

Significance Testing

Evaluation, session 6

Statistical Significance

IR and other experimental sciences are concerned with measuring the effects of competing systems and deciding whether they are really different.

For instance, “Does stemming improve my results enough that my search engine should use it?”

Statistical hypothesis testing is a collection of principled methods for setting up these tests and making justified conclusions from their results.

Hypothesis Testing

In statistical hypothesis testing, we try to isolate the effect of a single change so we can decide whether it makes an impact.

The test allows us to choose between the **null hypothesis** and an **alternative hypothesis**.

The outcome of a hypothesis test does *not* tell us whether the alternative hypothesis is true. Instead, it tells us the probability that the null hypothesis could produce a “fake improvement” at least as extreme as the data you’re testing.

Null Hypothesis: what we believe by default – the change did not improve performance.

Alternative Hypothesis: the change improved performance.

The hypotheses we’re testing

Test Steps

1. Prepare your experiment carefully, with only one difference between the two systems: the change whose effect you wish to measure. Choose a **significance level** α , used to make your decision.
2. Run each system many times (e.g. on many different queries), evaluating each run (e.g. with AP).
3. Calculate a **test statistic** for each system based on the distributions of evaluation metrics.
4. Use a **statistical significance test** to compare the test statistics (one for each system). This will give you a **p-value**: the probability of the null hypothesis producing a difference at least this large.
5. If the p-value is less than α , reject the null hypothesis.

The probability that you will correctly reject the null hypothesis using a particular statistical test is known as its **power**.

Error Types

Hypothesis testing involves balancing between two types of errors:

- **Type I Errors**, or false positives, occur when the null hypothesis is true, but you reject it
- **Type II Errors**, or false negatives, occur when the null hypothesis is false, but you don't reject it.

The probability of a type I error is α – the significance level. The probability of a type II error is $\beta = (1 - \text{power})$.

What Can Go Wrong?

The power of a statistical test depends on:

- The number of *independent* runs (e.g. queries). In IR, we generally use 50 queries, but empirical studies suggest that 25 may be enough.
- Any bias in the experimental setup (are you using the wrong test collection?).
- Whether the true distribution of test statistic values matches the distribution assumed by your statistical test.

A common mistake is repeating a test until you get the p-value you want. Repeating a test decreases its power.

Wrapping Up

For a very clear and detailed explanation of the subtleties of statistical testing, see the excellent guide “Statistics Done Wrong,” at:

<http://www.statisticsonewrong.com>.

In the next two sessions, we’ll look at two specific significance tests.

T-tests

Evaluation, session 7

T-Tests

There are many types of T-Tests, but here we'll focus on two:

- **One-sample tests** have a single distribution of test statistics, and compare its mean to some pre-determined value μ .
- **Paired-sample tests** compare the means of two systems on the same queries.

Each comes in two flavors:

- **One-tailed tests** ask whether the difference is $>\mu$ or $<\mu$, but not both (or whether the mean of one group is greater/less than the mean of the other).
- **Two-tailed tests** ask whether the mean $=\mu$ (or whether the means of the two samples are equal).

One-sample T-tests

Suppose you were developing a new type of IR system for your company, and your management decided that you can release it if its precision is above 75%.

To check this, run your system against 50 queries and record the mean of the precision values. Then calculate the t-value and p-value that correspond to your vector of precision values.

Let $\bar{x} :=$ the mean of the values

(we assume x is normally distributed

$s :=$ the std. dev. of the values

$n :=$ the number of samples

$\mu :=$ the target mean

Then:

$$t := \frac{\bar{x} - \mu}{(s/\sqrt{n})}$$

t is on the Student's t-distribution

with $n-1$ degrees of freedom

$$p := Pr(T > t)$$

Example: One-tailed T-test

$$\vec{x} = \begin{pmatrix} 0.74 \\ 0.82 \\ 0.71 \\ 0.76 \\ 0.79 \end{pmatrix}; n = 5; \bar{x} = 0.764; s = 0.0428; \mu = 0.75$$

$$t = \frac{0.764 - 0.75}{0.0383/\sqrt{5}} \\ = 0.7318$$

$$Pr(x > \mu) = 0.23$$

Let $\bar{x} :=$ the mean of the values

(we assume x is normally distributed)

$s :=$ the std. dev. of the values

$n :=$ the number of samples

$\mu :=$ the target mean

Then:

$$t := \frac{\bar{x} - \mu}{(s/\sqrt{n})}$$

t is on the Student's t-distribution

with $n-1$ degrees of freedom

$$p := Pr(T > t)$$

Example: Two-tailed T-test

$$\vec{x} = \begin{pmatrix} 0.74 \\ 0.82 \\ 0.71 \\ 0.76 \\ 0.79 \end{pmatrix}; n = 5; \bar{x} = 0.764; s = 0.0428; \mu = 0.75$$

$$t = \frac{0.764 - 0.75}{0.0383/\sqrt{5}} \\ = 0.7318$$

$$Pr(x = \mu) = 0.46$$

Only the p-value changes

Let $\bar{x} :=$ the mean of the values

(we assume x is normally distributed)

