

VK Multimedia Information Systems

Mathias Lux, mlux@itec.uni-klu.ac.at

Retrieval Evaluation: Agenda



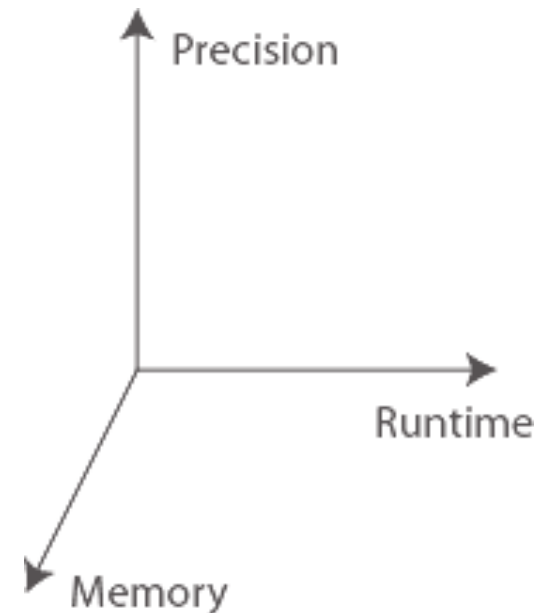
- **Retrieval Evaluation**
- The Lucene Search Engine
- Exercise 03



Retrieval Evaluation: Motivation



- Compare **objectively** different
 - Search engines
 - Models & weighting Schemes
 - Methods & techniques
- **Scope**
 - Academic
 - Commercial & industrial
- **Different aspects**
 - Runtime, retrieval performance



Retrieval Evaluation



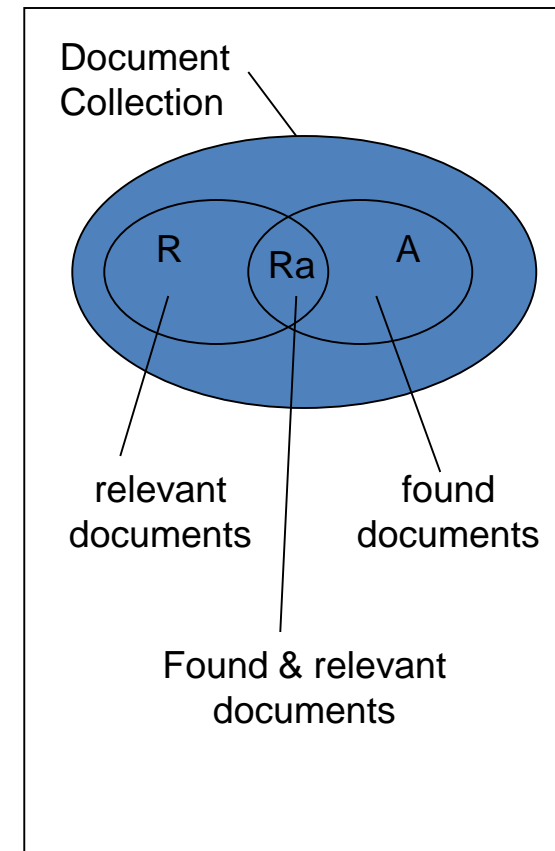
- Comparability issues:
 - Test collections
 - Experts assessing retrieval performance
 - Metrics
 - What's good? / What's bad?
- Overall problem:
 - What is relevant?

Metrics: Precision & Recall



Within a document collection
 D with a given query q

- $|R|$.. num. of relevant docs
- $|A|$.. num. of found docs
- $|R_a|$.. num. found & relevant



Metrics: Precision



$$\text{Precision} = \frac{|Ra|}{|A|} = \frac{\text{found relevant docs}}{\text{found docs}}$$

- Gives % how many of the actual found documents have been relevant
- Between 0 and 1
 - Optimum: 1 ... all found docs are relevant

Metrics: Recall



$$\text{Recall} = \frac{|Ra|}{|R|} = \frac{\text{found relevant docs}}{\text{relevant docs}}$$

- Gives % how many of the actual relevant documents have been found
- Between 0 and 1
 - Optimum: 1 ... all relevant docs are found

Metrics: Precision & Recall



- With a query only 1 document has been found, but this one is relevant (100 would be relevant):
 - Precision & Recall
 - **Precision = 1**
 - **Recall = 0,01**

