CS3245

Information Retrieval

9

Lecture 9: IR Evaluation

Last Time

The VSM Reloaded

… optimized for your pleasure!

Heuristics to make search faster:

1. Don't compute what you don't need
2. Approximate things that take a lot of time

Today: Evaluation

- How do we know if our results are any good?
  - Evaluating a search engine
    - Benchmarks
    - Precision and Recall; Composite measures
- A/B Testing (not in textbook)

EVALUATING SEARCH ENGINES

The measure of a IR engine

- How fast does it index?
  - Number of documents/hour
  - (Average document size)
- How fast does it search?
  - Latency as a function of index size
- Expressiveness of query language?
  - Ability to express complex information needs
  - Speed on complex queries
- User Interface?
- Is it free?

Measures for a search engine

- All of the preceding criteria are *measurable*: we can quantify speed/size
  - we can make expressiveness precise
- The key measure: user happiness
  - Speed of response/size of index are factors
  - But blindingly fast, useless answers won't make a user happy
- Need a way of quantifying user happiness

Measuring user happiness

- Question: who is the user we are trying to make happy?
  - Depends on the setting
- <u>Web engine</u>:
  - User finds what they want and return to the engine
    - Can measure rate of returning users
  - User completes their task – search as a means, not end
- <u>eCommerce site</u>: user finds what they want and buy
  - Is it the end-user, or the eCommerce site, whose happiness we measure?
  - Measure time to purchase, or fraction of searchers who become buyers?

Measuring user happiness

- <u>Enterprise</u> (company/govt/academic): Care about "user productivity"
  - How much time do my users save when looking for information?
  - Many other criteria having to do with breadth of access, secure access, etc.

Happiness: elusive to measure

- Most common proxy: *relevance* of search results
- But how do you measure relevance?

We'll examine one method and the issues around it

- Relevance measurement requires **3** elements:
  1. A set document collection
  2. A set suite of queries
  3. A usually binary assessment of either Relevant or Non-relevant for each query and each document
     - Some work on graded relevance, but not the standard

Evaluating an IR system

- Note: the **information need** is translated into a **query**
- Relevance is assessed relative to the **information need** *not* the **query**

- E.g., Information need: *I'm looking for information on whether drinking red wine is more effective at reducing your risk of heart attacks than white wine.*
- Query: ***wine red white heart attack effective***

i.e., we evaluate whether the doc addresses the information need, not whether it has these words

Slide courtesy Google Inc.

Why it's important: Example Think-Aloud Session

00:12 [actor most oscars]

1:15 [actor most oscars Academy]

00:10 So this is celebrity with most Oscars…
00:11 Actor… ah… most…
00:13 I'm just going to try that…most Oscars… don't know…
00:19 (reading) "News results for 'actors most Oscars' … " huh..
00:25 Oh, then that would be currently "Brokeback"… "prior voices"… "truth in Oscar's relevance"…
00:32 …now I know…
00:35 … you get a lot of weird things..hold on…
00:38 "Are Filipinos ready for gay flicks?"
00:40 How does that have to do with what I just….did...?
00:43 Ummm…
00:44 So that's where you can get surprised… you're like, where is this… how does this relate…umm…
00:45 Bond…I would think…
00:46 So I don't know, it's interesting…
01:08 **Dan**: Did you realize you were in the News section?
01:09 Oh, no I didn't. How did I get that? . . .
01:10 Oooh… no I didn't.

6

Unranked retrieval evaluation: Precision and Recall

- **Precision**: fraction of retrieved docs that are relevant = P(relevant|retrieved)
- **Recall**: fraction of relevant docs that are retrieved = P(retrieved|relevant)

	Relevant	Non-relevant
Retrieved	true positive	false positive
Not Retrieved	false negative	true negative

Precision $P = tp/(tp + fp)$

Recall $R = tp/(tp + fn)$

Should we use accuracy for evaluation instead?