$s :=$ the std. dev. of the values

$n :=$ the number of samples

$\mu :=$ the target mean

Then:

$$t := \frac{\bar{x} - \mu}{(s/\sqrt{n})}$$

t is on the Student's t-distribution
with $n-1$ degrees of freedom

$$p := Pr(T = t)$$

Paired-Sample T-tests

Suppose you have runs from two different IR systems: a baseline run using a standard implementation, and a test run using the changes you're testing. You want to know whether your changes outperform the baseline.

To test this, run both systems on the same 50 queries using the same document collections and compare the difference in AP values per query.

Let $x_1 :=$ the baseline values

$x_2 :=$ the test values

$$\bar{d} := \overline{x_1 - x_2}$$

$$s_d := \text{stddev}(x_1 - x_2)$$

$n :=$ the number of samples

Then:

$$t := \frac{\bar{d}}{(s_d / \sqrt{n})}$$

t is on the Student's t-distribution
with $n-1$ degrees of freedom

$$p := \Pr(T > t)$$

Example: Paired-Sample T-test

$$\vec{x}_1 = \begin{pmatrix} 0.74 \\ 0.82 \\ 0.71 \\ 0.76 \\ 0.79 \end{pmatrix}; \vec{x}_2 = \begin{pmatrix} 0.77 \\ 0.86 \\ 0.74 \\ 0.72 \\ 0.77 \end{pmatrix}$$

$$\overline{x_1 - x_2} = 0.008; s_d / \sqrt{n} = 0.016$$

$$t = \frac{0.008}{0.016} \\ = 0.5020$$

$$Pr(\bar{x}_1 = \bar{x}_2) = 0.6421$$

Let $x_1 :=$ the baseline values

$x_2 :=$ the test values

$$\bar{d} := \overline{x_1 - x_2}$$

$$s_d := \text{stddev}(x_1 - x_2)$$

$n :=$ the number of samples

Then:

$$t := \frac{\bar{d}}{(s_d / \sqrt{n})}$$

t is on the Student's t-distribution
with $n-1$ degrees of freedom

$$p := Pr(T = t)$$

Wrapping Up

It's easy to glance at the data, see a bunch of bigger numbers, and conclude that your new system is working. You're often fooling yourself when you do this.

In order to really conclude that your new system is working, we need enough of the values to be “significantly” larger than the baseline values. A t-test will tell us whether the difference is big enough.

Next, we'll see what we can do if we don't want to assume that our data are normally-distributed.

Wilcoxon Signed Ranks Test

Evaluation, session 8

Nonparametric Significance Testing

The T-tests we used in the previous session assumed your data are normally-distributed. If they're not, the test has less power and you may draw the wrong conclusion.

The Wilcoxon Signed Ranks Test is *nonparametric*: it makes no assumptions about the underlying distribution. It has less power than a T-test when the data *is* normally distributed, but more power when it isn't.

This test is based on comparing the rankings of the data points implied by their evaluation measure (e.g. AP).

The Signed Ranks Test

1. Produce a vector of the differences between values for each point.
2. Sort the vector by absolute value.
3. Replace the values with their ranks, but keep the signs. (If there are duplicate values, use the mean of the ranks for all values with the appropriate sign).
4. The test statistic is the sum of these signed ranks.

This algorithm produces a discrete distribution that approximates a Normal distribution with mean 0.

If we have at least 10 samples, we can use the algorithm on the next slide to obtain a p-value.

Example: Signed Ranks

1. Produce a vector of the differences between values for each point.
2. Sort the vector by absolute value.
3. Replace the values with their ranks, but keep the signs. (If there are duplicate values, use the mean of the ranks for all values with the appropriate sign).
4. The test statistic is the sum of these signed ranks.

$$\vec{x}_1 = \begin{pmatrix} 0.74 \\ 0.82 \\ 0.71 \\ 0.76 \\ 0.79 \end{pmatrix}; \vec{x}_2 = \begin{pmatrix} 0.77 \\ 0.86 \\ 0.74 \\ 0.72 \\ 0.77 \end{pmatrix}$$

$$\vec{x}_2 - \vec{x}_1 = \begin{pmatrix} -0.03 \\ -0.04 \\ -0.03 \\ 0.04 \\ 0.02 \end{pmatrix}; \vec{r} = \begin{pmatrix} 1.5 \\ -1.5 \\ -3.5 \\ -3.5 \\ 5 \end{pmatrix}$$

Calculating Z-Ratios

$$W = \sum_{i=1}^n r_i$$

where r_i are the signed ranks

$$\mu_W = 0$$

the mean of the dist. for W

$$\sigma_W = \sqrt{\frac{n(n+1)(2n+1)}{6}}$$

the std. dev. of the dist. for W

$$z = \frac{(W - \mu_W) \pm 0.5}{\sigma_W}$$

0.5 added when W neg., subtracted when pos.

$$= \frac{W - 0.5}{\sigma_W}$$

when $W > 0$

Using Z-Ratios

The table shows the p-values that correspond to various z-ratios.

