EVALUATION IN IR

Introduction to Information Retrieval

CS 150

Donald J. Patterson

Content adapted from Hinrich Schütze
http://www.informationretrieval.org

Outline

- Intro to Evaluation
- Standard Test Collections
- Evaluation of Unranked Retrieval
- Evaluation of Ranked Retrieval
- Assessing relevance
- Broader perspectives
- Result Snippets

Intro to Evaluation

- There are many implementation decisions to be made in an IR system
  - Crawler
    - Depth-first or breadth-first?
  - Indexer
    - Use zones?
    - Which zones?
    - Use stemming?
    - Use multi-word phrases? Which ones?

Intro to Evaluation

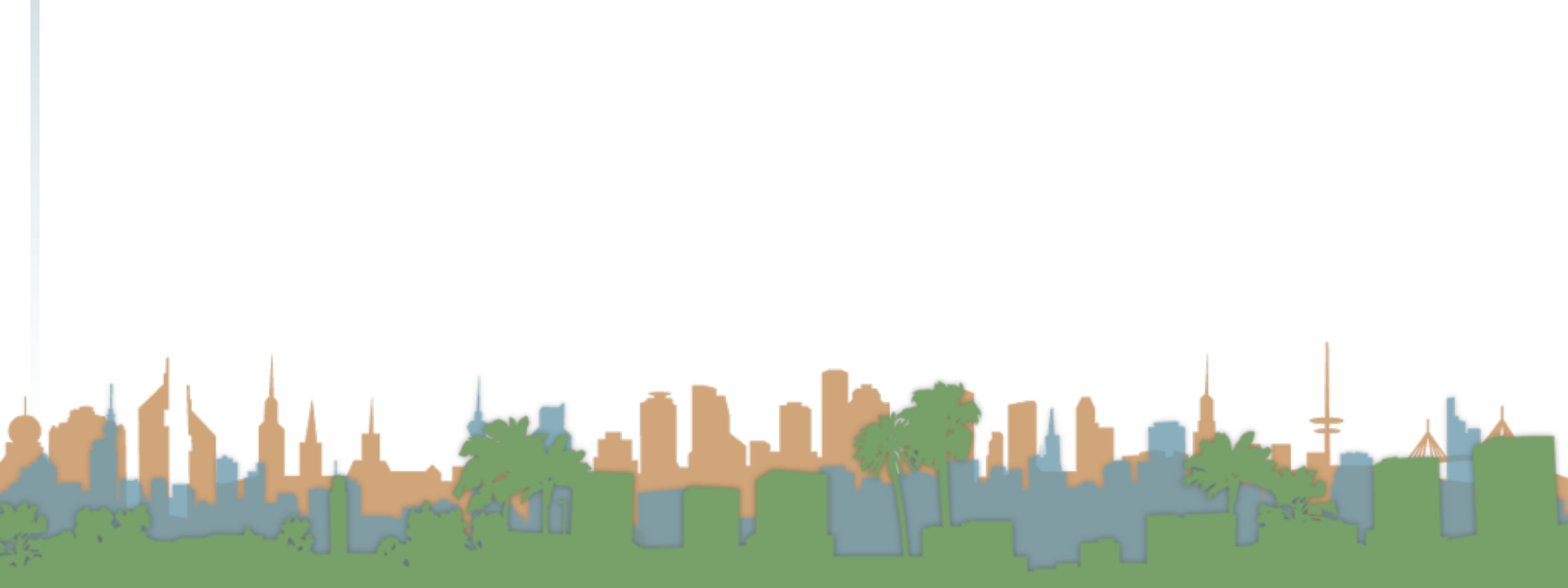
- There are many implementation decisions to be made in an IR system
  - Query
    - Ranked Results?
    - PageRank?
  - Which formula do we use in the TF-IDF Matrix?
  - Should we use Latent Semantic Indexing?
    - How many dimensions should we reduce?

Intro to Evaluation

- There are many implementation decisions to be made in an IR system
  - Results
    - How many do we show?
    - Do we show summaries?
    - Do we group them into categories?
    - Do we personalize the rankings?
    - Do we display results graphically?