- Given a query, a Boolean engine classifies each doc as Relevant or Non-Relevant
- The **accuracy** of an engine: the fraction of these classifications that are correct
  - (tp + tn) / (tp + fp + fn + tn)
- **Accuracy** is a commonly used evaluation measure in classification (e.g. HW1)

Quick Question: Why is this not a very useful evaluation measure in IR?

Precision/Recall

- You can get high recall (but low precision) by retrieving all docs for all queries!
- Recall is a non-decreasing function of the number of docs retrieved

- In a good system, precision decreases as either the number of docs retrieved or recall increases
  - This is not a theorem, but a result with strong empirical confirmation

Difficulties in using precision/recall

- Should average over large document collection/ query ensembles
- Need human relevance assessments
  - But people are subjective; they aren't reliable assessors
- Assessments have to be binary
  - Can we give graded assessments?
- Heavily skewed by collection/queries pairing
  - Results may not translate from one collection to another

We'll return to this point later

A combined measure: *F*

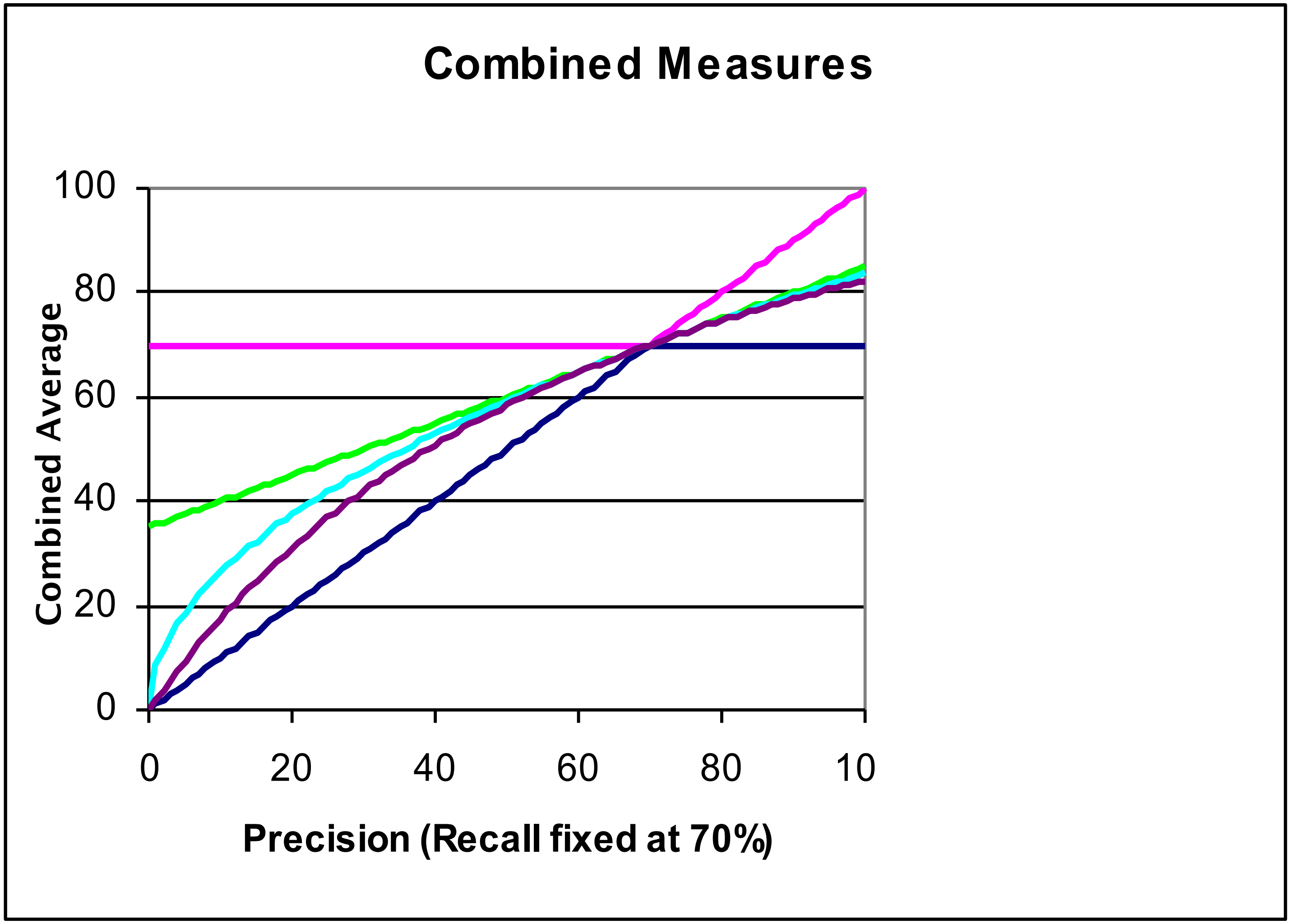
- Combined measure that assesses precision/recall tradeoff is **F measure** (weighted harmonic mean):