One-sided tests ask whether the difference is greater or less than zero; two-sided tests ask whether the difference is nonzero.

abs(z-ratio)

| | | | | |
|-------|------|-------|-------|-------|
| 1.645 | 1.96 | 2.326 | 2.576 | 3.291 |
|-------|------|-------|-------|-------|

One-sided Test p-values

| | | | | |
|-----|-------|------|-------|--------|
| 0.5 | 0.025 | 0.01 | 0.005 | 0.0005 |
|-----|-------|------|-------|--------|

Two-sided Test p-values

| | | | | |
|---|------|------|------|-------|
| — | 0.05 | 0.02 | 0.01 | 0.001 |
|---|------|------|------|-------|

Example

$$\vec{x}_1 = \begin{pmatrix} 0.74 \\ 0.82 \\ 0.71 \\ 0.76 \\ 0.79 \end{pmatrix}; \vec{x}_2 = \begin{pmatrix} 0.77 \\ 0.86 \\ 0.74 \\ 0.72 \\ 0.77 \end{pmatrix}$$

This is the same example used in the two-sample t-test.

These samples are simply too close to justify rejecting the null hypothesis.

$$(\vec{x}_2 - \vec{x}_1)^T = (0.03 \quad 0.04 \quad 0.03 \quad -0.04 \quad -0.02)$$
$$\text{abssort}(\vec{x}_2 - \vec{x}_1)^T = (-0.02 \quad 0.03 \quad 0.03 \quad 0.04 \quad -0.04)$$

$$r^T = (-1 \quad 2.5 \quad 2.5 \quad 4.5 \quad -4.5)$$

$$W = \sum r_i = 4$$

$$\sigma_W = \frac{\sqrt{5(5+1)(2 \cdot 5 + 1)}}{6} = 3.0277$$

$$z = \frac{W - 0.5}{\sigma_W} = 1.156$$

$$p > 0.05$$

Wrapping Up

The Wilcoxon Signed Ranks Test is a better choice when your data aren't normally-distributed.

It produces a distribution of signed ranks which approximates a normal distribution as the number of samples increases. For the TREC standard of 50 queries, this approximation is quite good.

For the rest of the module, we'll look at how to conduct user studies for system evaluation.

Implicit User Studies

Evaluation, session 9

Evaluation Datasets

There are several major sources of data for evaluating IR systems:

- Test collections, such as TREC data
- Search engine log data
- User studies in the lab
- Crowdsourcing studies (e.g. Amazon Mechanical Turk)

We've covered test collections. We'll focus on search engine log data now, and discuss explicit user studies in the next session.

Real People, Real Queries

Search engine query logs are massive, and can be grouped in many ways to focus on different aspects of IR.

They provide a real picture of what users actually do when performing various search tasks.

BUT we can't talk to the users, don't have demographic information, and don't know what the users were trying to accomplish. And this data is generally only available to search engine employees and their collaborators.

Available Data

Users generate a lot of data by interacting with search engines. Consider what can be inferred from the following interactions.

- A user runs a query, slowly scrolls through the list, then runs a new query with additional terms added.
- A user runs a query and, 10 seconds later clicks on the third link down.
- A user runs a query and immediately clicks on the third link down.
- 10 seconds after clicking on a link, the user uses the browser's "Back" button, scrolls through the list, and clicks on another link.
- 15 seconds after that, the user comes back and clicks on the first link again.

Utility of Data

The results of query log analysis have many uses in evaluation and tuning:

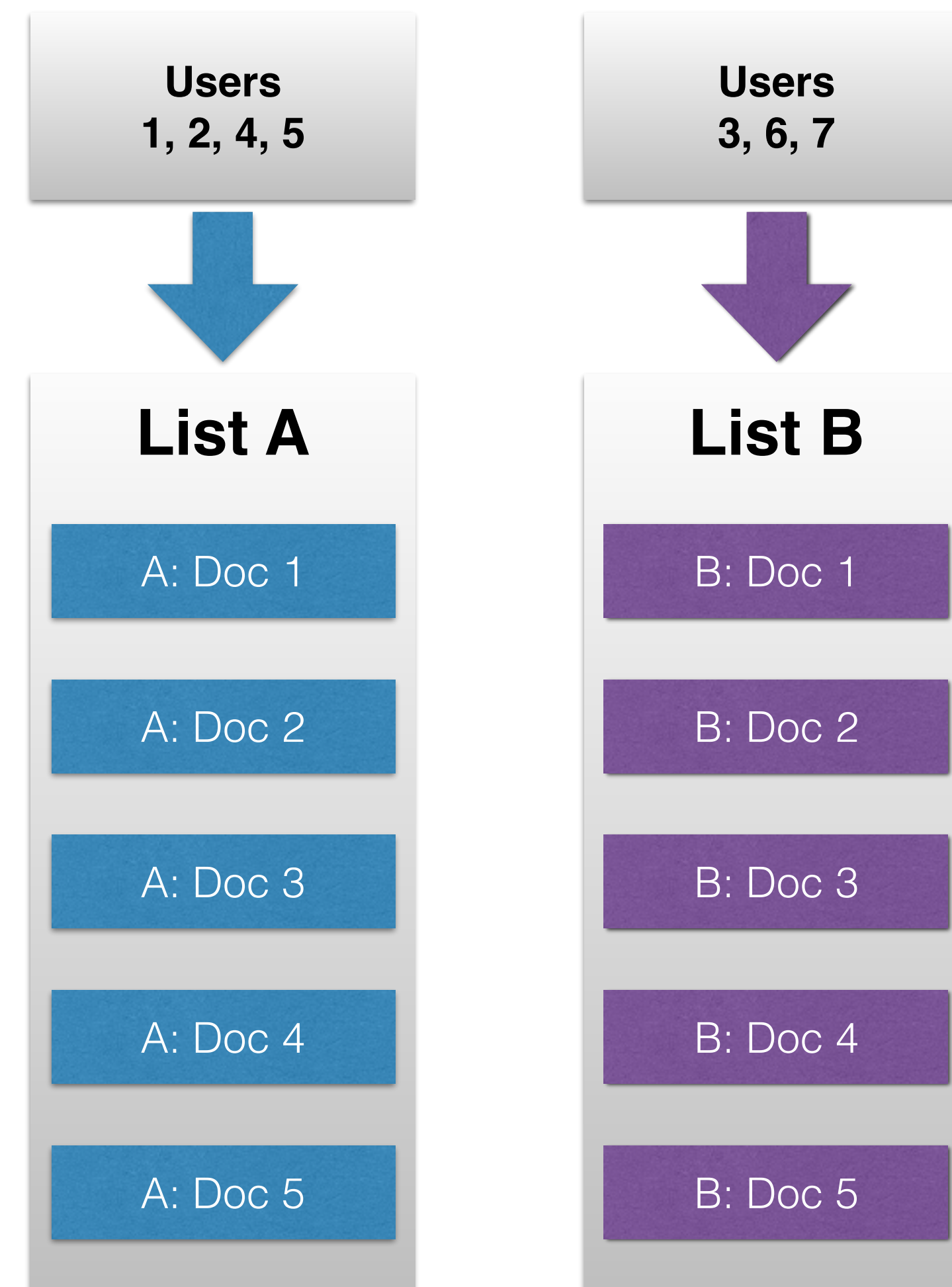
- Inferred relevance can produce precision estimates across tens of thousands of users.
- Similar queries point out different phrasings of the same information need, or similar phrasings for different information needs.
- Queries that tend to be repeated by the same or different users suggest caching strategies.
- If a user returns and repeats the same query, you can provide a better ranking based on their prior interaction.

A/B Testing

In addition to analyzing query logs, there are various ways the search engine results can be manipulated in order to compare systems.

In A/B testing, we show most users the normal system (system A) but show a small randomly-selected group of users a test system (system B).

