Lecture 1: Evaluation

Fernando Diaz
Microsoft Research NYC

September 15, 2014
Outline

Introduction to Evaluation

Batch Evaluation

Production Evaluation
Introduction to Evaluation
given a query and a corpus, find relevant documents.

query: user’s expression of the information need

_corpus: the repository of retrievable items

relevance: satisfaction of the information need
Evaluation

Fundamental issue in information retrieval
we can spend days discussing algorithms but we
need to quantify if they are good.
given a task is system A better than system B?
Many methods of information retrieval evaluation
User study
Batch study
Production test
Each evaluation experiment has benefits and usually
we conduct many types of evaluations before making a
claim about a system.
Always compare system performance to a strong baseline.
User Study

Method: Provide a small group of users with several retrieval systems and ask them to complete several search tasks; interview users afterward to learn about system performance.

Advantages

- Very detailed data about users’ reaction to the systems.
- Can leverage experimental methodology from psychology.

Drawbacks

- Costly to run user studies (pay users, scientist time, data coding).
- Difficult to generalize from small studies to broad populations.
- Laboratory experiments are often not representative of the normal user context.
- Need to rerun an experiment when a new system is being considered.
Batch study

**Method:** Gather a small pool of ‘test queries’ and judge the relevance of documents in the corpus; compare systems on their ability to rank relevant documents above non-relevant documents.

**Advantages**
- Allows repeatable experiments; we can compare systems on the same queries and judgments.
- Can construct data sets large enough to conduct significance tests on performance metrics.

**Drawbacks**
- Costly to get judgments (pay editors).
- Judgments gathered in synthetic environment, often not by the users generating the queries.
- Assumes relevance is the same across users.
- Batch studies underly the majority of information retrieval evaluation (core evaluation in much of TREC).
Production test

**Method:** In a production system, have x% of the traffic use system A and y% use system B; compare system effects on logged user interaction.

**Advantages**
System usage is naturalistic; users are not situated in a lab and often are not aware that a test is being conducted.
Can construct very large data sets.

**Drawbacks**
Requires a very good understanding of interpreting positive and negative user experience from logging data (do we want to just measure user retention? clicks?)
Experiments are very difficult to repeat.
Increasingly, this is how real world systems are evaluated; ‘user studies on steroids’.
Batch Evaluation
**Batch Data Gathering Procedure**

1. Generate a set of queries on which you want to test system performance.
2. Gather a set of documents of each query for which you want to judge relevance.
3. Judge the relevance of each document to the query.
Batch Data Gathering Procedure

Example

1. \{facebook, yahoo, google, bing\}

2. \{\{facebook.com,cnn.com/facebook-news.html,...\},\{cnn.com/yahoo-news.html,yahoo.com,...\},...\}

3. \{\{R,N,...\},\{N,R,...\},...\}
Batch Data Gathering Procedure

Queries

Sources of queries

- System experimenter hand-crafting a small pool.
- User study.
- Gather from a production query log.

We must be very careful about the queries we select because they represent the queries upon which we want to system to perform well.

What’s implication of the queries sampled in the previous example?

Usually we sample queries according to how often we expect to find them in the query traffic for our system.
Batch Data Gathering Procedure

Documents

Source of documents: the corpus

Problem: we cannot judge the relevance of every document in the corpus.

Solutions

judge a random sample of documents from the corpus.

what is the problem if queries are highly specialized?

only evaluate the top documents from the retrievals you are evaluating (pooling).

what does this imply about reproducibility?

manually try to find all relevant documents for a topic.

Unjudged documents are often considered to be non-relevant since the majority of the collection is non-relevant.
Batch Data Gathering Procedure

Judging Relevance

Relevance
Construct a clear and complete definition of what makes a document relevant to a query; it’s too late to change after judging has already started! Relevance can be defined according to the task: is the document relevant if it mentions the topic? is it relevant if it only mentions the topic? Will the user only be satisfied by a single document? Relevance can binary (relevant vs non-relevant) or graded (degree of relevance).