Metrics: Precision & Recall



- With a query all documents of D have been found (5% of D would be relevant)
 - Precision & Recall?
 - **Precision = 0,05**
 - **Recall = 1**

Example



- $D = \{D00, D01, \dots D99\}$
- Query 1:
 - Result Set 1: **{D2, D14, D25, D76, D84, D98}**
 - Relevant Docs {D1, D2, D14, D22, D23, D25, D84, D89, D90, D98}
- Query 2:
 - Result Set 1: **{D10, D14, D60, D63, D77, D95}**
 - Relevant Docs {D10, D14}

Recall vs. Precision Plot



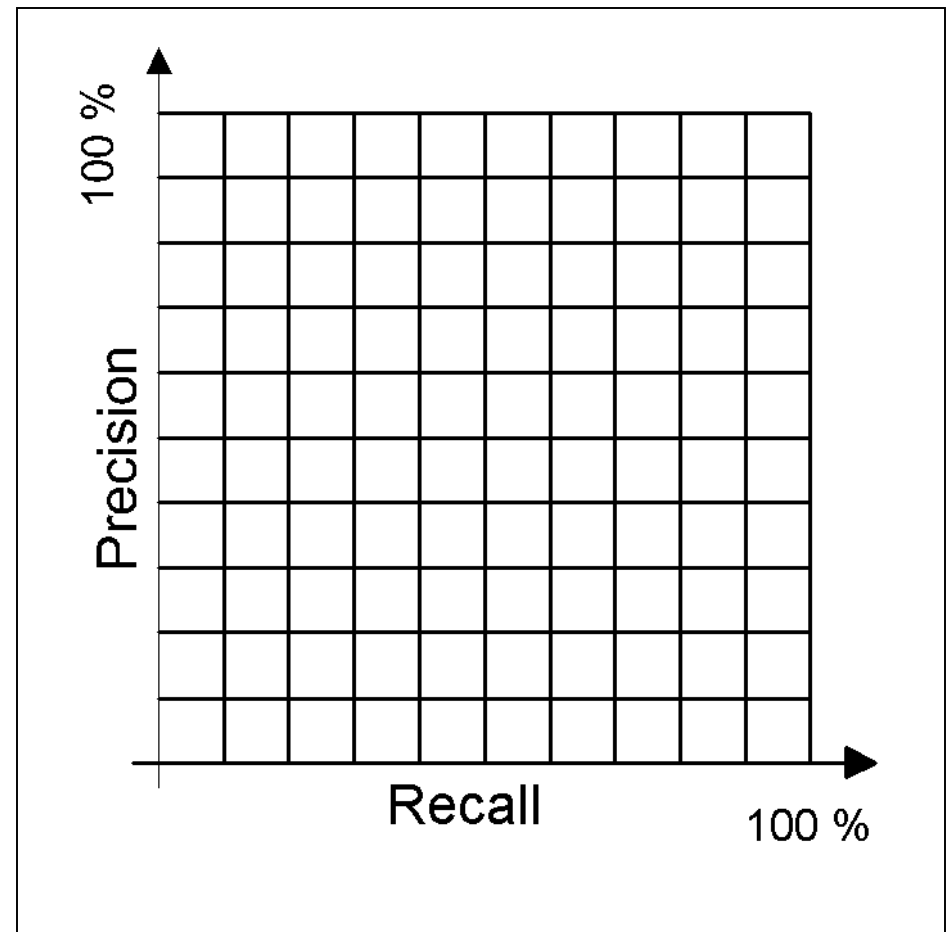
- **Assumption:**
 - Result list is sorted by descending relevance
 - User investigates result list linearly
 - when recall changes ...
- **Approach:**
 - Map different states to graph

Recall vs. Precision Plot



- | | | |
|------------|-----------|----------|
| 01. d123 * | 06. D9 * | 11. d38 |
| 02. d84 | 07. d511 | 12. d48 |
| 03. d56 * | 08. d129 | 13. d250 |
| 04. d6 | 09. d187 | 14. d113 |
| 05. d8 | 10. d25 * | 15. d3 * |