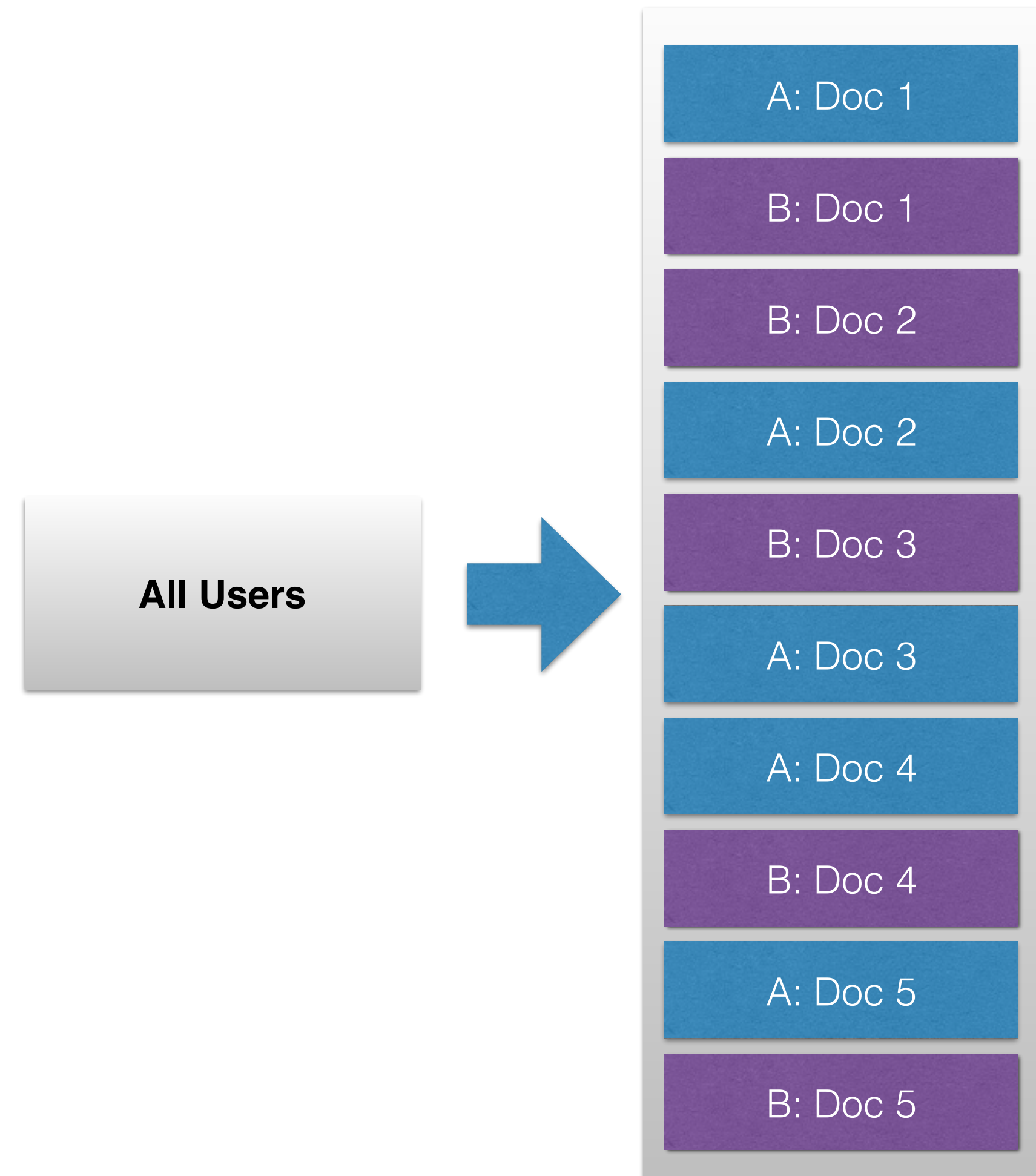
This is commonly used to test interface changes, ranking changes, etc.



Result Interleaving

Another approach is to randomly interleave the results from multiple systems' rankings, and measure which system's results get clicked more on average.

This makes it easier to determine which system a given user prefers, when the results of A/B testing are ambiguous.



Wrapping Up

Query log data provides very large numbers of users and queries and demonstrates real user behavior against real IR systems.

However, the data is superficial, in the sense that you can't ask the users what they're thinking, whether they're satisfied, why they changed their query, etc.

Next, we'll look at conducting user studies in a lab environment.

For many more details and citations to many relevant research articles, see the tutorial at <http://research.microsoft.com/en-us/um/people/sdumais/Logs-talk-HCIC-2010.pdf>

Explicit User Studies

Evaluation, session 10

The Gold Standard

Evaluating your system with users you can observe and talk to is considered the gold standard of IR evaluation.

- We can precisely determine or specify users' information needs.
- We can observe the actual behaviors of the people our system was designed for.
- We can ask them how difficult they think the interaction is, whether they were satisfied by the results, etc.

However, it's expensive, time-consuming, and requires careful experimental controls, so other evaluation methods often substitute.

History of User Studies

User studies have been conducted throughout the history of IR, back to Cyril Cleverdon's computer-free testing. In earlier studies, however, the "user" was an expert human searcher, not the end user with an information need.

In the 1980s, libraries started offering card catalog search tools (called "OPACs") directly to end users. Many experiments were done, often consisting of surveys about user demographics, information needs, and satisfaction levels.

Modern user studies often involve tailored search interfaces (to remove ads, search engine styling, etc.), eye-tracking, and detailed interaction logging. Users are sometimes asked to think aloud, or answer surveys before and after searching.

Selecting Users

Many studies recruit potential users from the closest available pool: grad students, friends, lab-mates, or even the researchers themselves. While convenient, this raises questions of the generalizability of the work.

One recent way people get a large pool of possibly-random subjects is through crowdsourcing sites, like Amazon Mechanical Turk. Is this group representative?

An ideal group would consist of a carefully-sampled selection of the *actual target users* of the IR system.

- For web search, this is a diverse sample of the general public.
- For legal, medical, or other search engines targeted at experts, this is a group of the experts themselves.