Editors
Train the editors (often require a few queries to get familiar with the relevance guidelines). Decide number of editors per query (more editors risks misinterpretation of the query).
Batch Data Gathering Output

\[ D = \{(q, r)\} \]

- \( q \) judged query
- \( r \) map from documents to judgments
Batch Data Evaluation Procedure

Evaluate-System($f, \mathcal{D}$)

1. $\mathcal{E} \leftarrow \{\}$
2. for $(q, r) \in \mathcal{D}$
   3. do
   4. $\pi \leftarrow \text{Rank}(f, \text{Keys}(r))$
   5. $e \leftarrow \text{Evaluate-Ranking}(\pi, r)$
   6. $\mathcal{E} \leftarrow \mathcal{E} \cup \{e\}$
7. return $\text{Aggregate-Metric}(\mathcal{E})$
Defining Evaluate-Ranking

Much of the work in information retrieval evaluation has focused on defining appropriate evaluation metrics for rankings.

A metric is evaluated by its correlation with user objectives (e.g. success, satisfaction, happiness).
Dog Food - Natural Dog Food and Organic Dog Food Available Online from Petco.com. Take advantage of the lowest prices on natural cat food and organic cat food.

Cat Food 101: What You Need to Know About Feeding Your Cat
pets.webmd.com/cats/.../cat-food-101-what-you-need-to-know-about...
Here’s an introduction to cat food, where we answer frequently asked questions from cat owners about cat food and feeding.

CANIDAE® All Natural Holistic Dog Food - All Life Stages dry formula
www.canidae.com/dogs/all_life_stages/dry.html
All Natural Dog Food Formula made with 4 Meat Meals. Dogs Love the Taste. Whole Grain Brown Rice and Diversified Carbohydrates Provide More Wholesome ...

Evermore Pet Food • Evermore
evermorepetfood.com/
Our gently cooked, nutritionally balanced formulas are carefully sourced using minimally processed, whole-food ingredients to improve upon the time-honored ...

Buy Iams Cat Food | Iams
www.iams.com/cat-food/
Find online and local retailers who carry the Iams dog food you’re looking for.
Defining Evaluate-Ranking

Kendall’s $\tau$

Statistical measure of the correlation between two rankings developed in 1938.
Infer an optimal ranking from the relevance judgments.
Compare the system’s ranking to the optimal ranking.
Comparison looks at the number of swapped pairs of items.
Defining Evaluate-Ranking

Kendall’s $\tau$

Evaluate-System-Kendall($\pi, r$)

1. $C \leftarrow 0$
2. $D \leftarrow 0$
3. $\rho^* \leftarrow \text{Sort}(r)$
4. for $(i, j) \in \text{Document-Pairs}(\pi)$
   5. do
   6. if $(\pi^{-1}(i) > \pi^{-1}(j)) \land (\rho^{-1}(i) > \rho^{-1}(j))$
   7. then
   8. $C \leftarrow C + 1$
   9. else
   10. $D \leftarrow D + 1$
11. return $\frac{C-D}{C+D}$
(a) $\rho^*$
(a) $\rho^*$  (b) $\tau = 0.64$  (c) $\tau = 0.47$
(a) $\rho^*$  
(b) $\tau = 0.64$  
(c) $\tau = 0.47$  
(d) $\tau = -0.07$
Defining Evaluate-Ranking
Kendall’s $\tau$

Kendall’s $\tau$ provides a nice formalism for evaluation but its user model is somewhat abstract. Do users really care about swaps at the bottom of the ranked list? What about ties?
Defining Evaluate-Ranking

Precision

Precision measures the amount of relevant content in the ranking provided by the system.
In set-based retrieval, we can count the number of relevant documents in the retrieved set.
In rank-based retrieval, we can count the number of relevant documents in the top of the ranked list.
Defining Evaluate-Ranking

Precision at $k$ documents

Evaluate-System-Precision@K($\pi, r, K$)

