Document Indexation and Multimedia Retrieval

Axel Carlier, Axel.Carlier@enseeiht.fr

INP ENSEEIHT

Information Retrieval Basics: Agenda

- **Information Retrieval - Searching**
- Information Retrieval Model - Reminder
- Evaluation

INP ENSEEIHT

Information Retrieval Basics: Searching

A **user** has an **information need**, which needs to be **satisfied**.

- Two different approaches:
  - Browsing
  - Searching

Searching & Browsing

Searching

- Explicit information need
- Definition through “query”
- Result lists
- e.g. Google

Browsing

- Not necessarily explicit need
- Navigation through repositories

Browsing

- Flat Browsing
  - User navigates through set of documents
  - No implied ordering, explicit ordering possible
  - Examples: One single directory, one single file

- Structure Guided Browsing
  - An explicit structure is available for navigation
  - Mostly hierarchical (file directories)
  - Can be generic digraph (WWW)
  - Examples: File systems, World Wide Web

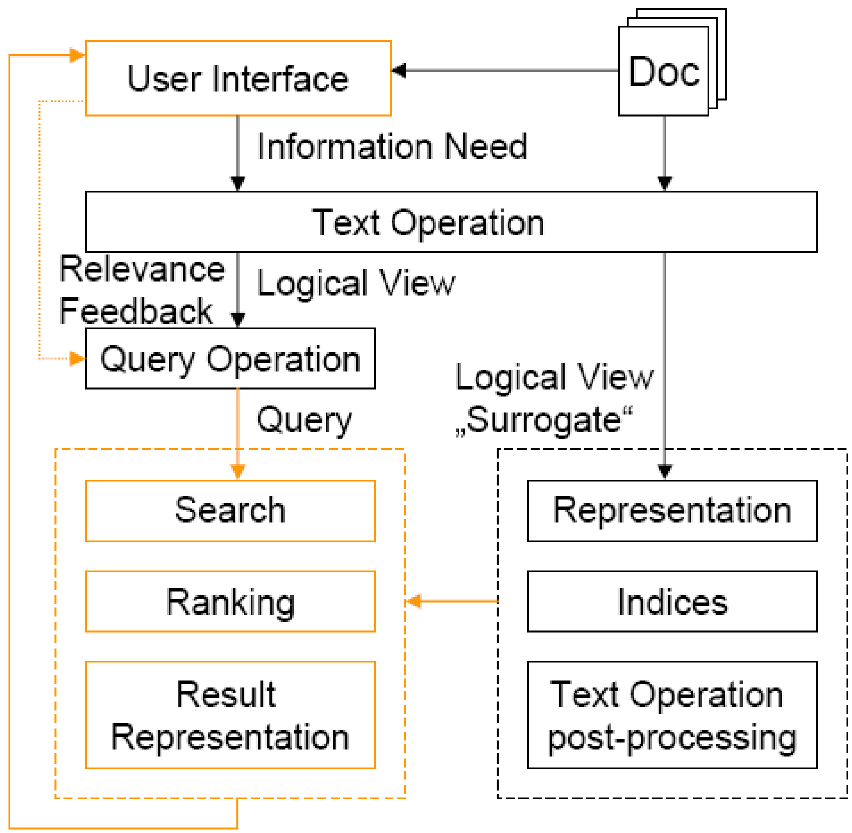
Information Retrieval Basics: Agenda

- Information Retrieval - Searching
- Information Retrieval Model - Reminder
- Evaluation

Information Retrieval System Architecture

Aspects

- Query & languages
- IR models
- Documents
- Internal representation
- Pre- and post-processing
- Relevance feedback
- HCI



Information Retrieval Models

- Boolean Model
  - Set theory & Boolean algebra
- Vector Model
  - Non binary weights on dimensions
  - Partial match
- Probabilistic Model
  - Modeling IR in a probabilistic framework

INP ENSEEIHT

Formal Definition of Models

An information retrieval model is a quadruple [D, Q, F, $R(q_i, d_j)$]

- D is a set of logical views (or representations) for the **documents** in the collection.
- Q is a set of logical views (or representations) for the user needs or **queries**.
- F is a **framework** for modeling document representations, queries and their relationship.
- $R(q_i, d_j)$ is a **ranking function** which associates a real number with a query q_i of Q and a document d_j of D.

Definitions
in Context of Text Retrieval

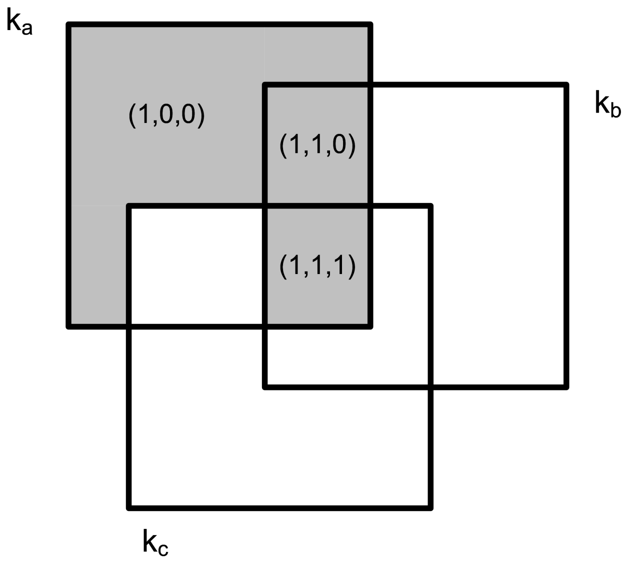
- **index term** – word of a document expressing (part of) document semantics
- **weight $w_{i,j}$** – quantifies the importance of index term t_i for document d_j
- index term vector for document d_j (having t different terms in all documents):

  $$d_j = (w_{1,j}, w_{2,j}, \ldots, w_{t,j})$$

Boolean Model

- Based on set theory and Boolean algebra
  - Set of index terms
  - Query is Boolean expression
- Intuitive concept:
  - Wide usage in bibliographic system
  - Easy implementation and simple formalisms
- Drawbacks:
  - Binary decision components (true/false)
  - No relevance scale (relevant or not)