Types of Studies

- **Laboratory studies** occur in the lab, generally using custom measuring equipment and pre-determined search tasks.
- **Naturalistic studies** observe users interacting with a system in an uncontrolled way, wherever they naturally do so.
- **Wizard of Oz studies** test user interaction with a simulated or manipulated “ideal” system, generally without the users’ knowledge.

Types of Measurements

- **Observation** – The user's activity is recorded by software, a camera, and/or direct observation by the researcher.
- **Think Aloud** – Users are asked to speak their thought process out loud while interacting with the system.
- **Talk After** – Users interact with the system, and then the researcher plays back the recorded interaction and asks questions.
- **Self-Reporting** – Users discuss their thoughts during the experiment, either spontaneously or when prompted.
- **Logging** – Server-side, or via client-side tools such as browser plugins

Wrapping Up

Direct user studies are expensive and time-consuming, but frequently produce useful insights into IR system performance.

For many details on setting up a proper user study, see Diane Kelly's tutorial, [Methods for evaluating interactive information retrieval systems with users.](#)

Next, we'll examine some of the things we've learned from user studies.

What We've Learned from Users

Evaluation, session 11

Users vs. Batch Evaluation

Are we aiming for the right target? Many papers, and the TREC interactive track, have studied whether user experience matches batch evaluation results.

The statistical power of these papers is in question, but the answer seems to be:

- Batch evaluation really corresponds to better rankings and more user satisfaction.
- But better rankings don't necessarily lead to users finding more relevant content: users adapt to worse systems by running more queries, scanning poor results faster, etc.

TF-IDF baseline vs. Okapi ranking

| | Mean average precision | Precision @ 10 documents | Precision @ 50 documents |
|-------------------------|------------------------|--------------------------|--------------------------|
| Baseline system | 0.36 | 0.35 | 0.27 |
| Improved system | 0.53 | 0.55 | 0.36 |
| Change | +46% | +55% | +33% |
| p-value (paired t-test) | 0.02 | 0.03 | 0.14 |

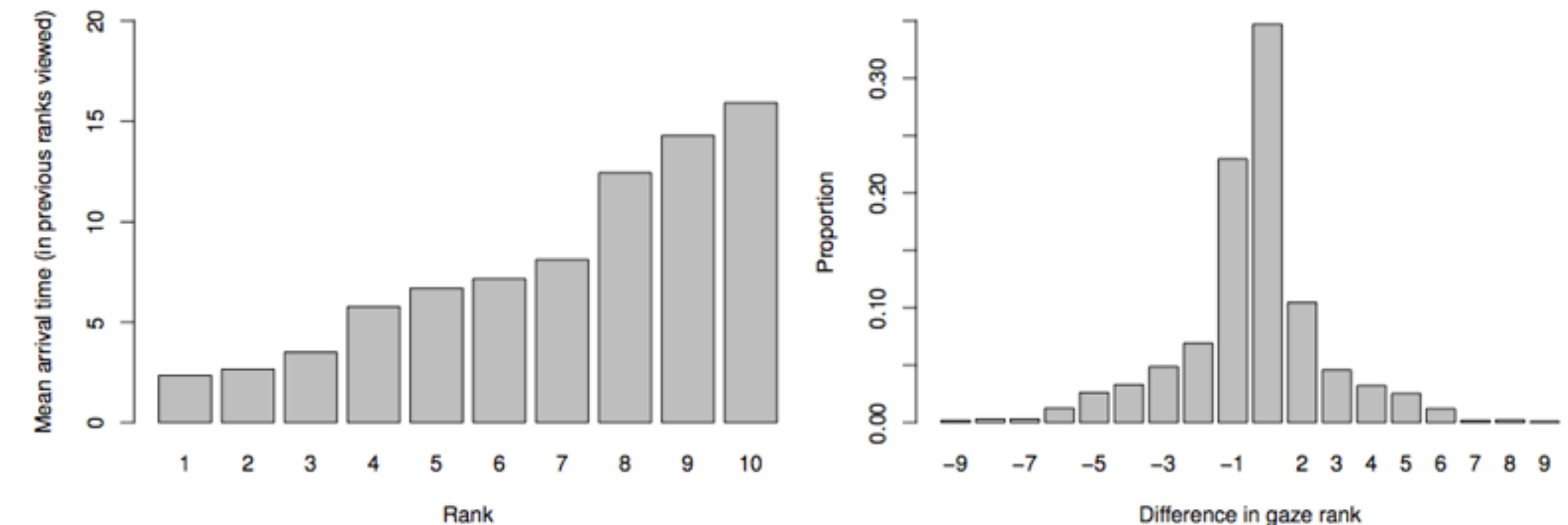
| | Queries per User | Documents Retrieved | |
|-------------------------|----------------------------|-------------------------------------|--|
| | Instance recall experiment | Instance recall experiment Relevant | Instance recall experiment Nonrelevant |
| Baseline system | 3.56 | 129.7 | 158.7 |
| Improved system | 2.98 | 131.8 | 103.0 |
| Change | -16% | +2% | -35% |
| p-value (paired t-test) | 0.16 | 0.93 | 0.01 |

Users vs. Metrics

Are we measuring in the right way? Do the user models implied by our batch evaluation metrics correspond to actual user behavior?