Intro to Evaluation

- How can we evaluate whether we made good decisions or not?
  - Measure them



Measures for a search engine

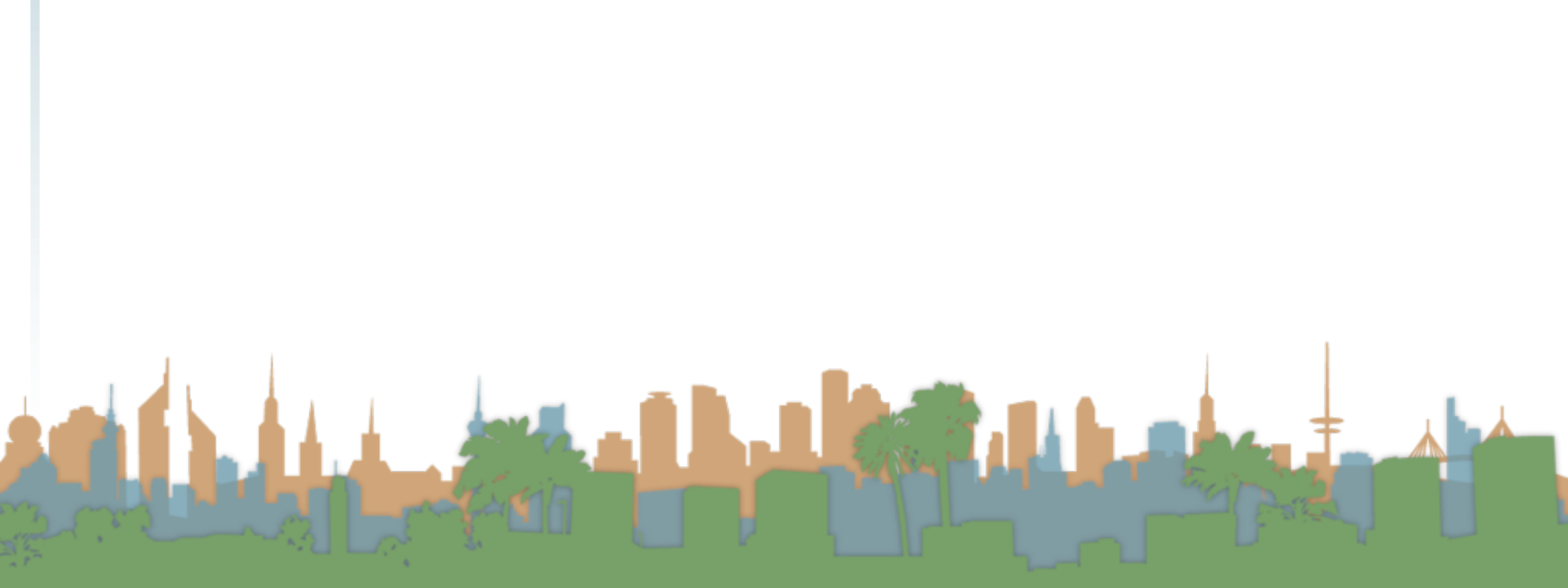
- How fast does it index?
  - Number of documents per hour
  - Average document size
- How fast does it search
  - Latency as a function of index size
- Expressiveness of query language
  - Ability to express complex information needs
  - Speed on complex queries

Measures for a search engine

- We can measure all of these things:
  - We can quantify size and speed
  - We can make this precise
- What about user happiness?
  - What is this?
  - Speed of response/size of index are factors
  - But fast, useless answers won't make a user happy
- Need to quantify user happiness also.