Boolean Model: Example



$$q = k_a \wedge (k_b \vee \neg k_c)$$

Boolean Model: DNF

$$q = k_a \wedge (k_b \vee \neg k_c) \; \ldots \; q_{dnf} = (1,1,1) \vee (1,1,0) \vee (1,0,0)$$

- Express queries in *disjunctive normal form* (disjunction of conjunctive components)
- Each of the components is a binary weighted vector associated with (k_a, k_b, k_c)
- Weights $w_{i,j} \in \{0,1\}$

Boolean Model: Ranking function

$$sim(d_j, q) = \begin{cases} 1 & \text{if } \exists q_{cc} \,\middle|\, (q_{cc} \in q_{dnf}) \wedge (\forall_{k_i}, g_i(d_j) = g_i(q_{cc})) \\ 0 & \text{otherwise} \end{cases}$$

- similarity is one if one of the conjunctive components in the query is exactly the same as the document term vector.

Boolean Model

- Advantages
  - Clean formalisms
  - Simplicity
- Disadvantages
  - Might lead to too few / many results
  - No notion of **partial match**
  - Sequential ordering of terms not taken into account.

Vector Model

- Integrates the notion of partial match
- Non-binary weights (terms & queries)
- Degree of similarity computed

$$d_j = (w_{1,j}, w_{2,j}, \ldots, w_{t,j})$$

$$q = (w_{1,q}, w_{2,q}, \ldots, w_{t,q})$$

INP ENSEEIHT

Vector model: Similarity

$$sim(d_j, q) = \frac{d_j \bullet q}{\left|d_j\right| \times \left|q\right|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}}$$

Vector Model: Example

Another Example:

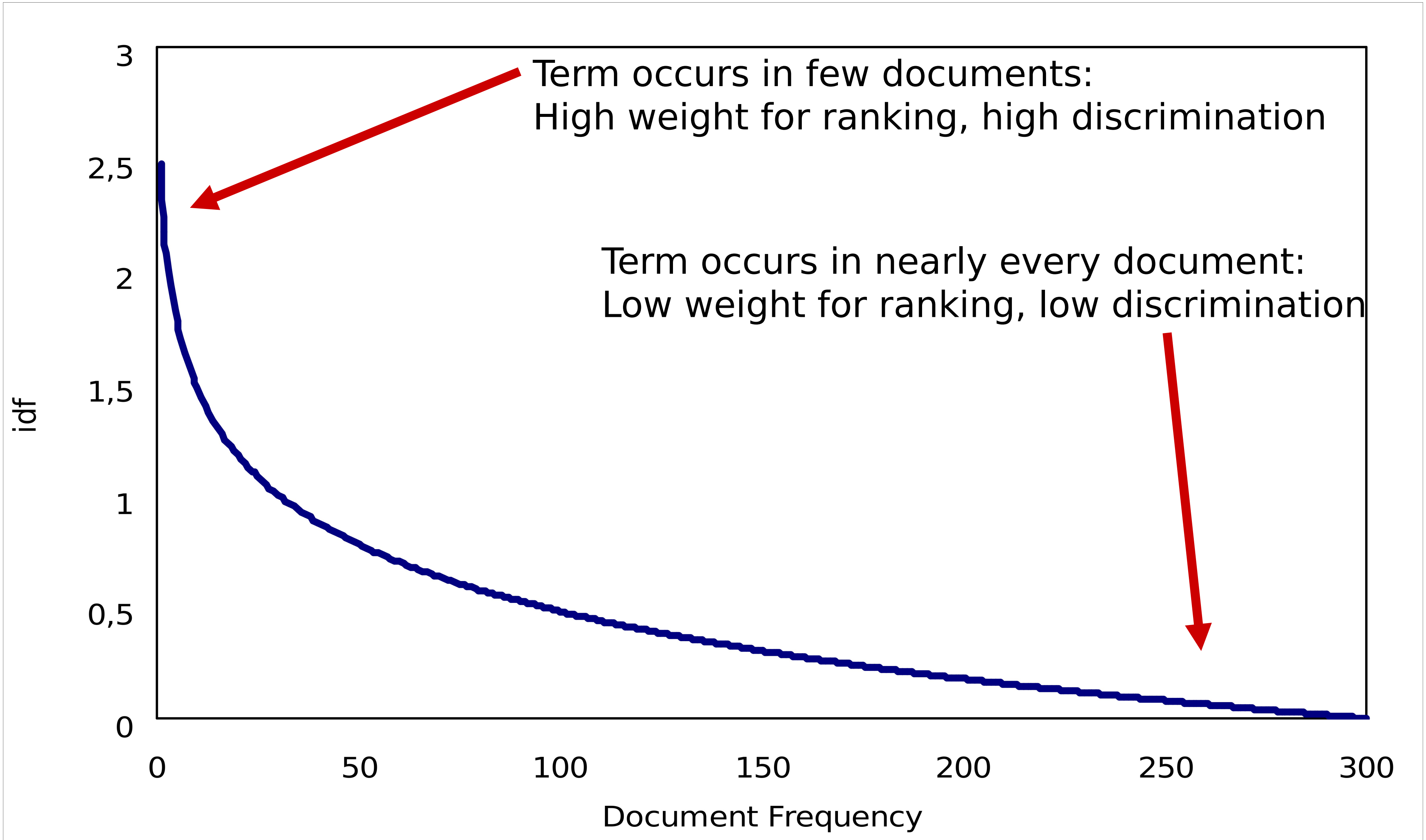
- Document & Query:
  - D = “The quick brown fox jumps over the lazy dog”
  - Q = “brown lazy fox”
- Results:

Term weighting: TF*IDF

Term weighting increases retrieval performance

- Term frequency
  - How often does a term occur in a document?
  - Most intuitive approach
- Inverse Document Frequency
  - What is the information content of a term for a document collection?
  - Compare to *Information Theory* of Shannon

Example: IDF
300 documents corpus



Definitions:
Normalized Term Frequency

$$f_{i,j} = \frac{freq_{i,j}}{\max_l(freq_{l,j})} \quad \text{... normalized term frequency}$$

$freq_{i,j}$ ... raw term frequency of term i in document j

- Maximum is computed over all terms in a document
- Terms which are not present in a document have a raw frequency of *0*

INP ENSEEIHT

Definitions: Inverse Document Frequency

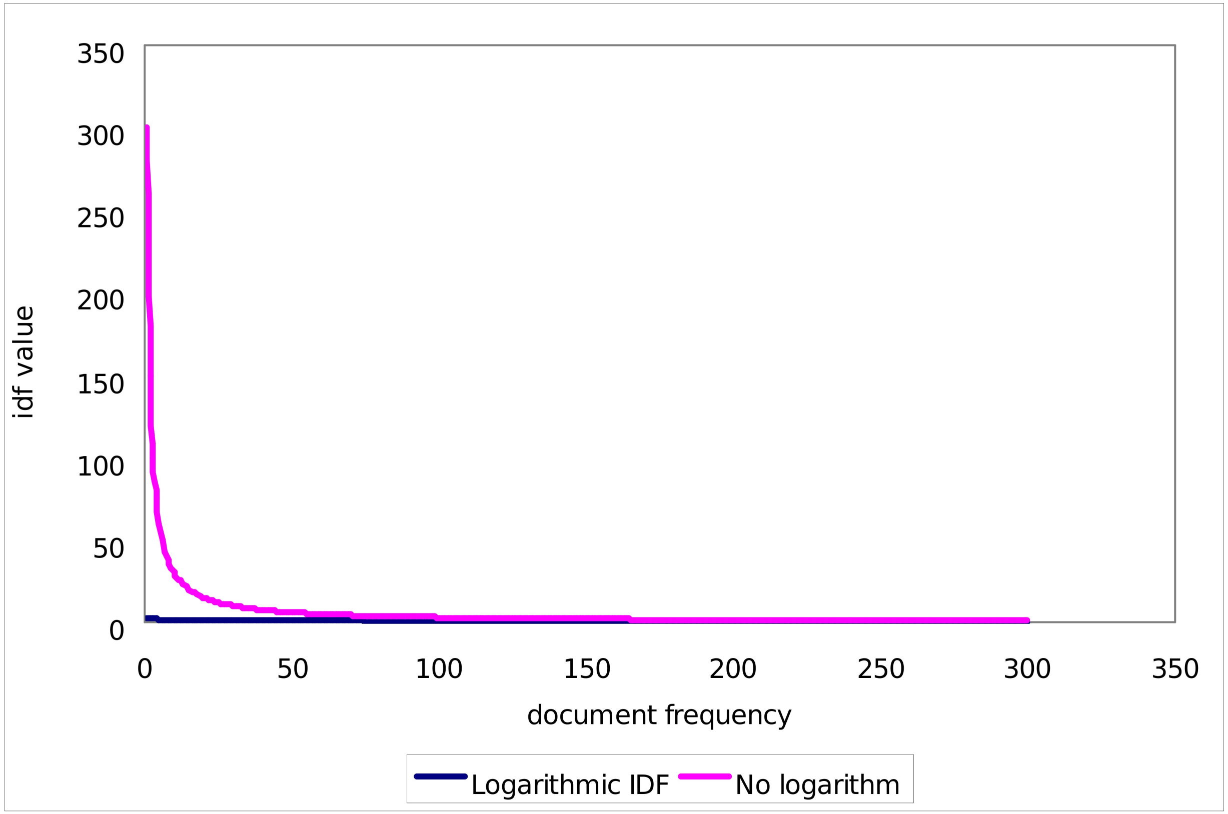
$$idf_i = \log \frac{N}{n_i} \text{ ... inverse document frequency for term } i$$

N ... number of documents in the corpus

n_i ... number of document in the corpus which contain term i

- Note that idf_i is independent from the document.
- Note that the whole corpus has to be taken into account.

Why log(...) in IDF?