- Users scan in order overall, but with lots of smaller jumps forward and backward.
- Users *usually* just look at the top few documents, but *sometimes* look very deeply into the list. This depends on the individual, the query, the number of relevant documents they find, and...

User eye-tracking results



(a) Mean arrival time at each result rank

(b) Jumps between consecutive fixations

Factors affecting prob. of continuing

| Factor | Effect |
|---|------------|
| (intercept) | 11.70 |
| User | 0.11–10.95 |
| Proportion of T collected | 0.34 |
| Proportion of docs viewed that are relevant | 0.50 |
| Gaze sequence | 0.97 |
| Rank i | 1.06 |
| Query count, in task | 1.10 |

Table 4: Factors in a fitted model of DC. Users become more persistent (have higher $C(i)$) as they issue more queries and look further down each result page; and become less persistent as they accumulate relevant documents.

Users vs. Relevance

Batch evaluation treats relevance as a binary or linear concept. Is this really true?

- Users respond to many attributes in order to determine relevance. Document attributes interact with user attributes in complex ways.
- Different users weight these factors differently, and the weights may change over the course of a session.
- Users' ability to perceive relevance improves over a session, and their judgements become more stringent.

Factors Affecting Relevance

- *Content*: Topic, quality, depth, scope, currency, treatment, clarity.
- *Object*: Characteristics of information objects (e.g., type, organization, representation, format, availability, accessibility, costs).
- *Validity*: Accuracy of information provided, authority, trustworthiness of sources, verifiability.
- *Use or situational match*: Appropriateness to situation, or tasks, usability, urgency; value in use.
- *Cognitive match*: Understanding, novelty, effort.
- *Affective match*: Emotional responses to information, fun, frustration, uncertainty.
- *Belief match*: Credence given to information, acceptance as to truth, reality, confidence.

Source: Tefko Saracevic. Relevance: A review of the literature and a framework for thinking on the notion in information science. Part III: Behavior and effects of relevance. *J. Am. Soc. Inf. Sci. Technol.* 58, 13 (November 2007)

Experts vs. General Users

How do experts search differently, and how can we improve rankings for experts?

- Experts use different vocabulary and longer queries, so they can be identified with reasonable accuracy.
- Experts visit different web sites, which could be favored for their searches.
- The search engine could play a role in training non-experts, by moving them from tutorial sites to more advanced content.

Finding Thousands of Experts in Log Data

1. Viewed ≥ 100 pages over three months
2. 1% or more domain-related pages
3. Visited costly expert sites (such as dl.acm.org)

Preferred Domain Differences By Expertise

| Domain | Expert | Non-expert |
|--------|---|---|
| CS | acm.org ieee.org nist.gov sigmod.org columbia.edu cornell.edu cmu.edu msdn.com computer.org codeplex.com | microsoft.com download.com msdn.com codeproject.com nist.gov sun.com codeplex.com dell.com w3schools.com adobe.com |

Query Vocabulary Change By Expertise

| Domain | Experts | | | Non-expert | | |
|-----------------|---------|-------|-------|------------|-------|-------|
| | ↓ | ↔ | ↑ | ↓ | ↔ | ↑ |
| <i>Medicine</i> | 9.8% | 74.9% | 15.3% | 8.3% | 43.1% | 48.6% |
| <i>Finance</i> | 10.1% | 75.8% | 14.3% | 9.9% | 52.0% | 38.1% |
| <i>Legal</i> | 13.2% | 73.2% | 13.6% | 15.2% | 54.1% | 30.7% |
| CS | 9.4% | 72.1% | 18.5% | 11.1% | 51.7% | 37.2% |

Source: Ryen W. White, Susan T. Dumais, and Jaime Teevan. Characterizing the influence of domain expertise on web search behavior. WSDM 2009.

Social vs. IR Searching

Many recent studies have investigated the relative merit of search engines and social searching (e.g. asking your Facebook friends).

One typical study asked 8 users to try to discover answers to several “Google hard” questions, either using only traditional search engines or only social connections (via online tools, “call a friend,” etc.).

- Search engines returned more high-quality information in less time.
- But social connections helped develop better questions, and helped synthesize material (when they took the question seriously), so led to better understanding.

“Google hard” Queries

55 MPH: If we lowered the US national speed limit to 55 miles per hour (MPH) (89 km/h), how many fewer barrels of oil would the US consume every year?

Pyrolysis: What role does pyrolytic oil (or pyrolysis) play in the debate over carbon emissions?