$R_q = \{d3, d5, d9, d25, d39, d44, d56, d71, d89, d123\} \rightarrow 10$



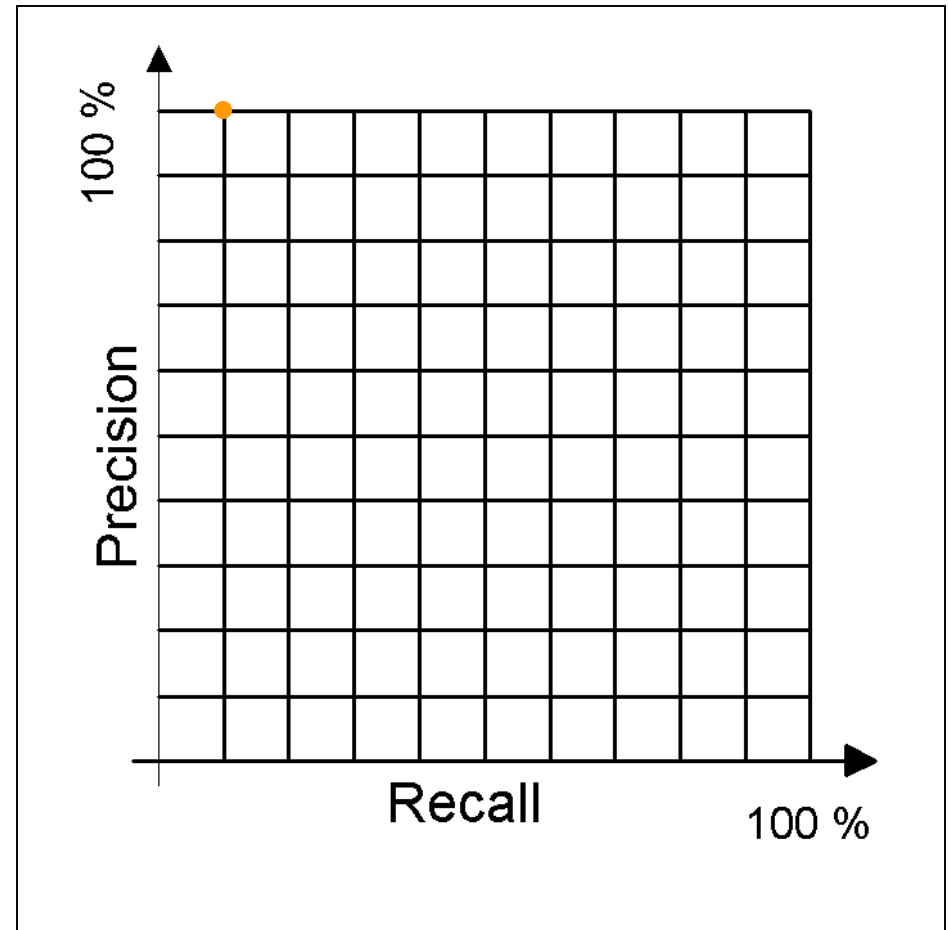
Recall vs. Precision Plot



- | | | |
|------------|-----------|----------|
| 01. d123 * | 06. D9 * | 11. d38 |
| 02. d84 | 07. d511 | 12. d48 |
| 03. d56 * | 08. d129 | 13. d250 |
| 04. d6 | 09. d187 | 14. d113 |
| 05. d8 | 10. d25 * | 15. d3 * |