TF*IDF

- TF*IDF is a very prominent weighting scheme
  - Works fine, much better than TF or Boolean
  - Quite easy to implement

$$w_{i,j} = f_{i,j} \times \log \frac{N}{n_i}$$

Weighting of query terms

$$w_{i,q} = (0.5 + \frac{0.5 \times f_{i,q}}{\max_l (f_{l,q})}) \times \log \frac{N}{n_i}$$

- Also using IDF of the corpus
- But TF is normalized differently
  - TF > 0.5
- Note: the query is not part of the corpus!

Vector Model

- Advantages
  - Weighting schemes improve **retrieval performance**
  - Partial matching allows retrieving documents that **approximate query** conditions

Simple example (i)

- Scenario
  - Given a **document corpus on birds**: nearly each document (say 99%) contains the word bird
  - someone is searching for a document about sparrow nest construction with a query **"sparrow bird nest construction"**
  - Exactly the document which would satisfy the user needs **does not have the word "bird"** in it.

INP ENSEEIHT

Simple example (ii)

- TF*IDF weighting
  - knows upon the low discrimative power of the term bird
  - The weight of this term is near to zero
  - This term has virtually no influence
  - on the result list.

Exercise

- Given a document collection ...
- Find the results to a query ...
  - Employing the Boolean model
  - Employing the vector model (with TF*IDF)

Exercise

- Document collection (6 documents)
  - Sparrow, blackbird, bird, bluebird, finch, falcon, flight
  - Sparrow, bird, flight, nest, blackbird, blackbird, blackbird
  - Cuckoo, nest, nest, egg, egg, egg, flight, blackbird, blackbird, bird
  - Amsel, magpie, magpie, throttle, bird, egg
  - falke, katze, nest, nest, flug, vogel
  - Sparrow, sparrow, construction, nest, egg
- Queries:
  - sparrow, bird, nest, construction
  - blackbird, egg, nest

Query Modification

- Query expansion
  - General method to increase either
    - number of results or
    - accuracy
  - Query itself is modified:
    - Terms are added (co-occurrence, thesaurii)

Relevance Feedback

- Popular Query Reformulation Strategy:
  - –User gets list of docs presented
  - –User marks relevant documents
  - –Typically~10-20 docs are presented
  - –Query is refined, new search is issued
- Proposed Effect:
  - –Query moves more toward relevant docs
  - –Away from non relevant docs
  - –User does not have to tune herself

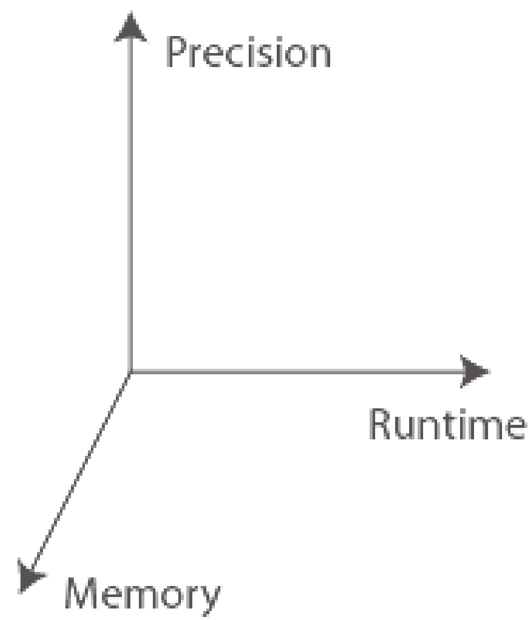
INP ENSEEIHT

Information Retrieval Basics: Agenda

- Information Retrieval - Searching
- Information Retrieval Model - Reminder
- **Evaluation**

Retrieval Evaluation: Motivation

- Compare **objectively** different
  - Search engines
  - Models & Weighting Schemes
  - Methods & Techniques
- Scope
  - Academic
  - Commercial & Industrial
- Different aspects
  - Runtime, Retrieval performance



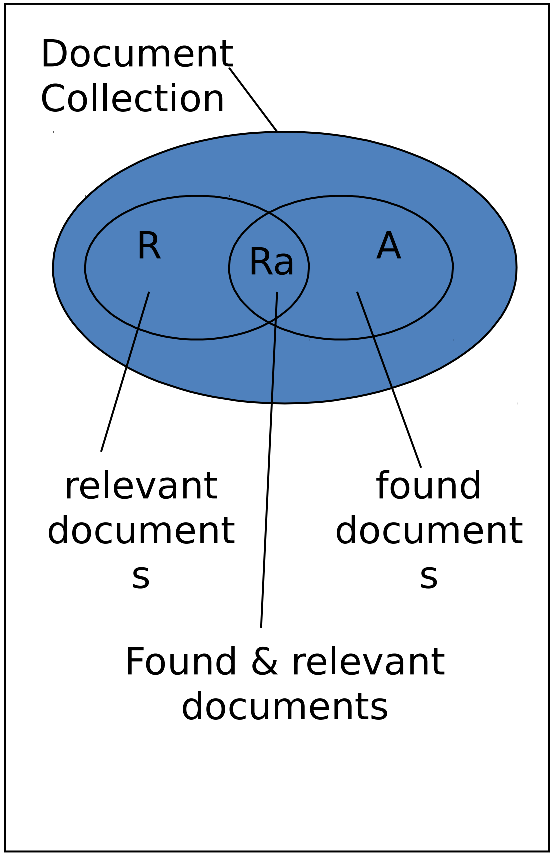
Retrieval Evaluation

- 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
- |Ra| .. 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