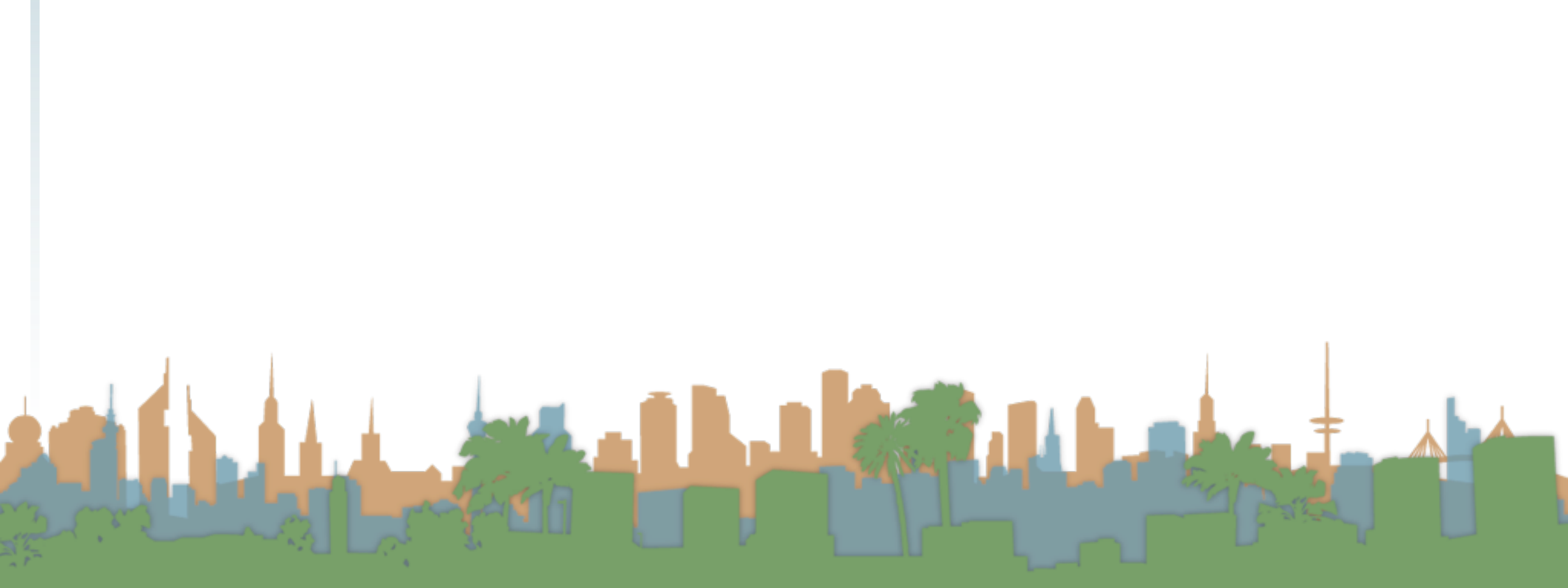
Measuring user happiness

- Issue: Who is the user we are trying to make happy?
  - It depends.



Measuring stakeholder happiness

- Issue: Who is the user we are trying to make happy?
  - Search engine:
    - The user finds what they want.
    - Measure whether or not they come back.



Measuring stakeholder happiness

- Issue: Who is the user we are trying to make happy?
  - local search on an eCommerce Site
    - User finds what they want
    - Are we interested in the happiness of the webmaster?
    - Are we interested in the happiness of the customer?
    - Measure the $$ of sales per user
    - Measure number of transactions per user
    - Measure time to purchase
    - Measure conversion rate (lookers -> buyers)

Measuring stakeholder happiness

- Issue: Who is the user we are trying to make happy?
  - Search on an Enterprise site
    - Are the users "productive"?
    - Measure time savings when using site
    - Measure "things accomplished"
      - careful about confounding factors
  - Measure how much a user utilizes the site's features

Measuring stakeholder happiness

- Can we measure happiness?
- Do we want to measure happiness?
- What are some proxies for happiness?
  - Relevance of search results
- How do we measure relevance?

Measuring Relevance Instead

- What do we need to measure relevance?
  - A document collection, a test corpus
  - A set of queries, benchmark queries
  - A set of answers, a gold standard
    - i.e., Document, d, {is, is not} relevant to query q
    - Alternatives to binary exist, but atypical
- Cross-validation methodology
  - Parameter tuning

Information need

- Remember the user has an information need
  - not a query
- Relevance is assessed in relation to the information need, not the query
  - e.g., I am looking for information on whether drinking red wine is more effective than eating chocolate at reducing risk of heart attacks
- Query: red wine heart attack effective chocolate risk
- Does the document address the need, not the query

Relevance benchmarks

- TREC - National Institute of Standards and Testing (NIST) has run a large IR test bed for many years
- Reuters and other benchmark document collections
- Retrieval tasks which are specified
  - sometimes as queries
- Human experts mark, for each query and for each document
  - Relevant or Irrelevant

MICROSOFT LEARNING TO RANK DATASETS

Microsoft Learning to Rank Datasets

We release two large scale datasets for research on learning to rank: MSLR-WEB30k with more than 30,000 queries and a random sampling of it MSLR-WEB10K with 10,000 queries.

- Introduction
- Download
- Feature List
- Related Links

Dataset Descriptions

The datasets are machine learning data, in which queries and urls are represented by IDs. The datasets consist of feature vectors extracted from query-url pairs along with relevance judgment labels:

(1) The relevance judgments are obtained from a retired labeling set of a commercial web search engine (Microsoft Bing), which take 5 values from 0 (irrelevant) to 4 (perfectly relevant).

(2) The features are basically extracted by us, and are those widely used in the research community.