$$\text{Recall} = \frac{|Ra|}{R} = \frac{1}{10}$$

$$\text{Precision} = \frac{|Ra|}{A} = \frac{1}{1}$$



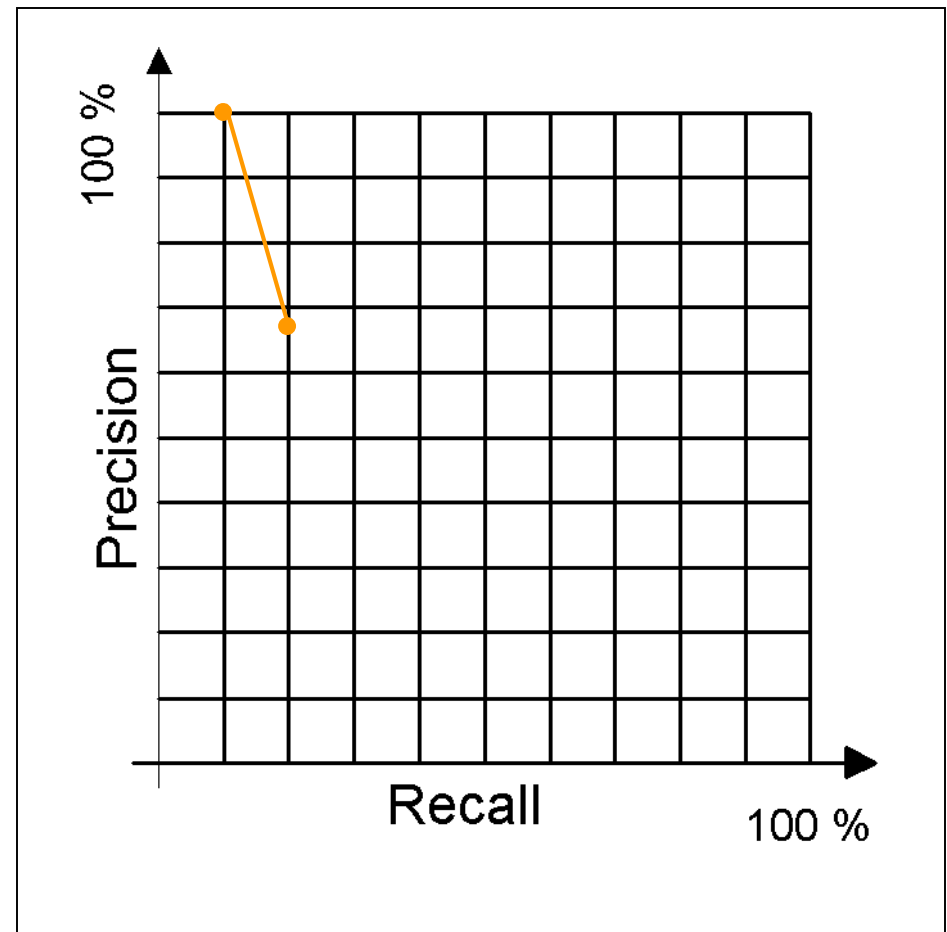
Recall and Precision



- | | | |
|------------|-----------|----------|
| 01. d123 * | 06. D9 * | 11. d38 |
| 02. d84 | 07. d511 | 12. d48 |
| 03. d56 * | 08. d129 | 13. d250 |
| 04. d6 | 09. d187 | 14. d113 |
| 05. d8 | 10. d25 * | 15. d3 * |

$$\text{Recall} = \frac{|Ra|}{R} = \frac{2}{10}$$

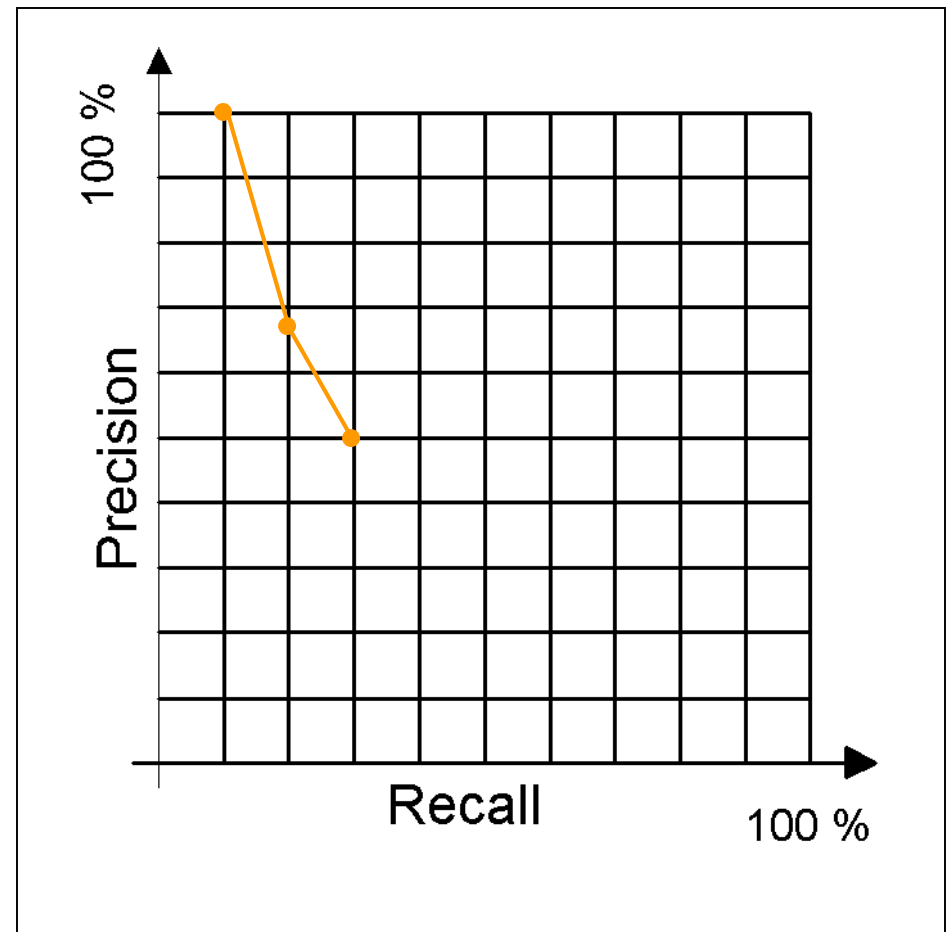
$$\text{Precision} = \frac{|Ra|}{A} = \frac{2}{3}$$