Example

- D = {D00, D01, … 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

F-Measure

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

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=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)

- 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)

- 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

MAP

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

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

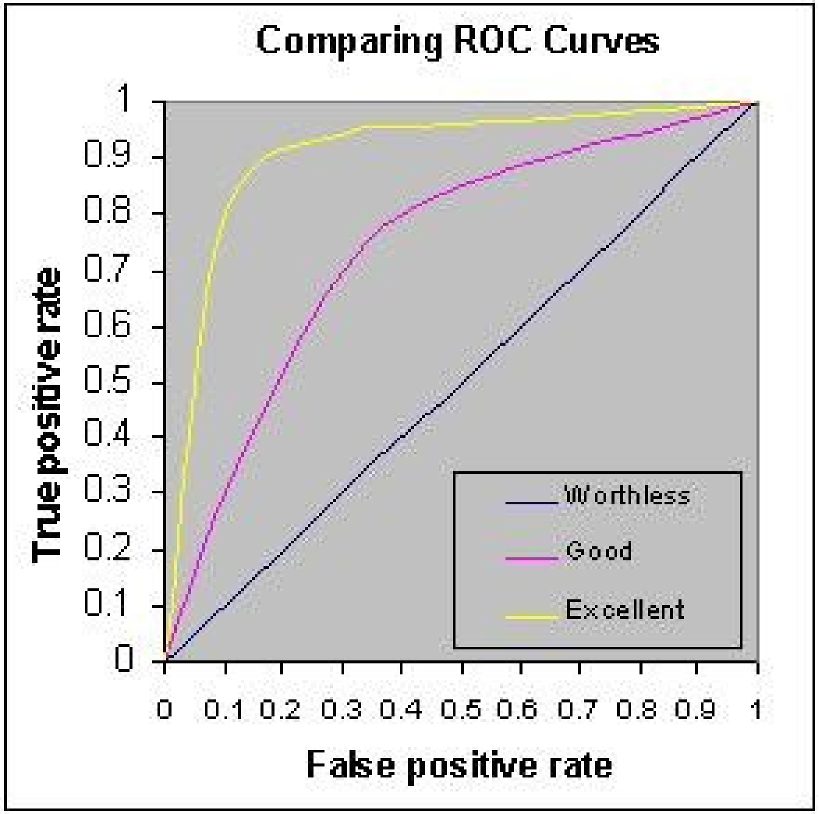
True/False Positives/Negatives

		Ground Truth	
		Pertinent	Non Pertinent
System	Pertinent	True Positive (TP)	False positive (FP)
	Non Pertinent	False Negative (FN)	True Negative (TN)

Recall = TP / (TP + FN)
False Positive Rate: FP% = FP / (FP + TN)

Receive Operator Characteristics (ROC) Curve: Recall vs. FP%

Area Under (ROC) Curve



http://gim.unmc.edu/dxtests/roc3.htm

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, ...

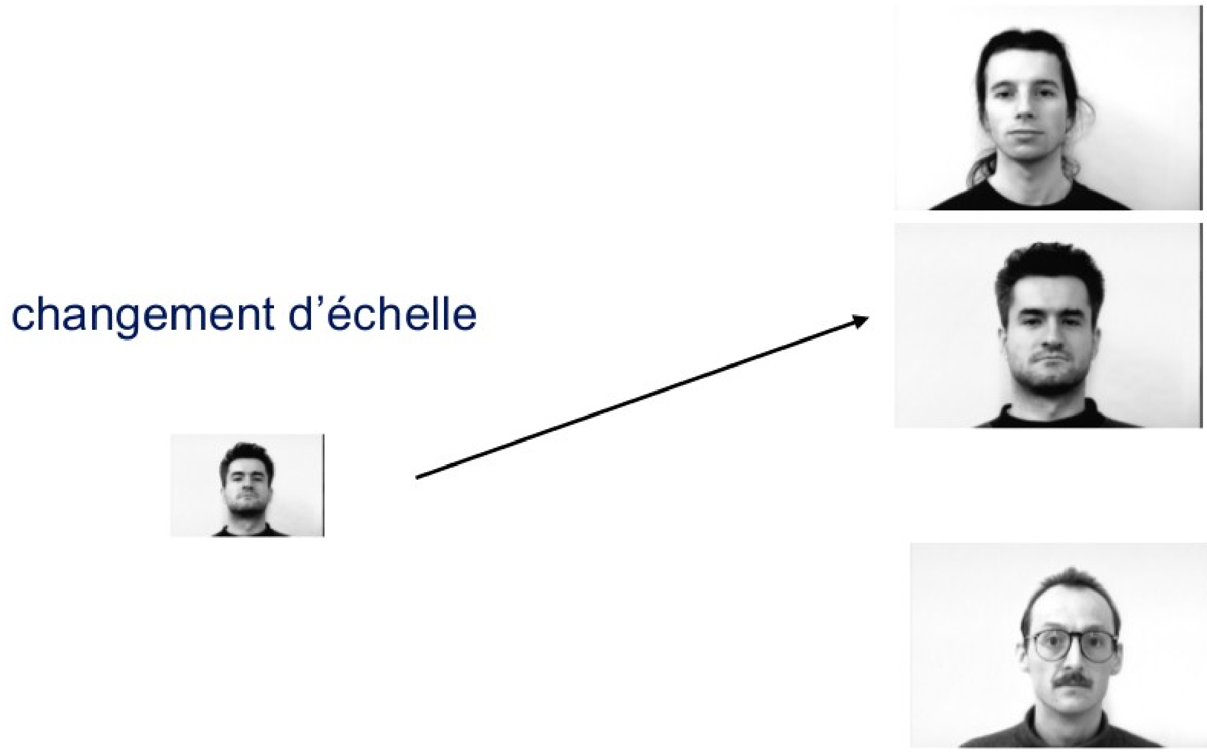
Preparation for labs

- Build your own image dataset
  - 50 images
  - 10 queries
  - Ground Truth for each query
  - Depict and explain in your report