1 $RR \leftarrow 0$
2 \textbf{for} $0 < i < K$
3 \textbf{do}
4 \hspace{1em} \textbf{if} $r(\pi(i))$
5 \hspace{2em} \textbf{then}
6 \hspace{3em} $RR \leftarrow RR + 1$
7 \textbf{return} $\frac{RR}{K}$
(a) $P@1 = 1.0$
(b) $P@1 = 1.0$
(c) $P@1 = 0.0$
(d) $P@1 = 0.0$
(a) $P@4 = ?$

(b) $P@4 = ?$

(c) $P@4 = ?$

(d) $P@4 = ?$
P@4

(a) P@4 = 1.0
(b) P@4 = 0.5
(c) P@4 = 0.5
(d) P@4 = 0.0
(a) $P@10 = 0.40$  (b) $P@10 = 0.40$  (c) $P@10 = 0.40$  (d) $P@10 = 0.40$
Defining Evaluate-Ranking

Precision

What happens to precision as we increase $K$?

Precision is good if the user only cares about retrieving a few relevant documents.

In some cases, users are interested in retrieving all relevant documents (e.g. researching for a paper, legal search, medical informatics).
Defining Evaluate-Ranking

Recall

Recall measures the coverage of relevant content in the ranking provided by the system.
In set-based retrieval, we can count the fraction of relevant documents which have been retrieved.
In rank-based retrieval, we can count the fraction of relevant documents which have been retrieved in the top of the ranked list.
Defining Evaluate-Ranking

Recall at \( k \) documents

Evaluate-System-Recall@K(\( \pi, r, K \))

1. \( RR \leftarrow 0 \)
2. \( R \leftarrow \text{Count-Relevant}(r) \)
3. for \( 0 < i < K \)
4. do
5. if \( r(\pi(i)) \)
6. then
7. \( RR \leftarrow RR + 1 \)
8. return \( \frac{RR}{R} \)
(a) $R@1 = ?$
(b) $R@1 = ?$
(c) $R@1 = ?$
(d) $R@1 = ?$
(a) $R@1 = 0.25$  
(b) $R@1 = 0.25$  
(c) $R@1 = 0.00$  
(d) $R@1 = 0.00$
(a) \( R@4 = ? \)
(b) \( R@4 = ? \)
(c) \( R@4 = ? \)
(d) \( R@4 = ? \)
R@4

(a) R@4 = 1.0
(b) R@4 = 0.5
(c) R@4 = 0.5
(d) R@4 = 0.0
R@10

(a) R@10 = 1.0
(b) R@10 = 1.0
(c) R@10 = 1.0
(d) R@10 = 1.0
What happens to recall as we increase $K$? Recall is good if the user only cares about retrieving all relevant documents.
Oftentimes users want a ranking with high precision and high recall. We can compute the harmonic mean of precision and recall. The weighted harmonic mean of precision and recall is the F-Measure.
Defining Evaluate-Ranking

F-Measure at $k$ documents

Evaluate-System-F@K($\pi$, $r$, $K$, $\alpha$)

1. $P \leftarrow \text{Evaluate-System-Precision@K}(\pi, r, K)$
2. $R \leftarrow \text{Evaluate-System-Recall@K}(\pi, r, K)$
3. return $\left(\alpha \left(\frac{1}{P}\right) + (1 - \alpha) \left(\frac{1}{R}\right)\right)^{-1}$
$F_{0.50}$
$\textit{F}_{0.90}$
$F_{0.10}$
Defining Evaluate-Ranking

F-Measure

How does the metric behave as we increase $\alpha$?
The F-Measure is a good summary number for a retrieval to present.
However, it conflates information about precision and recall.
Defining Evaluate-Ranking
Precision-Recall Graph

The problem with metrics measured at $K$ documents is that we must pick $K$.

sometimes this is makes sense given a task/user model.