Recall and Precision



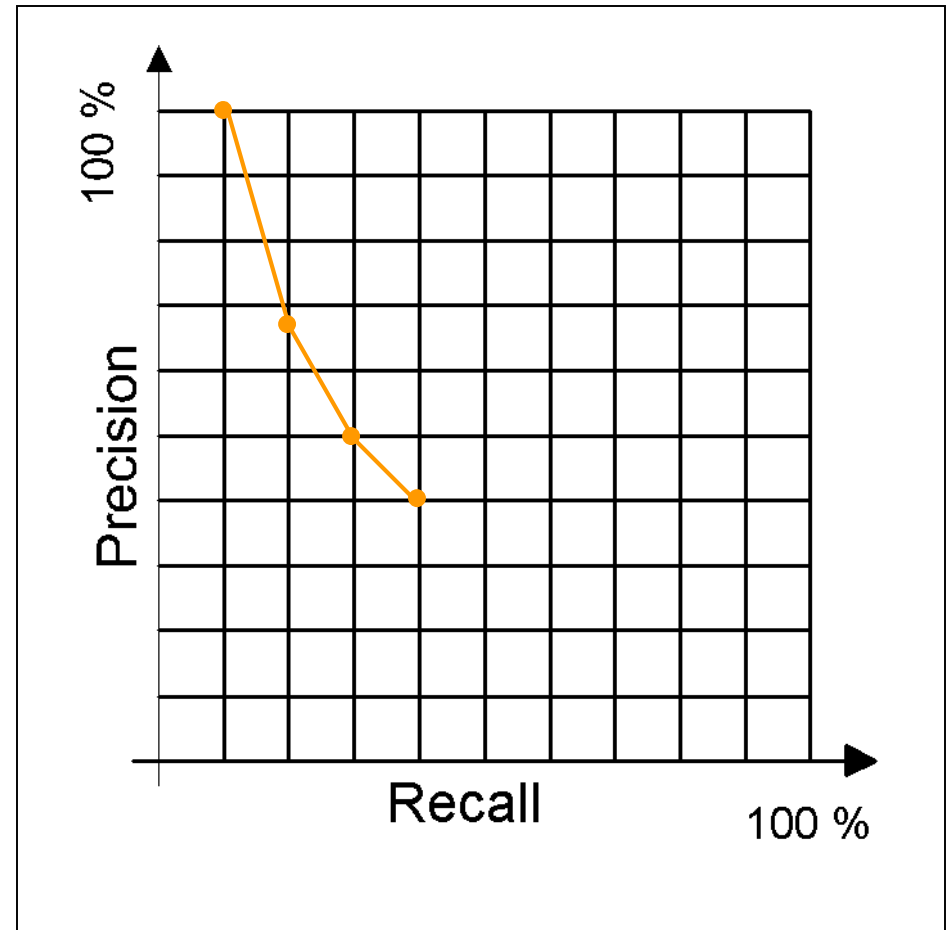
- | | | |
|------------|-----------|----------|
| 01. d123 * | 06. D9 * | 11. d38 |
| 02. d84 | 07. d511 | 12. d48 |
| 03. d56 * | 08. d129 | 13. d250 |
| 04. d6 | 09. d187 | 14. d113 |
| 05. d8 | 10. d25 * | 15. d3 * |