- Be mindful of the challenges in image retrieval

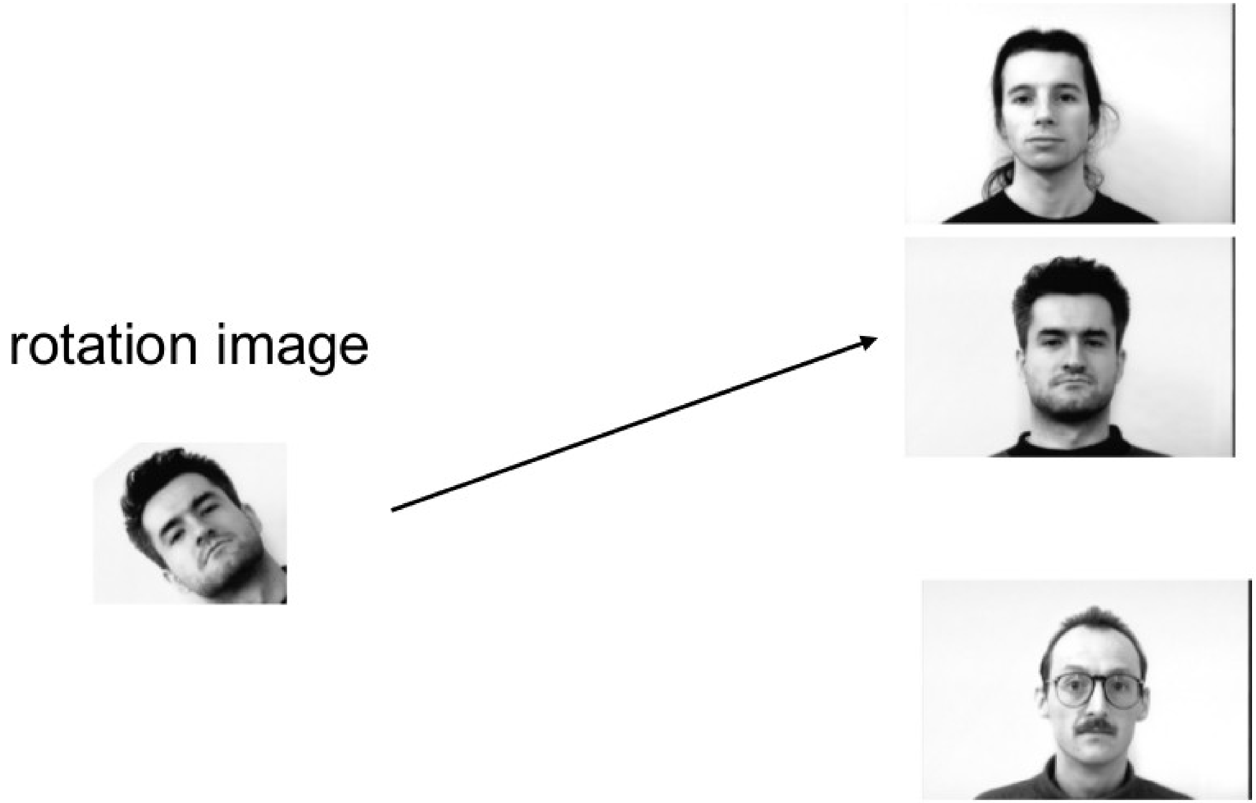
Challenges

- Scaling



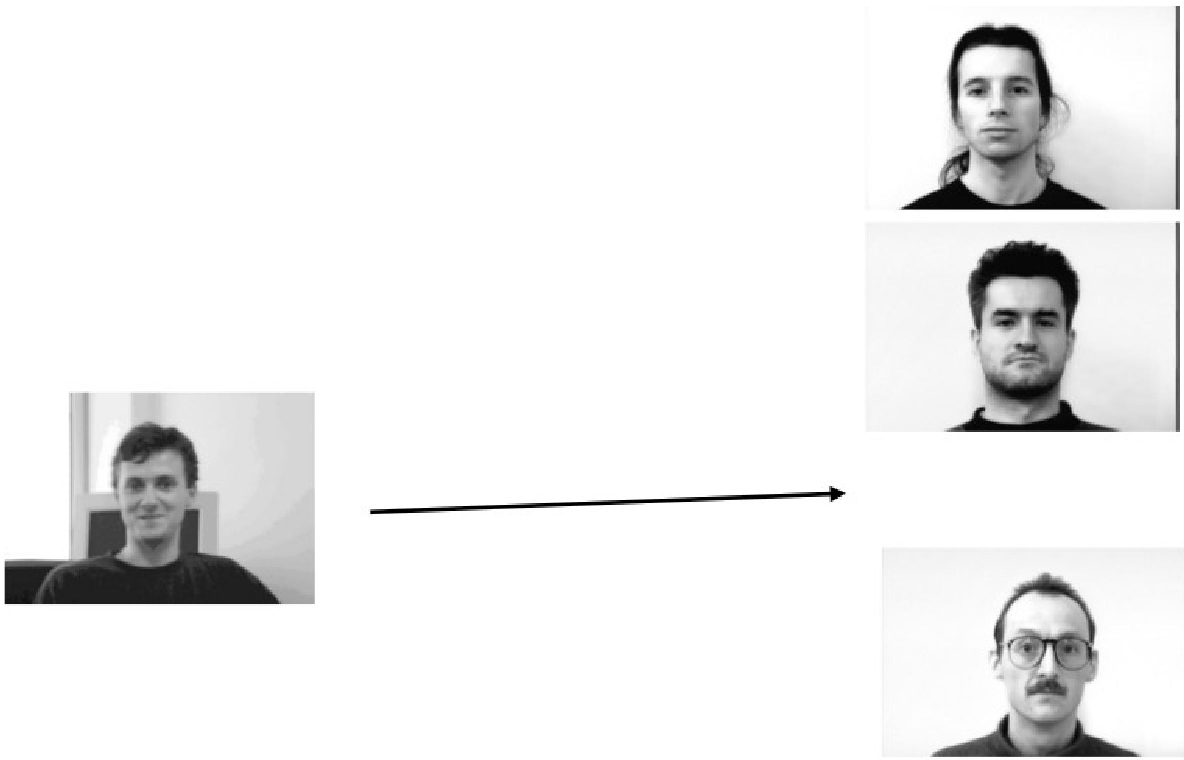
