm{E}(X_i)$



10

Experiments 2 -- estimating DCG

- Methodology
  - Test sets: lists with complete relevance judgments and >500 impressions (1720 lists)
  - Train sets: remaining lists with >200 impressions (>5000 lists)
  - Ground-truth: DCG
  - Predictions: $E(\hat{DCG})$, $E(CTR) = \frac{1}{k}\sum c_k$
- Experiment 2-1

  $Cov[E(CTR), DCG] = 0.662$

  $Cov[E(\hat{DCG}), DCG] = 0.876$ **Better!**

11

Experiments 2 -- estimating DCG

- Experiment 2-2

  without vs. with additional two manual judgments on documents recommended by the system

Confidence in ΔDCG: P(ΔDCG<0)

Confidence	0.5 – 0.6	0.6 – 0.7	0.7 – 0.8	0.8 – 0.9	0.9 – 0.95	0.95 – 1.0
Accuracy clicks-only	0.522	0.617	0.734	0.818	–	–
Accuracy 2 judgments	0.572	0.678	0.697	0.890	0.918	0.940

12

Summary

- Propose a method to evaluate search engines by modeling the relationship between relevance and clicks
  - Predict relevance using clicks
    - Dependence model $p(\mathbf{X} \mid q, \mathbf{c}) = \prod_{i=1}^{l} p(X_i \mid q, \mathbf{c})$
  - Estimate DCG with the predicted relevance
  - Compare different rankings

13

Thank you !

Questions?

14

Appendix

Select Documents to Judge

- What if confident estimates are low?

  Obtain more relevance judgments from human.

  Intuitions:

  1. $r_1(i) = r_2(i)$ ignore
  2. $r_1(i) >> r_2(i) \,\|\, r_2(i) >> r_1(i)$ informative
  3. $rel_i > rel_j$

$$\Delta DCG_l = \sum_{i=1}^{N} g_{i1} - g_{i2}$$

$$g_{ij} = \begin{cases} rel_i & r_j(i) = 1 \\ \dfrac{rel_i}{\log_2 r_j(i)} & 1 < r_j(i) \le l \\ 0 & r_j(i) > l \end{cases}$$

Algorithm 1 Iteratively select documents to judge until we have high confidence in ΔDCG.

1: **while** $1 - \alpha \le P(\Delta DCG \le 0) \le \alpha$ **do**
2: $i^* \leftarrow \max_i |E[G_{i1}] - E[G_{i2}]|$ for all unjudged documents i
3: judge document i^*
(human annotator provides rel_{i^*})
4: $P(X_{i^*} = rel_{i^*}) \leftarrow 1$
5: $P(X_{i^*} \neq rel_{i^*}) \leftarrow 0$
6: estimate $P(\Delta DCG)$ using Monte Carlo simulation
7: **end while**

15