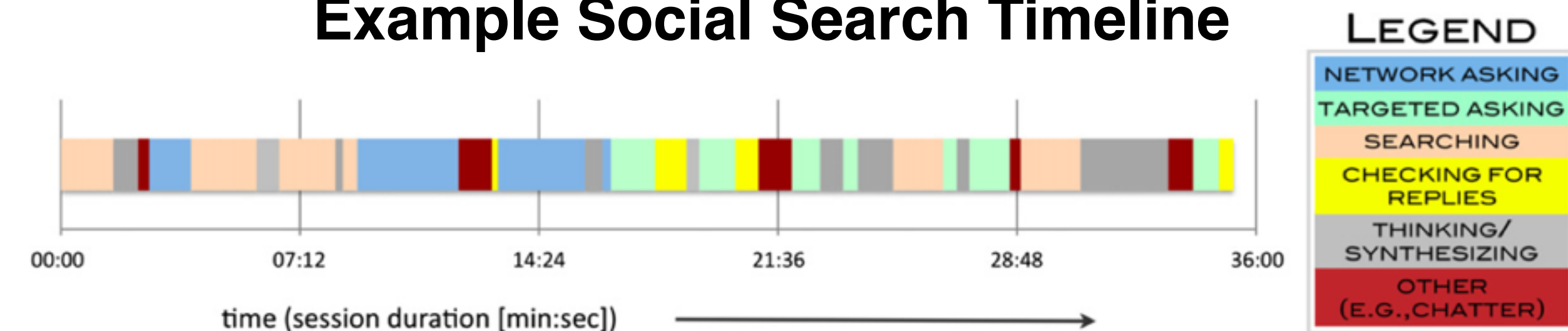
Social Tactics Used

Targeted Asking: Asking specific friends for help via e-mail, phone, IM, etc.

Network Asking: Posting a question on a social tool such as Facebook, Twitter, or a question-answer site.

Social Search: Looking for questions and answers posted to social tools, such as question-answer sites.









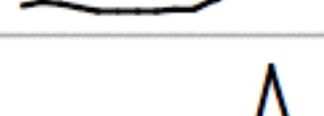

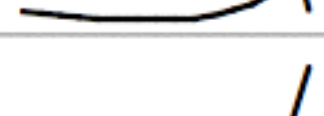

Example Social Search Timeline



Revisited Pages

Studies indicate that 50-80% of web traffic involves revisiting pages the user has already visited. What can we learn about the user's intent from the delays between visits?

- There are clear trends in visit delays based on content type and the user's intent, with high variance between users.
- This can inform design of web browsers (e.g. history, bookmarks display) and search engines (e.g. document weighting based on individual revisit patterns).

| Cluster Group | Name | Shape | Description |
|---|------|---|---|
| Fast Revisits ($<$ hour) 23611 pages | F1 |  | Pornography & Spam, Hub & Spoke, Shopping & Reference Web sites, Auto refresh, Fast monitoring |
| | F2 |  | |
| | F3 |  | |
| | F4 |  | |
| | F5 |  | |
| Medium (hour to day) 9421 pages | M1 |  | Popular homepages, Communication, .edu domain, Browser homepages |
| | M2 |  | |
| Slow Revisits ($>$ day) 18422 pages | S1 |  | Entry pages, Weekend activity, Search engines used for revisitation, Child-oriented content, Software updates |
| | S2 |  | |
| | S3 |  | |
| | S4 |  | |
| Hybrid 3334 pages | H1 |  | Popular but infrequently used, Entertainment & Hobbies, Combined Fast & Slow |

Wrapping Up

The papers shown here are just the tip of the iceberg in terms of meaningful insights drawn from user studies.

Interesting future directions:

- More nuanced relevance judgements, and test collections and batch evaluations that reflect the complex, dynamic user reality.
- Better integration of web search into browsers, social sites, and other tools, with real use patterns informing design decisions.
- More customized experiences taking into account user type, information need complexity, prior individual usage patterns, etc.

Module Wrap Up

Evaluation, session 12

Why do we evaluate?

It's often tempting, when you have a great idea for a new product or a better solution to a problem, to just implement it and use it. Why bother going through a formal evaluation process?

Evaluation is testing for scientific claims. Just as you shouldn't release a program without some sort of formal verification that it's correct, it's unwise to change your search engine or update your product recommendation service without measuring how it compares to the old system.

How do we evaluate?

Choosing the right approach to evaluation depends on your budget and other resources, what you want to measure, your tolerance for errors, and other factors.

- Explicit user studies allow you to run carefully controlled experiments, but are expensive and time-consuming.
- Implicit user studies can collect much more data, but often require access to the resources of a large company.
- Batch evaluation allows a rapid development cycle based on a simplified user model, but generally requires the use of an adequate test collection.
- Proper statistical tests are required in any case to determine whether your conclusions are justified.

Coming Up...

Next, we'll learn more about how to apply machine learning techniques to retrieval.