$$F = \frac{1}{\alpha \frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

- People usually use balanced F_1 measure
  - i.e., with $\beta = 1$ or $\alpha = ½$
- Harmonic mean is a conservative average

Blanks on slides, you may want to fill in

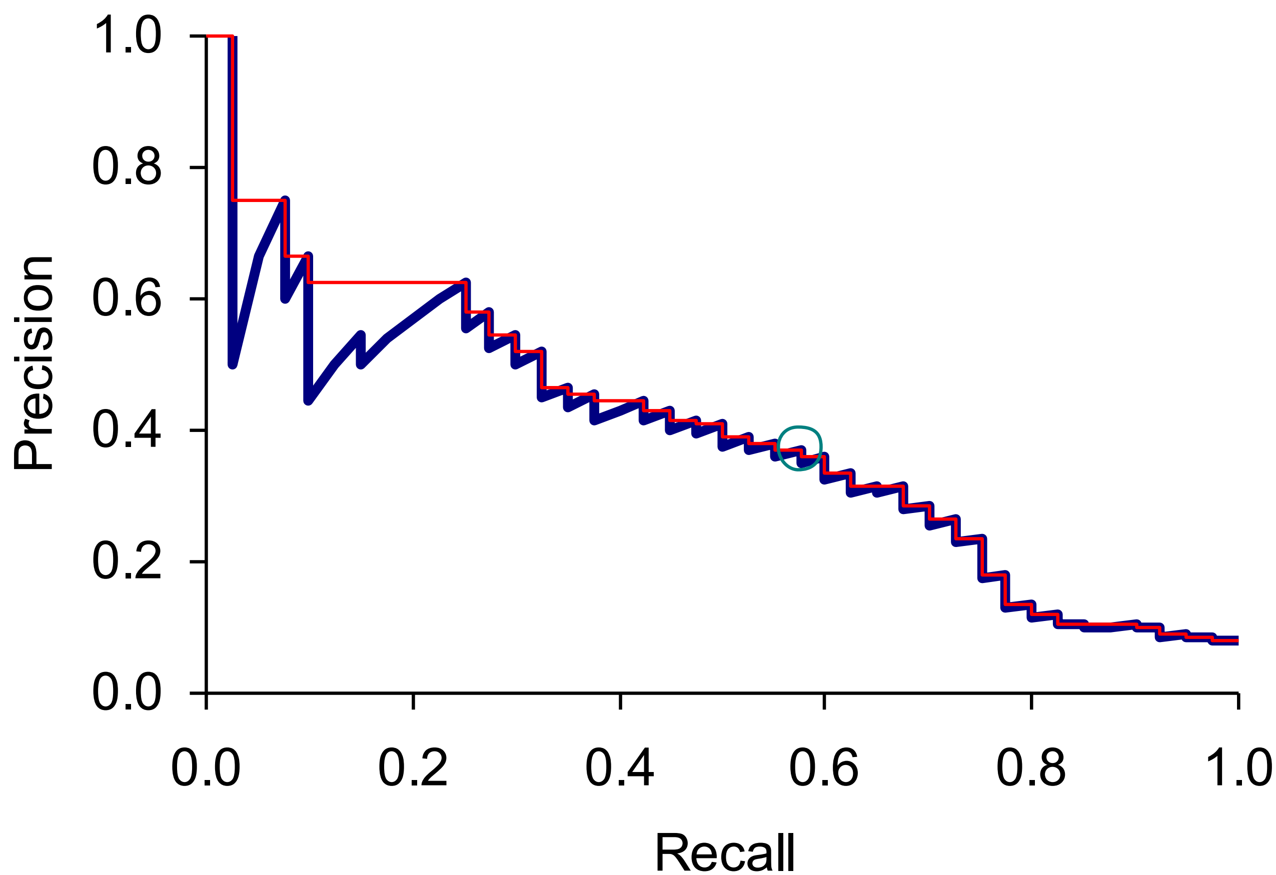
F_1 and other averages



Evaluating ranked results

- Evaluation of ranked results:
  - The system can return any number of results
  - By taking various numbers of the top returned documents (levels of recall), we can produce a *precision-recall curve*