Challenges

- Rotation



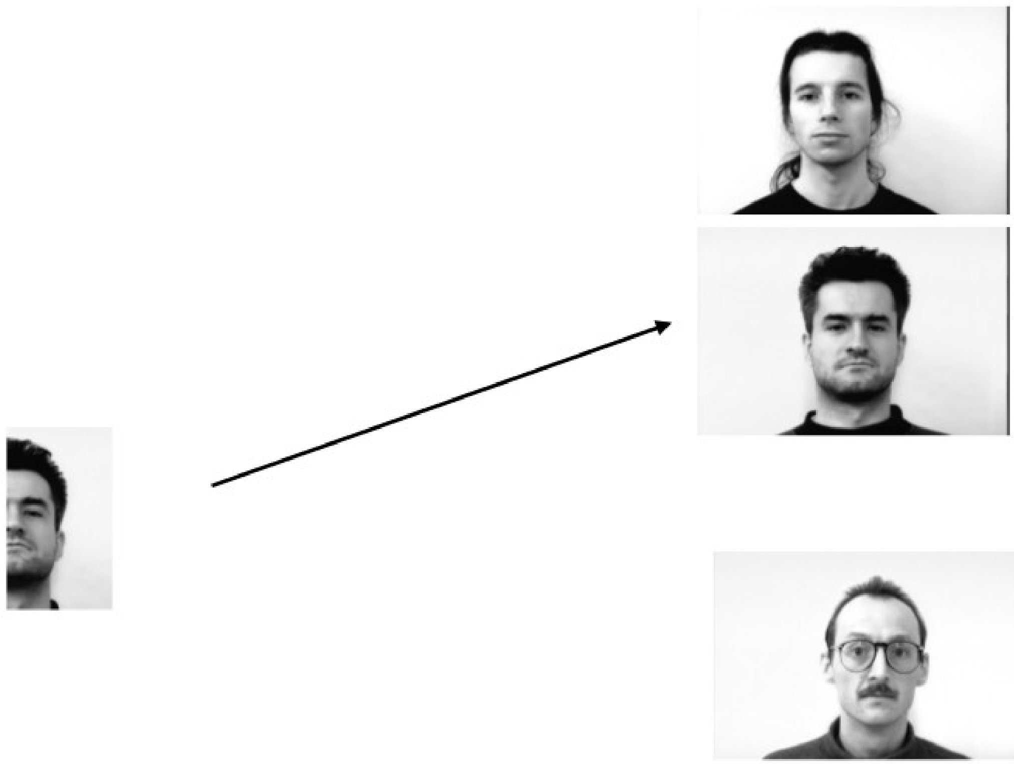
Challenges

- Clutter



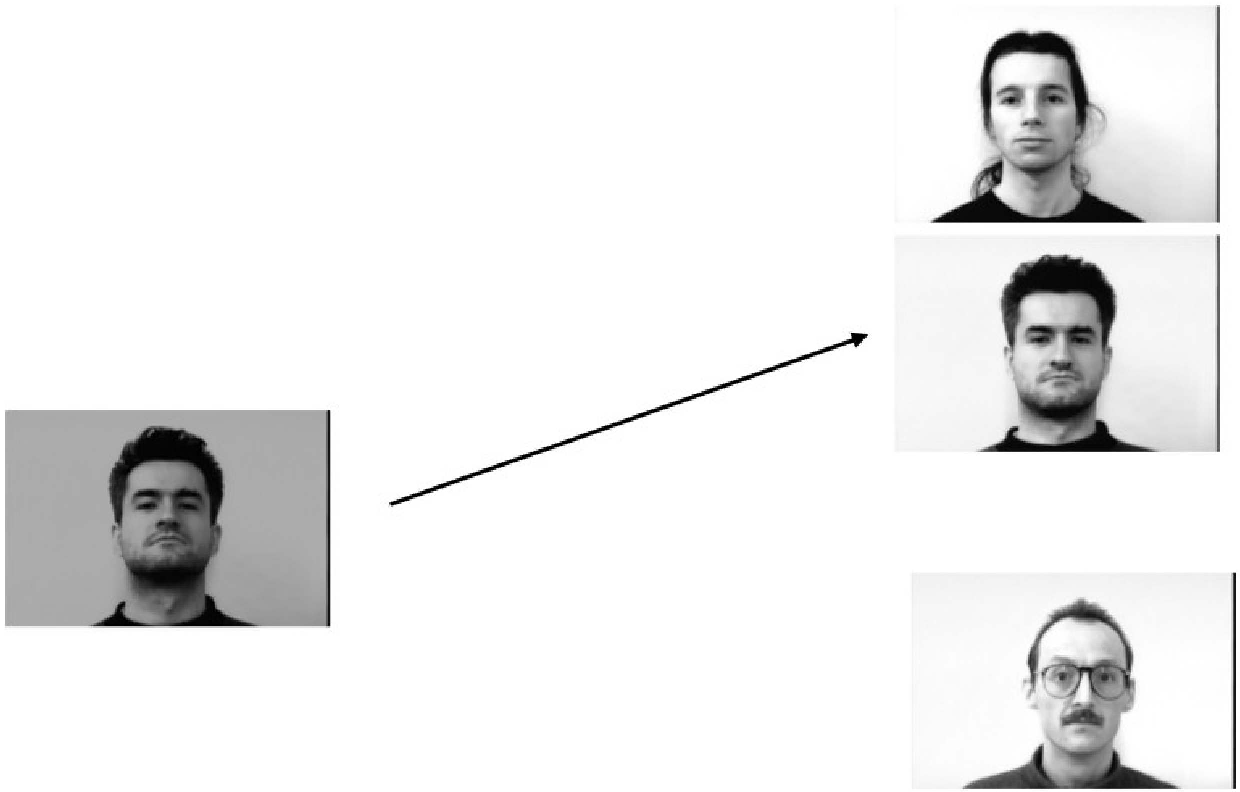