If we expect a diversity of search lengths, then we should inspect several values of $K$. 
Defining Evaluate-Ranking

Precision-Recall Graph

Evaluate-System-PrecisionRecallGraph($\pi, r$)

1. for $0 < i < \text{Length}(\pi)$
2. do
3. $P \leftarrow \text{Evaluate-System-Precision@K}(\pi, r, i)$
4. $R \leftarrow \text{Evaluate-System-Recall@K}(\pi, r, i)$
5. $PR \leftarrow PR \cup \{(P, R)\}$
6. return $\text{Plot}(PR)$
Defining Evaluate-Ranking

Precision-Recall Graph
Another solution to dealing with selecting $K$ is to measure precision along different points of the ranked list and then aggregate. One method is to compute precision at each relevant document.
Defining Evaluate-Ranking

Average Precision

Evaluate-System-AveragePrecision(\( \pi, r \))

1. \( AP \leftarrow 0 \)
2. \( RR \leftarrow 0 \)
3. for \( 0 < i < \text{Length}(\pi) \) do
   4. if \( r(\pi(i)) \) then
      5. \( RR \leftarrow RR + 1 \)
      6. \( AP \leftarrow AP + \frac{RR}{i} \)
6. return \( \frac{AP}{RR} \)
Average Precision

1.0  1.0  0.5  0.4
Average Precision

(a) AP = 1.00  
(b) AP = 0.74  
(c) AP = 0.39  
(d) AP = 0.28
Defining Evaluate-Ranking

Average Precision

If we expect increasingly more non-relevant documents deeper in the ranking, this will have the effect of weighting the top of the ranked list more. Average precision is closely related to the area under the precision recall curve.

Can we be more explicit about the position weighting?
Defining Evaluate-Ranking

Reciprocal Rank

If the user will be satisfied after one relevant document, we can evaluate according to the position of the first relevant document. We want to aggressively penalize the system if the first relevant document is deep in the ranking.
Defining Evaluate-Ranking

Reciprocal Rank

Evaluate-System-ReciprocalRank(\(\pi, r\))

1  for 0 < \(i\) < Length(\(\pi\))
2     do
3       if \(r(\pi(i))\)
4             then
5                 return \(\frac{1}{i}\)
6     return 0
Reciprocal Rank

(a) RR = 1.00

(b) RR = 1.00

(c) RR = 0.33

(d) RR = 0.14
Reciprocal rank is nice because it models the importance of the top of the ranked list to the user. However, the assumption of satisfaction after one relevant document is brittle.

What if the judgments are graded?
What if the user would like more than one relevant document?
Defining Evaluate-Ranking
Discounted Cumulative Gain

In order to deal with graded relevance and position bias, we can explicitly model relevance and degradation of performance.


<table>
<thead>
<tr>
<th>grade</th>
<th>gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perfect</td>
<td>10</td>
</tr>
<tr>
<td>Excellent</td>
<td>7</td>
</tr>
<tr>
<td>Good</td>
<td>5</td>
</tr>
<tr>
<td>Fair</td>
<td>1</td>
</tr>
<tr>
<td>Bad</td>
<td>0</td>
</tr>
</tbody>
</table>
Defining Evaluate-Ranking
Discounted Cumulative Gain

Evaluate-System-DCG(\(\pi, r, K\))

1. \(DCG \leftarrow 0\)
2. \(\text{for } 0 < i < K\)
3. \(\text{do}\)
4. \(DCG \leftarrow DCG + \frac{\text{Gain}(r(\pi(i)))}{\log_2(i+1)}\)
5. \(\text{return } DCG\)
Discounted Cumulative Gain

(a) DCG = 1.00
(b) DCG = 0.89
(c) DCG = 0.59
(d) DCG = 0.48
Defining Evaluate-Ranking

Discounted Cumulative Gain

\[ w = \frac{1}{\log_2(r + 1) - 1} \]

Satisfies a clear user browsing model and incorporation of graded judgments.