Recall and Precision



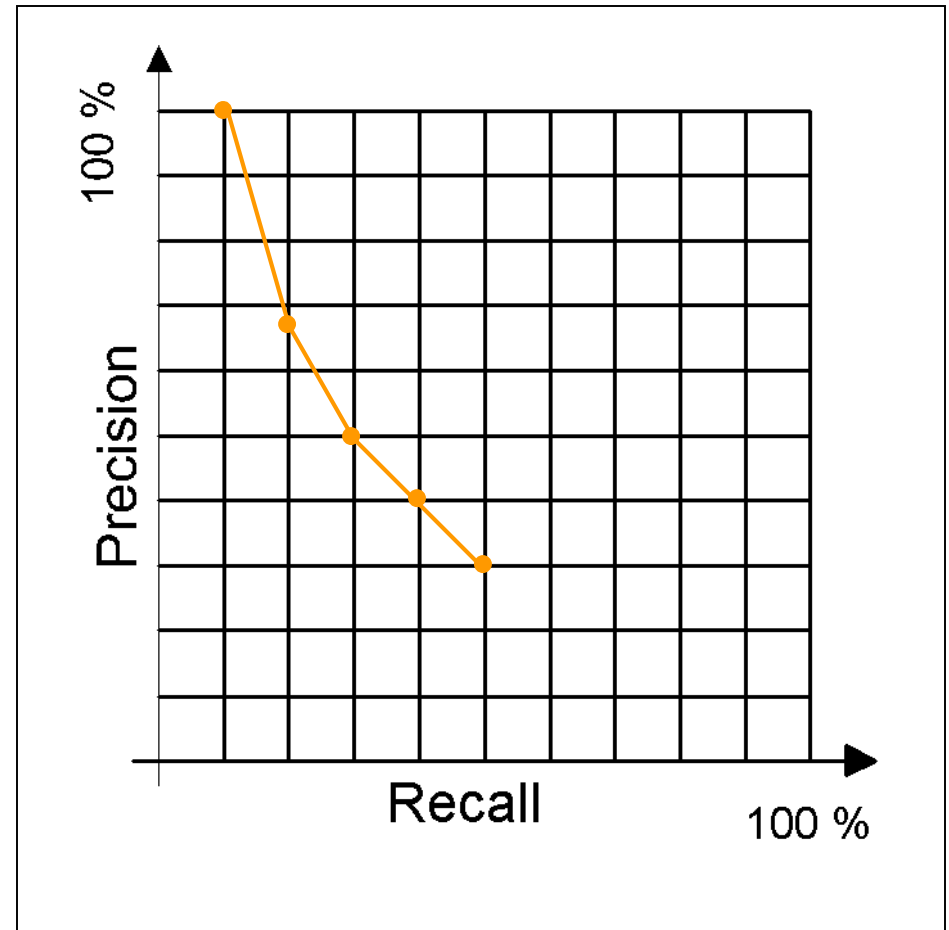
- | | | |
|------------|-----------|----------|
| 01. d123 * | 06. D9 * | 11. d38 |
| 02. d84 | 07. d511 | 12. d48 |
| 03. d56 * | 08. d129 | 13. d250 |
| 04. d6 | 09. d187 | 14. d113 |
| 05. d8 | 10. d25 * | 15. d3 * |



Recall and Precision



- | | | |
|------------|-----------|----------|
| 01. d123 * | 06. D9 * | 11. d38 |
| 02. d84 | 07. d511 | 12. d48 |
| 03. d56 * | 08. d129 | 13. d250 |
| 04. d6 | 09. d187 | 14. d113 |
| 05. d8 | 10. d25 * | 15. d3 * |



F-Measure



$$E(j) = 1 - \frac{1 + b^2}{\frac{b^2}{\text{recall}(j)} + \frac{1}{\text{precision}(j)}}$$