A precision-recall curve



Averaging over queries

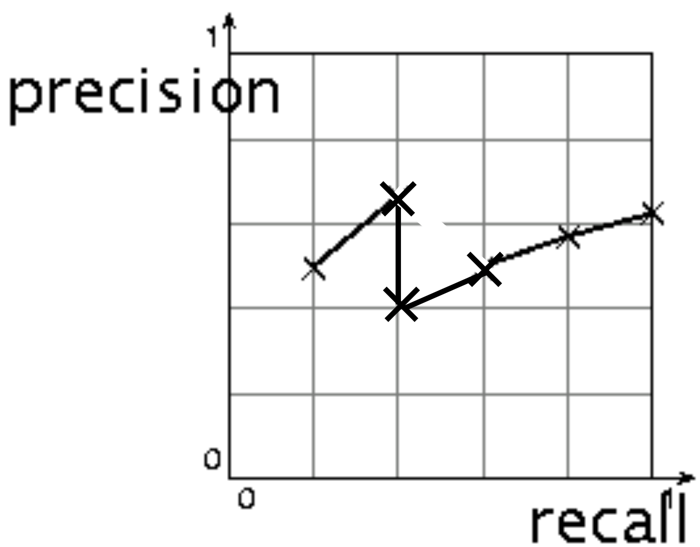
- A precision-recall graph for one query isn't a very sensible thing to look at
- Instead, average performance over a query collection.

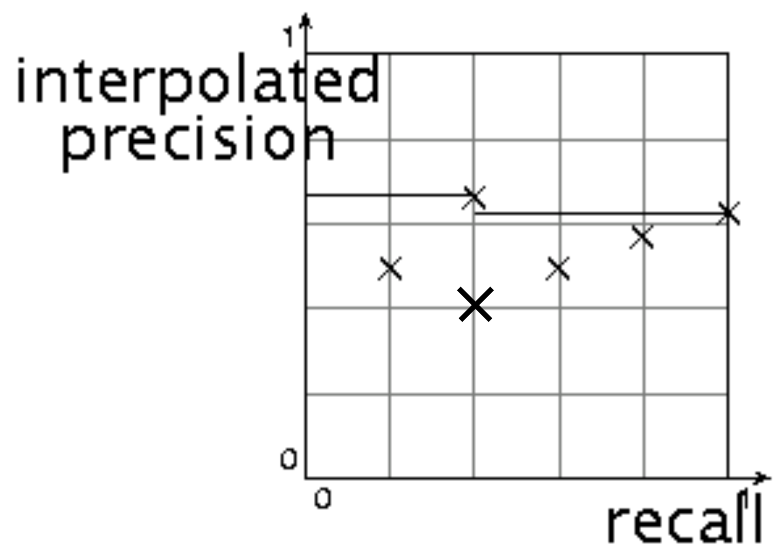
But there's a technical issue:

- Precision-recall calculations place some points on the graph
- How do you determine a value (interpolate) between the points?