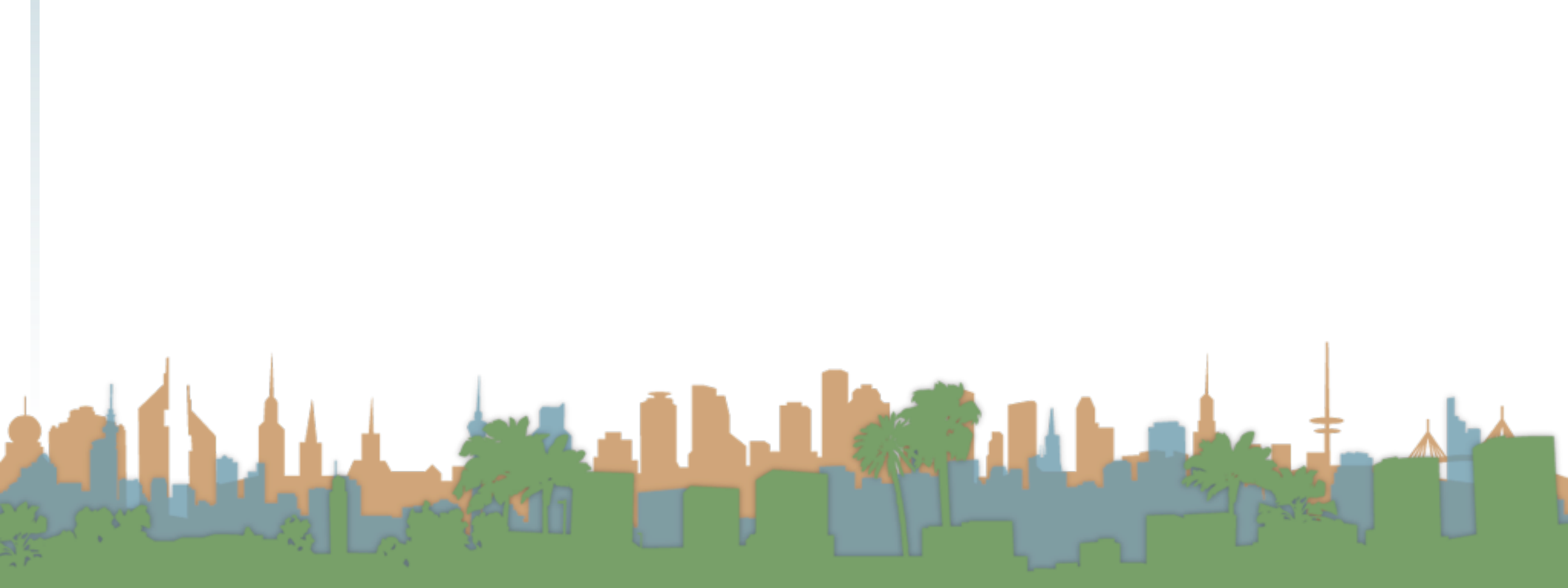
In the data files, each row corresponds to a query-url pair. The first column is relevance label of the pair, the second column is query id, and the following columns are features. The larger value the relevance label has, the more relevant the query-url pair is. A query-url pair is represented by a 136-dimensional feature vector. The details of features can be found here.

Below are two rows from MSLR-WEB10K dataset:

```
=================================================================
0 qid:1 1:3 2:0 3:2 4:2 ... 135:0 136:0
2 qid:1 1:3 2:3 3:0 4:0 ... 135:0 136:0
=================================================================
```

Unranked retrieval

- Precision:
  - Fraction of retrieved documents that are relevant
- Recall:
  - Fraction of relevant documents that are retrieved



Unranked retrieval

- Precision:
  - Fraction of retrieved documents that are relevant
- Recall:
  - Fraction of relevant documents that are retrieved

	Relevant	*Not Relevant*
Retrieved	*TP*	*FP*
Not Retrieved	*FN*	*TN*

Unranked retrieval

- Precision:
  - Fraction of retrieved documents that are relevant
- Recall:
  - Fraction of relevant documents that are retrieved

	Relevant	*Not Relevant*
Retrieved	TP	FP
Not Retrieved	FN	TN

? $$Precision = \frac{TP}{TP + FP}$$

? $$Recall = \frac{TP}{TP + FN}$$

Unranked retrieval - Accuracy

- The difficulty with measuring "accuracy"
  - In one sense accuracy is how many judgments you make correctly

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

	Relevant	*Not Relevant*
Retrieved	TP	FP
Not Retrieved	FN	TN

Exercise

- Documents A - F, Query q

$Document$	$Relevant(q)$	$Not\ Relevant(q)$
A	$\surd$	
B		$\surd$
C		$\surd$
D	$\surd$	
E		$\surd$
F	$\surd$	