$F(j) = 1 - E(j)$... van Rijsbergen

- Lower values -> lower performance
- If $b=1$, $F(j)$ is average
- If $b=0$, $F(j)$ is precision
- If $b=\text{inf}$, $F(j)$ is recall
- $b=2$ is a common choice

Mean Average Precision (MAP)



- Find average precision for each query
- Compute mean AP over all queries
 - Macroaverage: all queries are considered equal
- For average recall-precision curves
 - Average at standard recall points

Mean Average Precision (MAP)



Example: Query Q1:

1. D12 (relevant) -> Precision: 1
2. D61
3. D39 (relevant) -> Precision: 2/3
4. D75 (relevant) -> Precision: 3/4
5. D66
6. D14 (relevant) -> Precision: 4/6
7. D52
8. D33 (relevant) -> Precision: 5/8

- Average Precision: $(1 + 2/3 + \dots) / 5 = 0.742$

Mean Average Precision (MAP)



- Compute MAP:
 - Q1: 0,742
 - Q2: 0,633
 - Q3: 0,874
 - Q4: 0,722
- $MAP = (0,742 + 0,633 + ..) / 4 = 0,743$

MAP



$$AP(q) = \frac{1}{N_R} \sum_{n=1}^{N_R} P_q(R_n), \quad MAP = \frac{1}{|Q|} \sum_{q \in Q} AP(q),$$

src. Deselaers, T., Keysers D., and Ney H., "Features for Image Retrieval: An Experimental Comparison", Information Retrieval, vol. 11, issue 2, Springer 2008.

Precision @ 10



- Precision for the first 10 results
- Measures the quality of the first page
- Motivated by
 - Subjective impression that they all should be relevant
 - Fact that many people examine only first page

Error classification rate



- Instance based learning
 - First result gives the class
- Inverse Precision @ 1

$$ER = \frac{1}{|Q|} \sum_{q \in Q} \begin{cases} 0 & \text{if the most similar image is relevant} \\ 1 & \text{otherwise} \end{cases}$$

src. Deselaers, T., Keysers D., and Ney H., "Features for Image Retrieval: An Experimental Comparison", Information Retrieval, vol. 11, issue 2, Springer 2008.

Test Collections & Initiatives



- Aim:
 - Provide data, topic & results
- Prominent Initiatives
 - Text Retrieval Conference (TREC)
 - Initiative for the Evaluation of XML Retrieval (INEX)
 - Cross Language Evaluation Forume (CLEF)

The TREC Collection

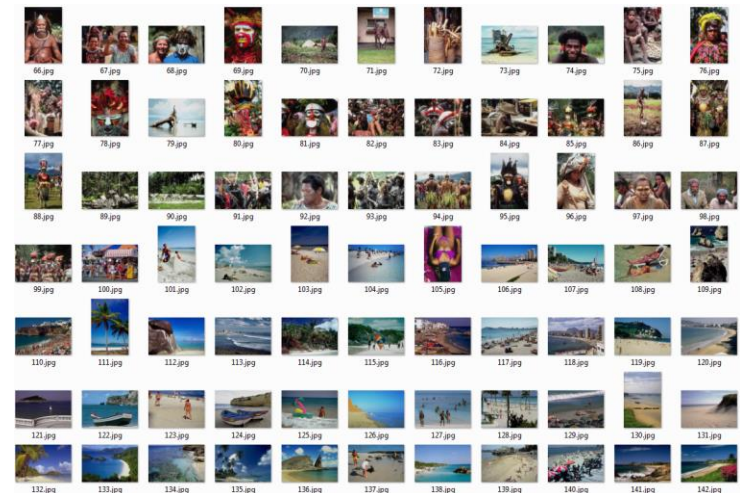


- Aim: Support IR Research on big data collections with
 - Test collection
 - Uniform measures and methods
 - Platform for comparison & challenges
- TREC collection size increases steadily
- Several different tracks:
 - Ad hoc, Web, Blog, Confusion, Genomics Track, Question Answering, Spam, Terabyte
- Examples:
 - Spam: ~ 91.000 messages (300 MB zipped)
 - Ad hoc has 5 sets:
 - e.g. Disk 5: 260.000 documents (1 GB zipped)

Prominent VIR data sets



- The SIMPLicity data set
 - 1,000 images from the Corel stock photos.
 - 10 categories with 100 images each.