Interpolated precision

- Idea: If locally precision increases with increasing recall, then you should get to count that…
- So you take the max of precisions to the right of the value





Evaluation

- Graphs are good, but often we want a summary measure!
  - Precision at fixed retrieval level
    - Precision-at-*k*: Precision of top *k* results
    - Perhaps appropriate for most of web search: all people want are good matches on the first one or two results pages
    - But: averages badly and has an arbitrary parameter of *k*
  - 11-point interpolated average precision
    - The standard measure in the early TREC competitions: you take the precision at 11 levels of recall varying from 0 to 1 by tenths of the documents, using interpolation (the value for 0 is always interpolated!), and average them
    - Evaluates performance at all recall levels

Yet more evaluation measures…

- Mean average precision (MAP)
  - Average of the precision value obtained for the top *k* documents, each time a relevant doc is retrieved
  - Avoids interpolation, use of fixed recall levels
  - MAP for query collection is arithmetic ave.
    - Macro-averaging: each query counts equally

Used in HW#4

- R-precision
  - If have known (though perhaps incomplete) set of relevant documents of size *Rel,* then calculate precision of top *Rel* docs returned
  - Perfect system could score 1.0.

Variance

- For a test collection, it is usual that a system does poorly on some information needs (e.g., MAP = 0.1) and excellent on others (e.g., MAP = 0.7)
- Indeed, it is usually the case that the variance in performance of the same system across queries is much greater than the variance of different systems on the same query.

- That is, there are easy information needs and hard ones!

CREATING TEST COLLECTIONS FOR EVALUATION

Test Collections

TABLE 4.3 Common Test Corpora

Collection	*NDocs*	*NQrys*	*Size (MB)*	*Term/Doc*	*Q-D RelAss*
ADI	82	35			
AIT	2109	14	2	400	>10,000
CACM	3204	64	2	24.5	
CISI	1460	112	2	46.5	
Cranfield	1400	225	2	53.1	
LISA	5872	35	3		
Medline	1033	30	1		
NPL	11,429	93	3		
OSHMED	34,8566	106	400	250	16,140
Reuters	21,578	672	28	131	
TREC	740,000	200	2000	89-3543	» 100,000

Scientific papers
Scientific papers
Medical
Medical
News
News

From document collections to test collections

- Still need the other **2** things
  - Test queries
  - Relevance assessments
- Test queries
  - Must be relevant to docs available
  - Best designed by domain experts
  - Random query terms generally not a good idea
- Relevance assessments
  - Human judges, time-consuming
  - Are human panels perfect?

Kappa measure for inter-judge (dis)agreement

- Kappa measure
  - Agreement measure among judges
  - Designed for categorical judgments
  - Corrects for chance agreement

- Kappa = [P(A) – P(E)] / [1 – P(E)]
- P(A) – proportion of time judges agree
- P(E) – what agreement would be by chance

- Gives 0 for chance agreement, 1 for total agreement.

Kappa Measure: Example

# of docs matching judgment type	Judge 1	Judge 2
300	Relevant	Relevant
70	Non-relevant	Non-relevant
20	Relevant	Non-relevant
10	Non-relevant	Relevant

What is P(A)?
How about P(E)?

Kappa Example

$P(A) = 370/400 = 0.925$

$P(\text{nonrelevant}) = (10+20+70+70)/800 = 0.2125$

$P(\text{relevant}) = (10+20+300+300)/800 = 0.7878$

$P(E) = 0.2125^2 + 0.7878^2 = 0.665$

$\text{Kappa} = (0.925 – 0.665)/(1-0.665) = 0.776$

- Kappa > 0.8 ➔ good agreement
- 0.67 < Kappa < 0.8 ➔ "tentative conclusions"

- Depends on purpose of study
- For >2 judges: average pairwise kappas (or ANOVA)