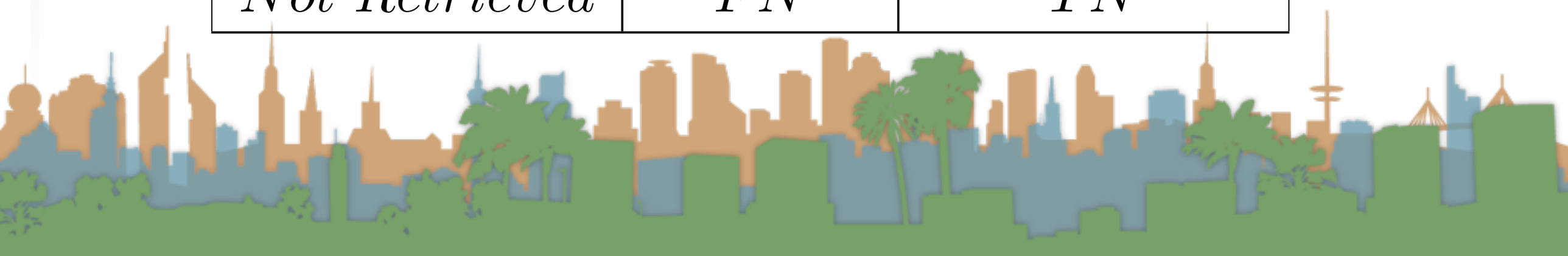
- If my system returns A,C,D,E to query q....
  - How many TP, TN, FP, FN do I have?

Exercise

$Retrieved: \ A \ C \ D \ E$

$Document$	$Relevant(q)$	$Not\ Relevant(q)$
A	$\surd$	
B		$\surd$
C		$\surd$
D	$\surd$	
E		$\surd$
F	$\surd$	

	$Relevant$	$Not\ Relevant$
$Retrieved$	TP	FP
$Not\ Retrieved$	FN	TN



Exercise

$Retrieved: \ A \ C \ D \ E$

$Document$	$Relevant(q)$	$Not\ Relevant(q)$
A	√	
B		√
C		√
D	√	
E		√
F	√	

	$Relevant$	$Not\ Relevant$
$Retrieved$	TP	FP
$Not\ Retrieved$	FN	TN

Exercise

- What is our precision?

$$Precision = \frac{TP}{TP + FP}$$

TP	2
FP	2
FN	1
TN	1

- What is our recall?

$$Recall = \frac{TP}{TP + FN}$$

- What is our accuracy?

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Exercise

- If my system returns A,C,D,E to query q....

$Document$	$Relevant(q)$	$Not\ Relevant(q)$
A	$\surd$	
B		$\surd$
C		$\surd$
D	$\surd$	
E		$\surd$
F	$\surd$	

$Precision$	$\frac{1}{2}$
$Recall$	$\frac{2}{3}$
$Accuracy$	$\frac{1}{2}$

- What do I want Precision to be?

	$Relevant$	$Not\ Relevant$
$Retrieved$	TP	FP
$Not\ Retrieved$	FN	TN

$$Precision = \frac{TP}{TP + FP}$$

Exercise

- If my system returns A,C,D,E to query q....

Document	*Relevant*(*q*)	*Not Relevant*(*q*)
A	√	
B		√
C		√
D	√	
E		√
F	√	

Precision	$\frac{1}{2}$
Recall	$\frac{2}{3}$
Accuracy	$\frac{1}{2}$

- What do I want Recall to be?

	Relevant	*Not Relevant*
Retrieved	*TP*	*FP*
Not Retrieved	*FN*	*TN*

$$Recall = \frac{TP}{TP + FN}$$

Exercise

- If my system returns A,C,D,E to query q....

$Document$	$Relevant(q)$	$Not\ Relevant(q)$
A	$\surd$	
B		$\surd$
C		$\surd$
D	$\surd$	
E		$\surd$
F	$\surd$	

$Precision$	$\frac{1}{2}$
$Recall$	$\frac{2}{3}$
$Accuracy$	$\frac{1}{2}$

- What do I want Accuracy to be?

	$Relevant$	$Not\ Relevant$
$Retrieved$	TP	FP
$Not\ Retrieved$	FN	TN

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

Unranked retrieval - Accuracy

Unranked retrieval - Accuracy

- Welcome to my search engine
  - I guarantee a 99.9999% accuracy.
  - Bring on the venture capital

Beta

PITTERPATTERSONFINDER

Search for:

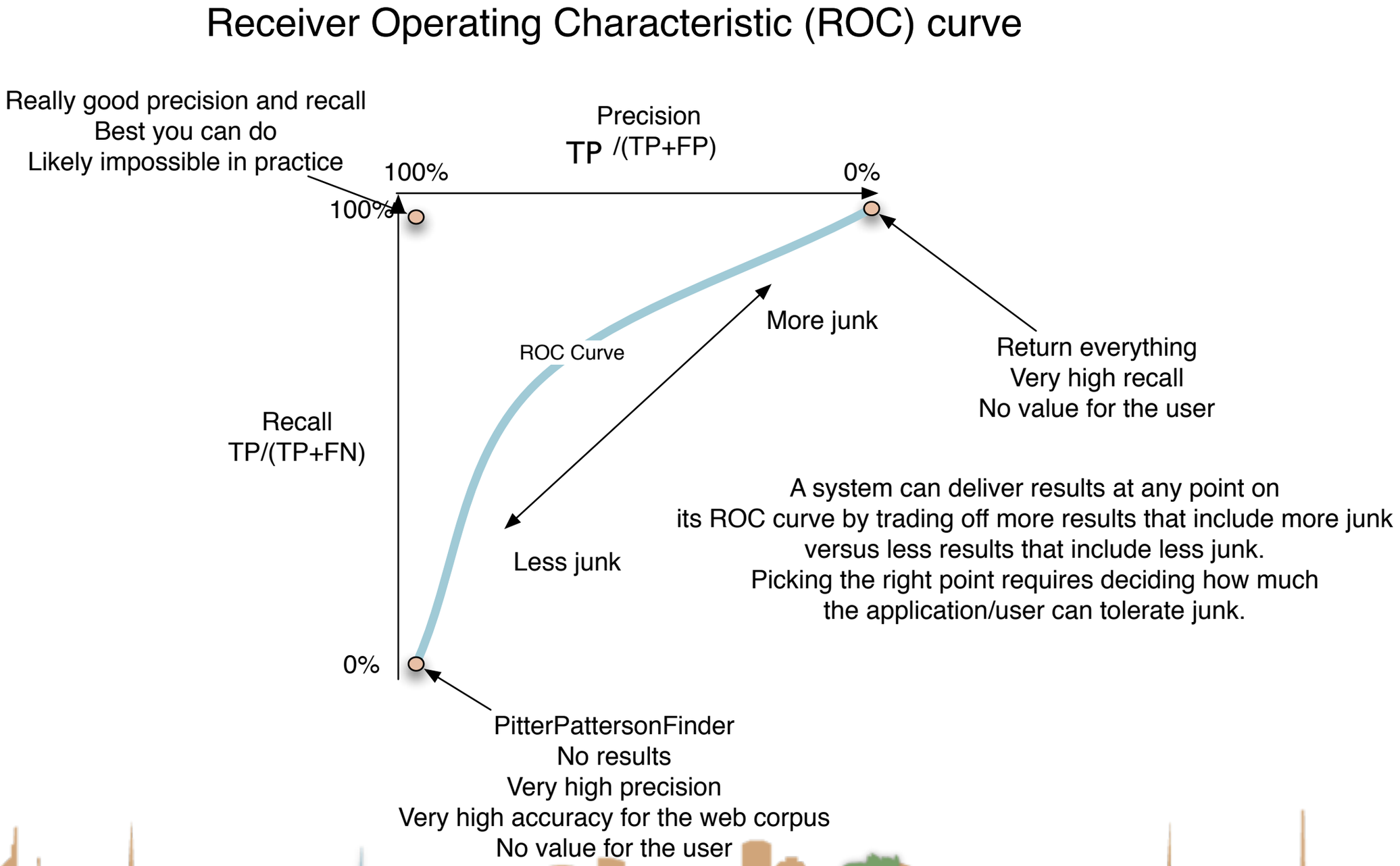
Unranked retrieval - Accuracy

- Most people want to find something and can tolerate some junk

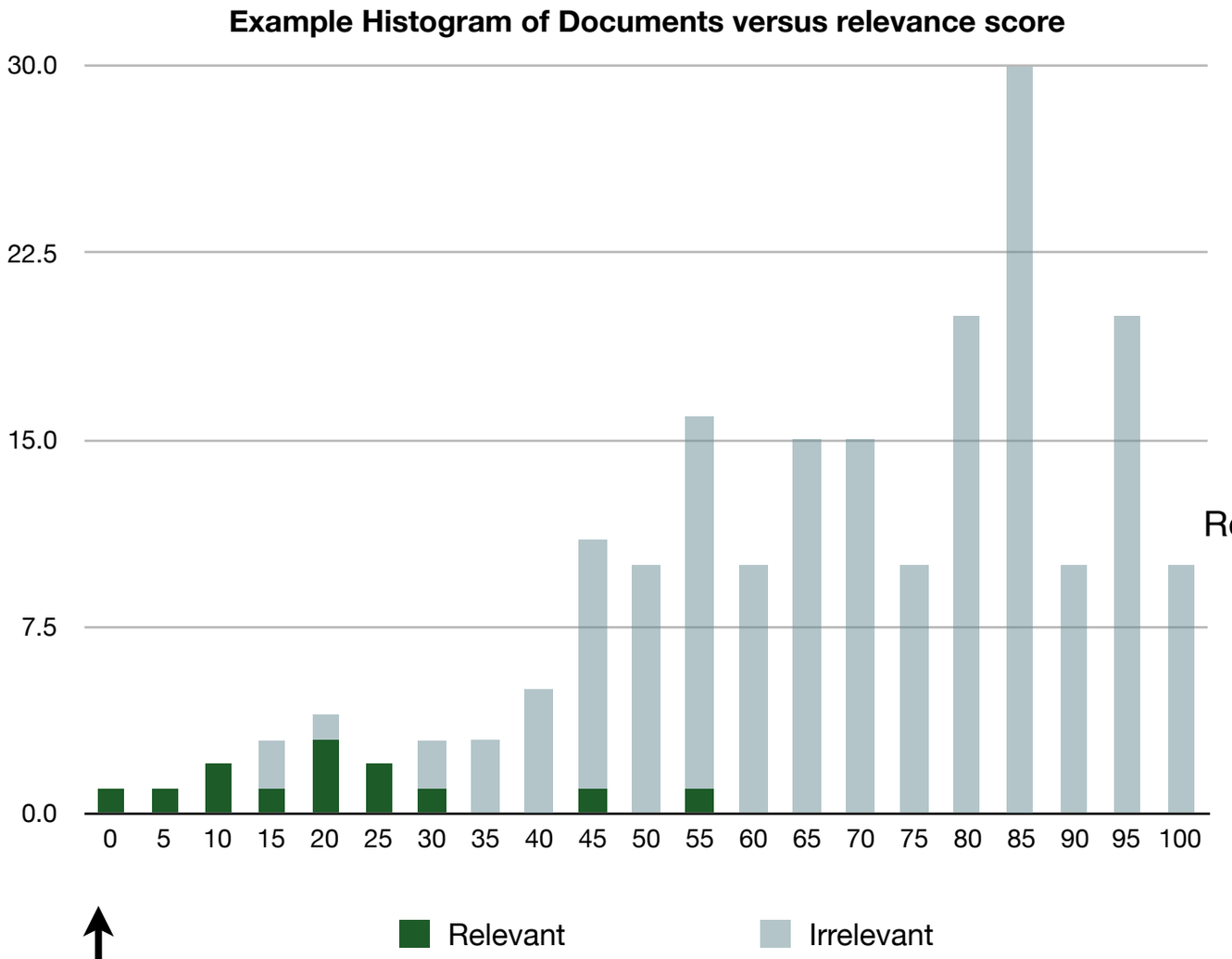