How to derive accurate discounting and grade values is an open area of research.
## Summary

<table>
<thead>
<tr>
<th></th>
<th>$\tau$</th>
<th>P@3</th>
<th>R@3</th>
<th>$F_{0.50}$</th>
<th>AP</th>
<th>RR</th>
<th>NDCG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td>0.75</td>
<td>0.88</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>0.64</td>
<td>0.66</td>
<td>0.50</td>
<td>0.57</td>
<td>0.74</td>
<td>1.00</td>
<td>0.89</td>
</tr>
<tr>
<td></td>
<td>0.47</td>
<td>0.33</td>
<td>0.25</td>
<td>0.29</td>
<td>0.39</td>
<td>0.33</td>
<td>0.59</td>
</tr>
<tr>
<td></td>
<td>-0.07</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.28</td>
<td>0.14</td>
<td>0.48</td>
</tr>
</tbody>
</table>
Batch Data Evaluation Procedure

Evaluate-System\((f, \mathcal{D})\)

1. \(\mathcal{E} \leftarrow \{\}\)
2. \textbf{for} \((q, r) \in \mathcal{D}\)
3. \quad \textbf{do}
4. \quad \pi \leftarrow \text{Rank}(f, \text{Keys}(r))
5. \quad e \leftarrow \text{Evaluate-Ranking}(\pi, r)
6. \quad \mathcal{E} \leftarrow \mathcal{E} \cup \{e\}
7. \textbf{return} \text{Aggregate-Metric}(\mathcal{E})
Defining Aggregate-Metric

Given a measure of performance across queries, how do we aggregate the metric?
The choice of aggregation reflects our desire system robustness.

- \( \max(\mathcal{E}) \): very risky; performance may be poor for other queries.
- \( \min(\mathcal{E}) \): very conservative; poor performance may be an anomaly.
- \( \frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} e \): arithmetic mean; incorporates performance on all queries.
- \( \prod_{e \in \mathcal{E}} e^{\frac{1}{|\mathcal{E}|}} \): geometric mean; prefers strong performance on all queries.
Empirical fact that as recall increases, precision decreases on average.

Therefore, we first convert query-level precision-recall curves into monotonically decreasing step functions. We then interpolate precision of these functions at fixed recall points (usually \{0, .1, \ldots, .9, 1\})
Defining Evaluate-Ranking

Averaging Precision-Recall Graphs
Evaluation When Data Is Needed For Parameter Tuning

When data is required to tune system parameters, we must not tune parameters on the evaluation set.

This is almost always the case with machine learned ranking models. Hurts generalizability since we may be tuning parameters for the evaluation set.

In order to address this, we split the data set into a training set (for setting system parameters) and a testing set (for evaluating system performance).

In practice, we use cross-validation to reduce the variance of performance estimates.
Evaluation When Data Is Needed For Parameter Tuning

Cross-Validation

Cross-Validated-Evaluation($\mathcal{D}, K$)

1. $\mathcal{F} \leftarrow$ Random-Partition($\mathcal{D}, K$)
2. $\mathcal{E} \leftarrow \{\}$
3. for $\mathcal{D}' \in \mathcal{F}$
4. do
5. $f \leftarrow$ Train-Parameters($\mathcal{D} - \mathcal{D}'$)
6. $e \leftarrow$ Evaluate-System($f, \mathcal{D}'$)
7. $\mathcal{E} \leftarrow \mathcal{E} \cup \{e\}$
8. return Aggregate-Metric($\mathcal{E}$)
Comparing Systems

We usually care more about the relative performance of systems, not just the absolute value of the metric. We *must* compare system performance on exactly the same data set $\mathcal{D}$. Comparing raw aggregated metrics is subject to statistical noise.

Because we would like to be scientific about our conclusions, we require significance testing.
Comparing Systems

In order to test the significance of a difference in performance, we use paired statistical tests,

- t-test
- Wilcoxon signed rank test
- bootstrap

Pairing is performed on evaluation queries. Any evaluation in this course must execute statistical testing of differences and report $p$-values.

The R software package contains all of the routines you need to perform these tests (http://www.R-project.org).
Is the difference significant?