TREC

- TREC's Ad Hoc task from first 8 TRECs was the standard IR task
  - 50 detailed information needs a year
  - Human evaluation of **pooled** results returned
  - More recently other related things: Web, Hard, QA, interactive track
- A query from TREC 5 (1996)

```
<top>
<num>225</num>
<desc>What is the main function of the Federal
  Emergency Management Agency (FEMA) and the
  funding level provided to meet emergencies?
  Also, what resources are available to FEMA such
  as people, equipment, facilities?</desc>
</top>
```

Interjudge Agreement: TREC 3

information need	number of docs judged	disagreements	NR	R
51	211	6	4	2
62	400	157	149	8
67	400	68	37	31
95	400	110	108	2
127	400	106	12	94

Shows that there are queries that are easier than others

Critique of pure relevance

- Relevance versus **Marginal Relevance**
  - A document can be redundant even if it is highly relevant
  - Duplicates
  - The same information from different sources
  - Marginal relevance is a better measure of utility for the user.
- Using facts/entities as evaluation units more directly measures true relevance.
- But then it's harder to create the evaluation set

Can we avoid human judgment?

Unfortunately, no.

- Makes experimental work hard
  - Especially on a large scale
  - Can be tedious, expensive to calculate
  - Recently, use crowdsourcing methods to collect data
- In some very specific settings, can use proxies
  - E.g.: for approximate vector space retrieval, we can compare the cosine distance closeness of the closest docs to those found by an approximate retrieval algorithm
- But once we have test collections, we can reuse them

Evaluation at large search engines

- Search engines have test collections of queries and hand-ranked results
- Recall is difficult to measure on the web
- Search engines often use precision at top k (e.g., k = 10)
- . . . or measures that reward you more for getting rank 1 right than for getting rank 10 right.
  - NDCG (Normalized Cumulative Discounted Gain)
  - MRR (Mean Reciprocal Rank)
- Search engines also use non-relevance-based measures.
  - Clickthrough on first result
    - Not very reliable if you look at a single clickthrough … but pretty reliable in the aggregate.
  - Studies of user behavior in the lab
  - A/B testing

A/B testing

Purpose: Test a single innovation

Prerequisite: You have a large search engine up and running.

- Have most users use old system
- Divert a small proportion of traffic (e.g., 1%) to the new system that includes the innovation
- Evaluate with an "automatic" overall evaluation criterion (OEC) like clickthrough on first result
- Now we can directly see if the innovation does improve user happiness.
- Probably the evaluation methodology that large search engines trust most
- In principle less powerful than doing a multivariate regression analysis, but easier to understand

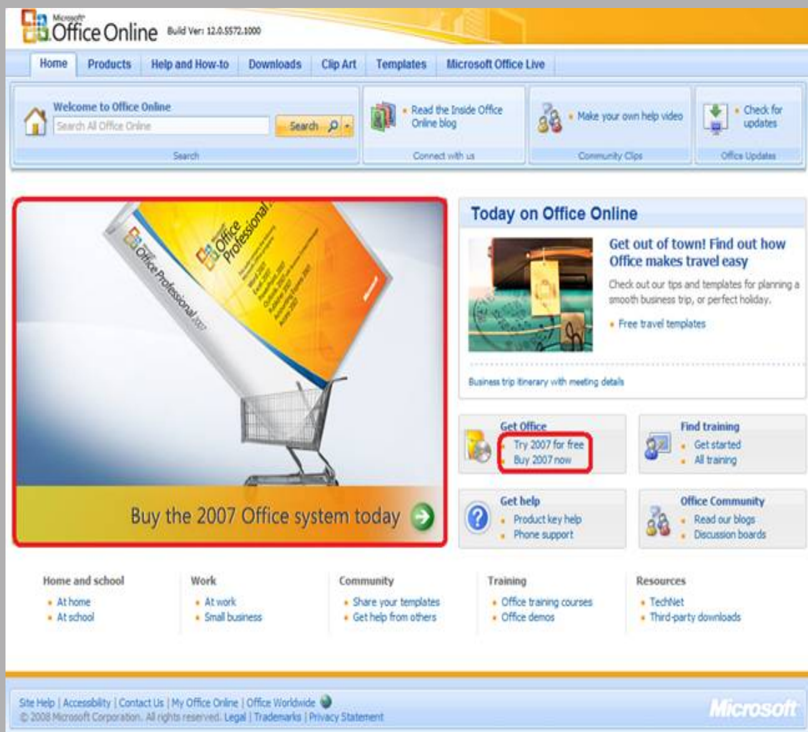
Slide courtesy Microsoft Inc.