$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

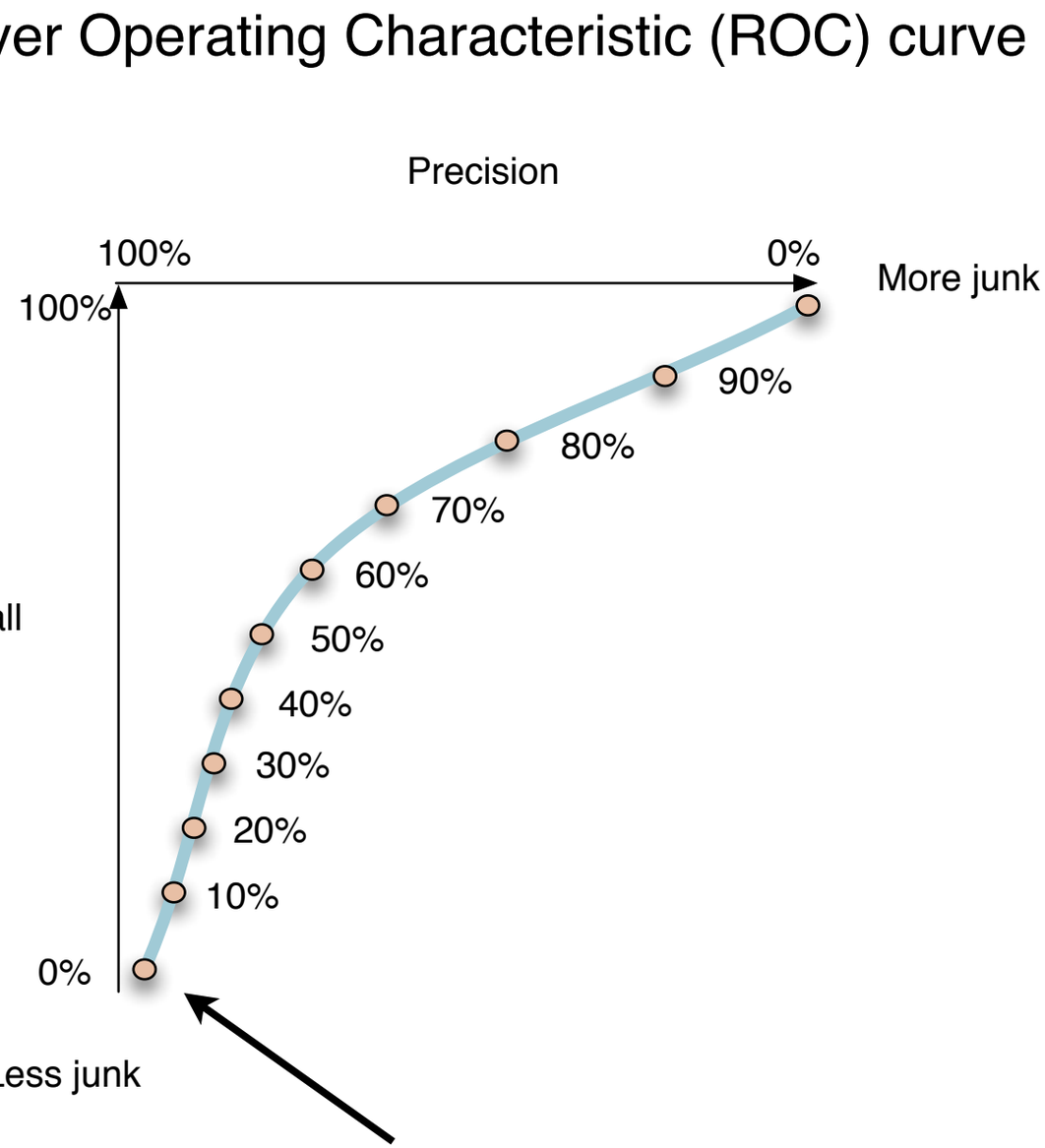
$$Accuracy = \frac{0 + \uparrow}{0 + 0 + \epsilon + \uparrow}$$