Challenges

- Occlusion



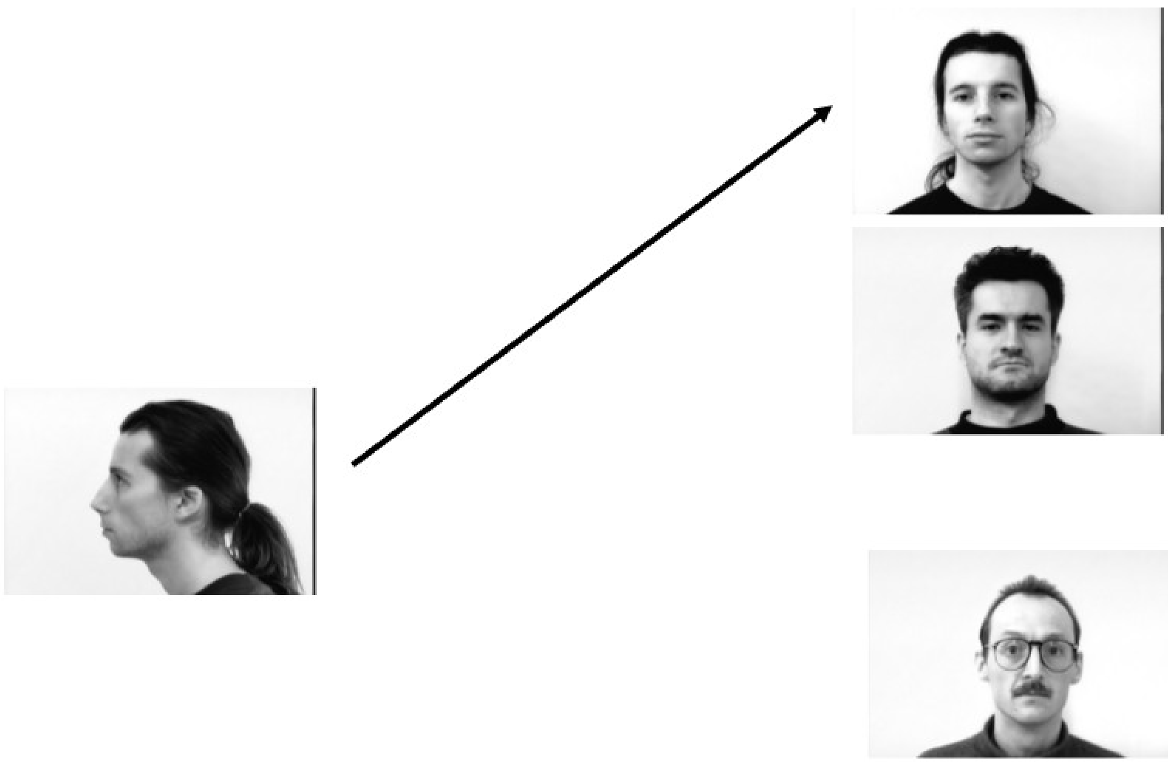
Challenges

- Lightning



Challenges

- 3D objects



Challenges

- Lightning