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>query 1</td>
<td>.1</td>
<td>.0</td>
<td>.1</td>
</tr>
<tr>
<td>query 2</td>
<td>.7</td>
<td>.9</td>
<td>.8</td>
</tr>
<tr>
<td>query 3</td>
<td>.3</td>
<td>.2</td>
<td>.3</td>
</tr>
<tr>
<td>query 4</td>
<td>.4</td>
<td>.2</td>
<td>.4</td>
</tr>
<tr>
<td>query 5</td>
<td>.9</td>
<td>.3</td>
<td>.9</td>
</tr>
<tr>
<td>query 6</td>
<td>.4</td>
<td>.1</td>
<td>.4</td>
</tr>
<tr>
<td>query 7</td>
<td>.5</td>
<td>.6</td>
<td>.5</td>
</tr>
<tr>
<td>query 8</td>
<td>.9</td>
<td>.1</td>
<td>.9</td>
</tr>
<tr>
<td>query 9</td>
<td>.5</td>
<td>.1</td>
<td>.5</td>
</tr>
<tr>
<td>query 10</td>
<td>.3</td>
<td>.4</td>
<td>.3</td>
</tr>
<tr>
<td>average</td>
<td>0.50</td>
<td>0.29</td>
<td>0.51</td>
</tr>
</tbody>
</table>
Is the difference significant? 

using R

```r
> a=c(.1,.7,.3,.4,.9,.4,.5,.9,.5,.3)
> b=c(0,.9,.2,.2,.3,.1,.6,.1,.1,.4)
> wilcox.test(a,b,"greater")

Wilcoxon rank sum test with continuity correction

data:  a and b
W = 74.5, p-value = 0.03356
alternative hypothesis: true location shift is greater than 0
```
Is the difference significant?

using R

```r
> a = c(.1, .7, .3, .4, .9, .4, .5, .9, .5, .3)
> c = c(.1, .8, .3, .4, .9, .4, .5, .9, .5, .3)
> wilcox.test(c, a, "greater")
```

Wilcoxon rank sum test with continuity correction

data:  c and a
W = 50.5, p-value = 0.5
alternative hypothesis: true location shift is greater than 0
Is the difference significant? using R

\[ t\text{.test}(a, b, "greater") \]

Welch Two Sample t-test

data:  a and b
t = 1.7413, df = 17.95, p-value = 0.04937
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:  
0.0008398375   Inf
sample estimates:  
mean of x mean of y  
0.50       0.29
Is the difference significant?

using R

> t.test(c,a,"greater")

Welch Two Sample t-test

data:  c and a
t = 0.0836, df = 17.974, p-value = 0.4672
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
  -0.1975412       Inf
sample estimates:
  mean of x  mean of y
0.51        0.50
Comparing Systems

In general, improvements are considered significant if $p < 0.05$.
Computing the significance is easy.
Finding significant results is difficult.
Summary

Batch evaluation is the principle way of testing systems without negatively impacting users.
Still used as an offline method for studying effectiveness.
Concepts underlie some production evaluation metrics.
Production Evaluation
Production Data Gathering Methods

Production web search engines can—and should—log as much information as possible which is related to system performance (subject to privacy terms of service).

Simplest/strongest source of information is *click logging*.

indicating user interest in a search result.

can be logged with easy architecture.
Click Logging with a Redirect Server
Click Logging with a Beacon Server

Dog Food - Natural Dog Food and Organic Dog Food Available Online from Petco.com. Take advantage of the lowest prices on natural cat food and organic cat food...

Cat Food 101: What You Need to Know About Feeding Your Cat
www.pets.webmd.com/cats/cat-food-101-what-you-need-to-know-about...
Here's an introduction to cat food, where we answer frequently asked questions from cat owners about cat food and feeding.

CANIDAE® All Natural Holistic Dog Food - All Life Stages dry formula
www.canidae.com/dogs/all_life_stages/dry.html
All Natural Dog Food Formula made with 4 Meat Meals. Dogs Love the Taste. Whole Grain Brown Rice and Diversified Carbohydrates Provide More Wholesome...

Evermore Pet Food • Evermore
www.evermorepetfood.com/
Our gently cooked, nutritionally balanced formulas are carefully sourced using minimally processed, whole-food ingredients to improve upon the time-honored...