SIMPLicity Example



- LireDemo

Benchmarking



- Show LIRE approach

Summary: Evaluation



- Lots of measures exist besides Precision & Recall
- Selection based on Use Case & Scenario
- Initiatives & Collections allow comparison
- Also user centered evaluation methods exist
- collections & initiatives are criticized:
 - Handling of outliers, significance of differences, ...

Retrieval Evaluation: Agenda



- Retrieval Evaluation
- **The Lucene Search Engine**
- Exercise 03



Lucene



- **A Java text search engine**
 - .NET Implementation exists
 - Also used in PHP, etc.
- **Initiated by Doug Cutting**
 - Later developed Hadoop
 - Yahoo! then Cloudera
 - On the board of the Apache Software Foundation

Lucene



- Implements an **inverted list**
 - Stores term -> document
 - Per field (e.g. title, content, ...)
 - And additional information (count, position, length, etc.)
 - File format & storage.
- **Preprocesses input**
 - Stemming, etc.
- **Provides search & index update**
 - Query, Ranking

Inverted list example ...



1. hello world
 2. hello bob
 3. hello! say hello to bob.
 4. around the world
 5. bob's world
- hello -> 1, 2, 3
 - world -> 1, 4, 5
 - bob -> 2, 3, 5
 - say -> 3
 - to -> 3
 - around ->4
 - the -> 4

even better:

hello (3 docs, 4 occ.) ->

1 (pos. 0), 2 (pos 0), 3 (pos. 0, pos. 12)

Inverted list example ...



- hello -> 1, 2, 3
- world -> 1, 4, 5
- bob -> 2, 3, 5
- say -> 3
- to -> 3
- around ->4
- the -> 4
- IDF
 - # of documents known
 - Size of corpus known
- TF
 - # of occurrences known
- Other stats
 - Length of document
 - Avg. length of docs
 - etc.

Lucene: Basic Usage



Let `lucene- $\{version\}$.jar` and `lucene-demos- $\{version\}$.jar` be in your classpath

- To index files type:
 - `java org.apache.lucene.demo.IndexFiles [dir]`
- To search in the index type:
 - `java org.apache.lucene.demo.SearchFiles`

Lucene: Queries



- Lucene has an extensive query parser
 - Parses text to internal representation
- Lucene supports several types of queries
 - Field based: title:"multimedia information"
 - Boolean clauses: multimedia AND image
 - Wildcards: te?t OR te*t
 - Fuzzy search: roam~ (e.g. *foam* and *roams*)
 - Proximity search: "java apache"~10
 - Term boosting: java^4 apache

Lucene File Format



- **Definitions:**
 - An index contains a sequence of documents.
 - A document is a sequence of fields.
 - A field is a named sequence of terms.
 - A term is a string.
- **Lucene uses**
 - different types of fields:
 - stored, indexed, tokenized
 - Sub-indexes (segments, upon insertion)

Lucene: Usage



- IndexWriter
 - Writes documents to the index
 - Uses Analyzer
- IndexSearcher
 - Searching documents in an index
 - Same Analyzer as for indexing needed
 - A Hits object is returned
- Document
 - Groups fields to logical unit

Lucene: Features



- It's really fast & stable
 - Even compared to commercial products
- Handles multiple indexes
 - MultiReader, distributed search
- Has strong development support
 - Yahoo! & Apache (top level project)
- Lots of Stemmers, Tokenizers, etc.
 - English, German, Korean, Chinese, ...

Lucene: Projects & Tools



- Nutch
 - Open source internet search engine
- Lucene .NET
 - Source code port to .NET
- Solr
 - Search server supporting web services, REST, ..
- Luke
 - GUI index management tool

Retrieval Evaluation: Agenda



- Retrieval Evaluation
- The Lucene Search Engine
- **Exercise 03**



Exercise 03



- Retrieval Evaluation
 - Collection of 35 documents
 - Query and unsorted list of relevant documents - R
 - Ranked lists of 2 different search engines (A1 & A2)
- Your task
 - Compute precision and recall
 - Draw precision vs. recall plot
 - Compare A1 and A2 based on your findings

Don't forget!



- Send me your results!