Office Online

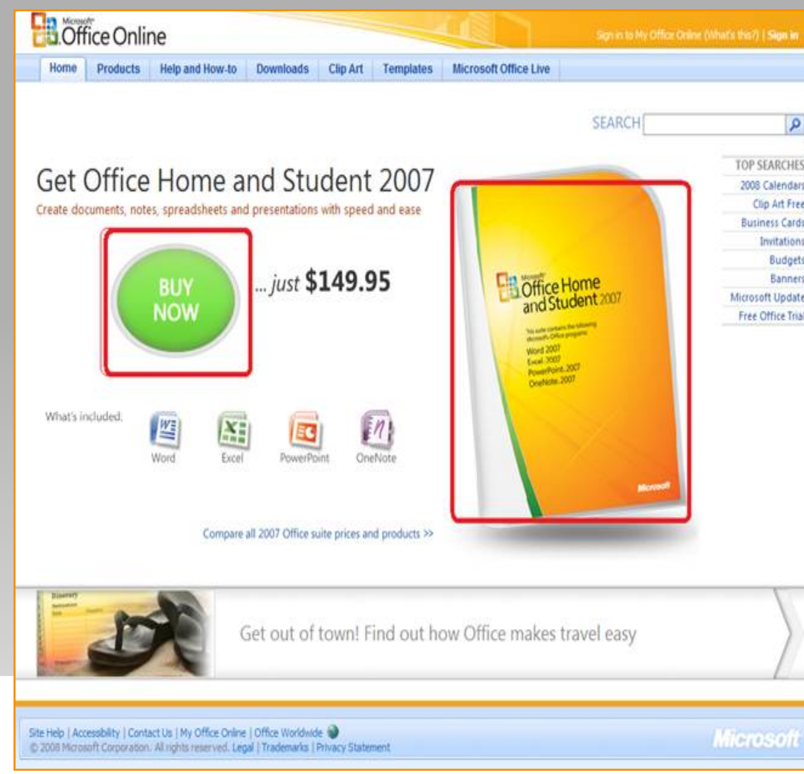
Test new design for Office Online homepage

OEC: Clicks on revenue generating links (red below)

A



B



Is A better, B better, or are they about the same?

Slide courtesy Microsoft Inc.

Office Online

- B was 64% worse
- The Office Online team wrote

A/B testing is a fundamental and critical Web services… consistent use of A/B testing could save the company millions of dollars

Slide courtesy Microsoft Inc.

The HiPPO

The less data, the stronger the opinions

- Our opinions are often wrong – get the data
- HiPPO stands for the Highest Paid Person's Opinion
- Hippos kill more humans than any other (non-human) mammal (really)
- Don't let HiPPOs in your org kill innovative ideas. ExPeriment!
- We give out these toy HiPPOs at Microsoft