Unranked retrieval - ROC curve



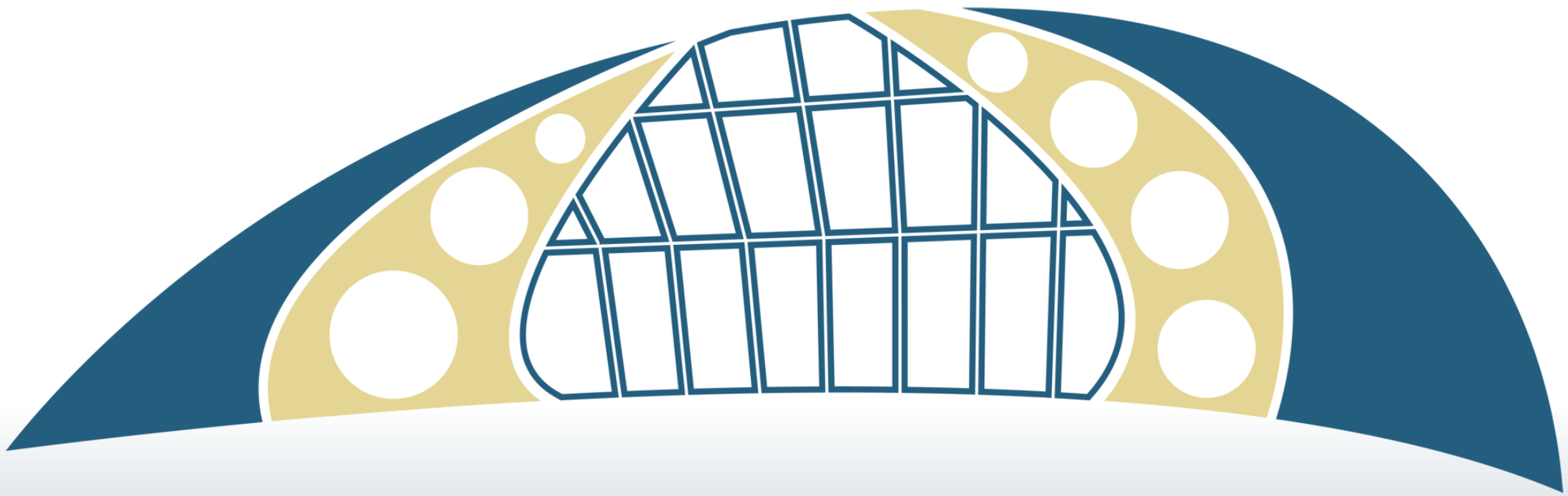
Unranked retrieval - ROC curve





Ranked Retrieval

- Precision and Recall are set-based measures
  - They are computed independent of order
  - But, web search return things in lists
  - Lists have order.
  - A better metric of user happiness/relevance is warranted



WESTMONT INSPIRED

COMPUTING LAB