Buy Iams Cat Food | Iams
www.iams.com/cat-food/
Find online and local retailers who carry the Iams dog food you're looking for.

$('#srtitle').delegate('a', 'click', function(event) {
  var url = this.href;
  // send source_data and target_data to beacon server
  window.location = url;
});
Click Logging with a Beacon Server
**Hypothesis:** observing a click on \( d \) for query \( q \) implies that \( d \) is relevant to \( q \).

allows reuse of batch evaluation metrics.
a lot of evidence supporting this hypothesis.
correlation is not perfect.
Click as Relevance

\[ C = \{ \langle q_0, d_0, 1 \rangle, \langle q_0, d_1, 0 \rangle, \langle q_0, d_2, 1 \rangle, \ldots, \langle q_0, d_0, 0 \rangle, \langle q_0, d_1, 1 \rangle, \langle q_0, d_2, 1 \rangle, \ldots, \langle q_1, d_0, 1 \rangle, \langle q_1, d_1, 0 \rangle, \langle q_1, d_2, 0 \rangle, \ldots \} \]

\[ C^* = \{ \langle q_0, d_0, .90 \rangle, \langle q_0, d_1, .75 \rangle, \langle q_0, d_2, .50 \rangle, \ldots, \langle q_1, d_0, .99 \rangle, \langle q_1, d_1, .40 \rangle, \langle q_1, d_2, .10 \rangle, \ldots \} \]
Click as Relevance

Clicks are Noisy

**accidental clicking:** need to detect and remove clicks which immediately return to the search results page.

**malicious clicking:** need to detect and remove clicks from robots trying to manipulate the system.

**position bias:** need to account for documents at higher positions being clicked first.

**presentation bias:** need to account for documents or results that are visually more attractive.

**ambiguous intent:** need to account for clicks representing different query intents.
Click as Relevance
Dealing with Position Bias

User scan behavior starts from the top of the ranked list and moves serially downward.
Documents at the top tend to receive more examinations and clicks.
Instead of modeling the ‘click’ or ‘no click’, we can model clicks conditioned on examination.
Click as Relevance
Dealing with Position Bias

$S$  user start state

$R$  user satisfied

Model parameters, $p(x_i)$, can be learned efficiently from large amounts of log data.
Click as Preference

**Hypothesis**: if $d_i$ is positioned above $d_{i+k}$ and we observe no click on $d_i$ we observe a click on $d_{i+k}$, then $d_{i+k}$ is more relevant than $d_i$ for $q$.

assumes top-down, serial scanning of ranked list.
addresses position bias effects.
preference labels tend to be less noisy than point-wise relevance labels.
correlation is not perfect.
Click as Preference

\[ C = \{ \langle q_0, d_0, 1 \rangle, \langle q_0, d_1, 0 \rangle, \langle q_0, d_2, 1 \rangle, \ldots, \]
\[ \langle q_0, d_0, 0 \rangle, \langle q_0, d_1, 1 \rangle, \langle q_0, d_2, 1 \rangle, \ldots, \]
\[ \langle q_1, d_0, 1 \rangle, \langle q_1, d_1, 0 \rangle, \langle q_1, d_2, 0 \rangle, \ldots \} \]

\[ C^* = \{ \langle q_0, d_0 \succ d_1, .75 \rangle, \langle q_0, d_0 \succ d_2, .99 \rangle, \langle q_0, d_0 \succ d_3, .50 \rangle, \ldots, \]
\[ \langle q_1, d_0 \succ d_1, .99 \rangle, \langle q_1, d_0 \succ d_2, .99 \rangle, \langle q_1, d_0 \succ d_3, .99 \rangle, \ldots \} \]

\((d_i \succ d_j: d_i \text{ is preferred to } d_j.\)
Click as Preference
Gathering Data

A fixed presentation order will result in a biased sampling of possible preferences. We will never observe data for $d_1 \succ d_2$. This can be addressed with randomization.
Click as Preference