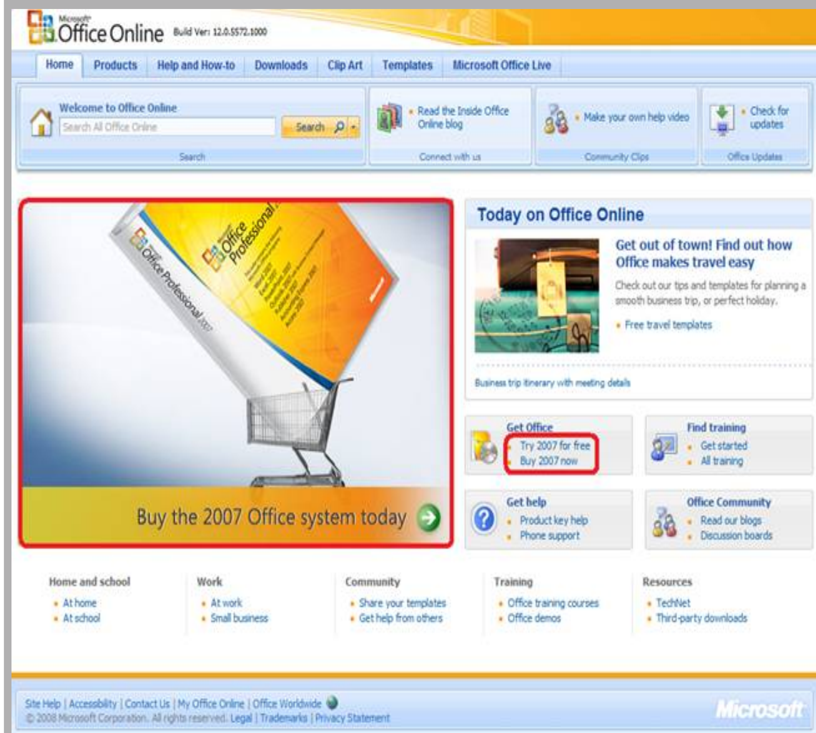
Slide courtesy Microsoft Inc.

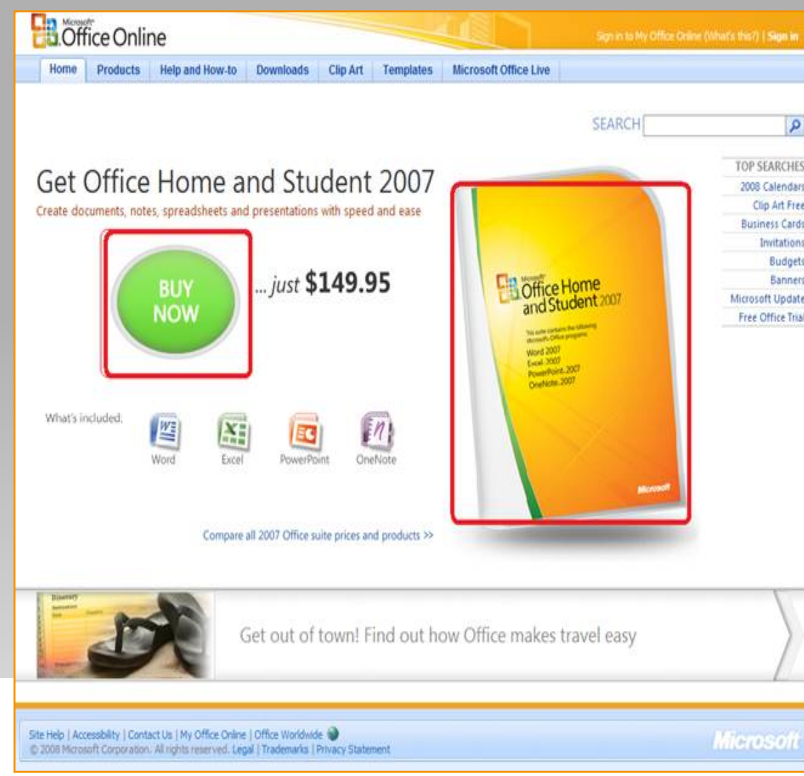
Pitfall 1: Wrong Success Metric

Remember this example?

OEC: Clicks on revenue generating links (red below)

A





B

Slide courtesy Microsoft Inc.

Pitfall: Wrong Overall Evaluation Criterion (OEC)

- B had a drop in the OEC of 64%
- Were sales correspondingly less also?
- No. The experiment is valid if the conversion from a click to purchase is similar
- The price was shown only in B, sending more qualified purchasers to the pipeline
- Lesson: measure what you really need to measure, even if it's difficult!

Summary: Evaluation

Different schemes for lab versus in-the-wild testing

- Benchmark testing
- A/B testing

Resources:

- IIR 8, MIR Chapter 3, MG 4.5