Summary

reliable representation of relative relevance data.
no absolute relevance score.
no clear mapping to classic evaluation metrics except $\tau$. 
Comparing Systems

Using click logs from a single production system introduces biases in relevance or preference estimates. Batch experiments address biased samples by pooling. Production experiments addressed biased samples by selective system application (e.g. A/B testing).
A/B Testing

Bucket-Test($\{f_0, \ldots, f_n\}, \{w_0, \ldots, w_n\}, U, T$)

1. $\{U_0, \ldots, U_n\} \leftarrow \text{Partition-Users}(U, \{w_i\})$
2. apply $f_i$ to user population $U_i$ for duration $T$.
3. $\{C_1, \ldots, C_n\} \leftarrow \text{Collect-Logs}(\{U_i\})$
4. $\{\mu_1, \ldots, \mu_n\} \leftarrow \text{Compute-Metrics}(\{C_i\})$
A/B Testing

(a) A (25%)
(b) B (50%)
(c) C (25%)
A/B Testing

Powerful tool for comparing system performance. Because the types of queries change with time (and location and demographics and ...), partitioning has to be very carefully performed.
Interleaving

**Hypothesis:** We can more efficiently use interaction data by explicitly comparing systems and exploiting the ranked list structure of web search.

- A/B testing computes metric for each bucket and then compares system metrics.
- More efficient to directly compare systems.
Interleaving

(a) A vs. B (25%)
(b) A vs. C (50%)
(c) B vs. C (25%)
Interleaving

**Input:** Rankings $A = (a_1, a_2, \ldots)$ and $B = (b_1, b_2, \ldots)$

$I \leftarrow () ; k_a \leftarrow 1 ; k_b \leftarrow 1 ;$

$AFirst \leftarrow \text{RandomBit}()$ \hfill \text{decide which ranking gets priority}

**while** $(k_a \leq |A|) \land (k_b \leq |B|)$ **do** \hfill \text{if not at end of A or B}

\hspace{1em} **if** $(k_a < k_b) \lor ((k_a = k_b) \land (AFirst = 1))$ **then**

\hspace{2.5em} **if** $A[k_a] \not\in I$ **then** $I \leftarrow I + A[k_a]$ \hfill \text{append next A result}

\hspace{2.5em} $k_a \leftarrow k_a + 1$

\hspace{2.5em} **else**

\hspace{4em} **if** $B[k_b] \not\in I$ **then** $I \leftarrow I + B[k_b]$ \hfill \text{append next B result}

\hspace{4em} $k_b \leftarrow k_b + 1$

**end if**

**end while**

**Output:** Interleaved ranking $I$

[Chapelle et al. 2012]
Interleaving

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>342</td>
<td>124</td>
<td>342</td>
</tr>
<tr>
<td>576</td>
<td>213</td>
<td>124</td>
</tr>
<tr>
<td>234</td>
<td>543</td>
<td>576</td>
</tr>
<tr>
<td>943</td>
<td>765</td>
<td>213</td>
</tr>
<tr>
<td>472</td>
<td>852</td>
<td>234</td>
</tr>
</tbody>
</table>

(a) A
(b) B
(c) interleaved
Interleaving

\[
\left( E \left[ \frac{C_A - C_B}{C} \right] > 0 \right) \rightarrow (A \succ B)
\]

\[
\left( E \left[ \frac{C_A - C_B}{C} \right] < 0 \right) \rightarrow (B \succ A)
\]

\[ C_i \text{ total clicks on results from } i \]
\[ C \text{ total clicks} \]
Correlated with editorial relevance target at lower cost. Converges much faster than editorial data.
Summary

Production evaluation is increasingly used in industry as the method for testing and deploying large systems. Many ways in which you can make mistakes so you have to be very careful interpreting results.
Summary

Evaluation is a **fundamental** part of building and understanding a production search engine. Many issues are still actively being researched.
Dealing with ambiguous queries.
Personalized metrics.
Multidimensional relevance (e.g. local, temporal).