# VK Multimedia Information Systems

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

# Information Retrieval Basics: Agenda

- **Vector Retrieval Model**
  - Exercise 01
- Other Retrieval Models
- Common Retrieval Methods
  - Query Modification
  - Co-Occurrence
  - Relevance Feedback
- Exercise 02

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

# Vector Model

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

$$\vec{d}_j = (w_{1,j}, w_{2,j}, \ldots, w_{t,j})$$

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

# Vector model: Similarity

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

# Vector Model: Example

$$\vec{d} = (0.3, 0.1, 0, 0.1, 1)$$

$$\vec{q} = (1, 0, 0, 0.5, 0)$$

$$\mathrm{sim}(\vec{d}, \vec{q}) = \frac{1 \cdot 0.3 + 0.1 \cdot 0.5}{\sqrt{0.3^2 + 0.1^2 + 0.1^2 + 1} \cdot \sqrt{1 + 0.5^2}} \approx \frac{0.35}{2.24} \approx 0.17$$

# Another Example:

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

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

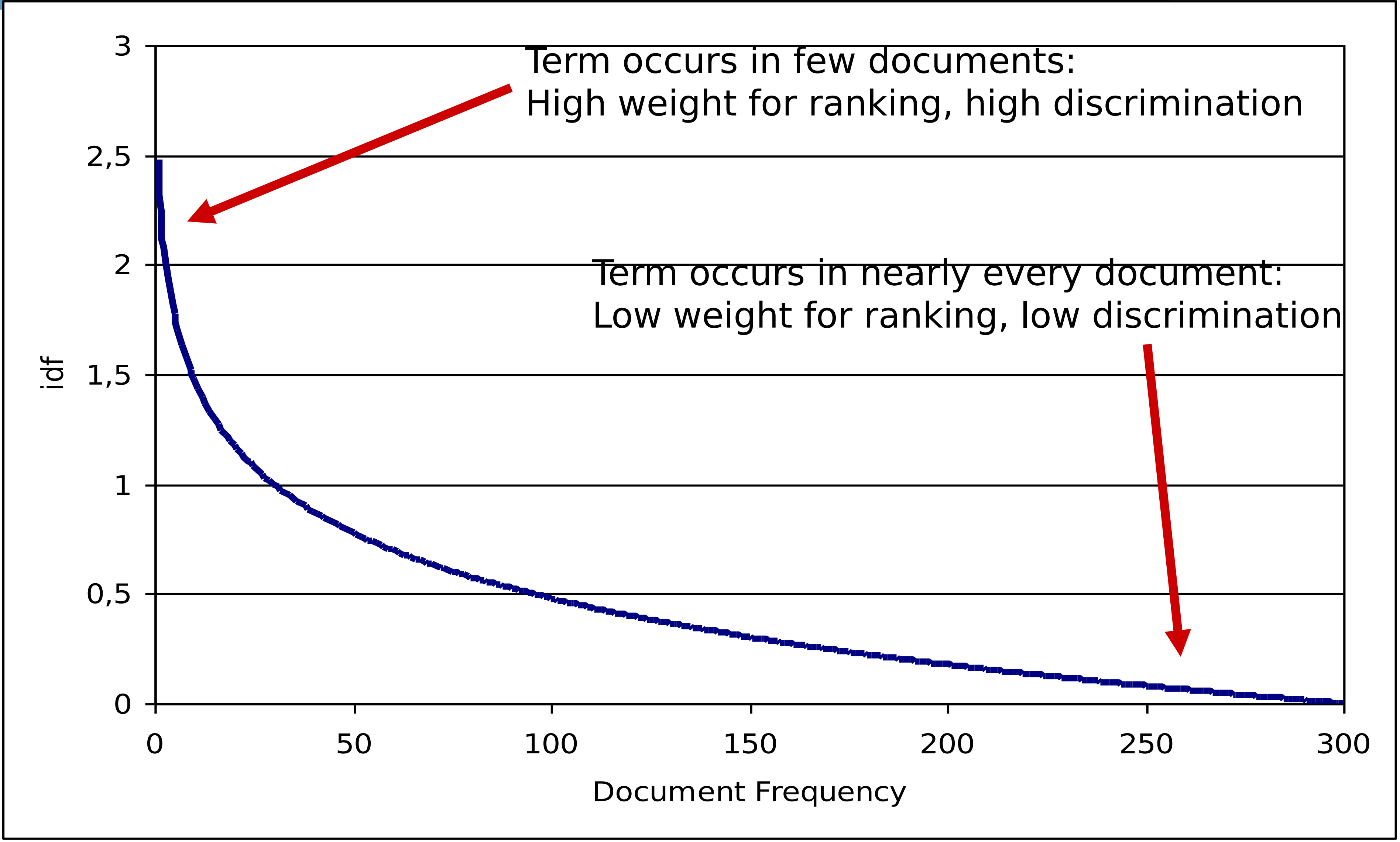
- Results:
  - $(1,1,1,1,1,1,1,1,1)^t * (1,1,1,0,0,0,0,0,0)^t = 3$
  - sqrt(9) * sqrt(3) = 5,196
  - Similarity = 3 / 5,196 = 0,577

# 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

$$f_{i,j} = \frac{freq_{i,j}}{\max_l(freq_{l,j})}$$ ... 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*

# Definitions: Inverse Document Frequency

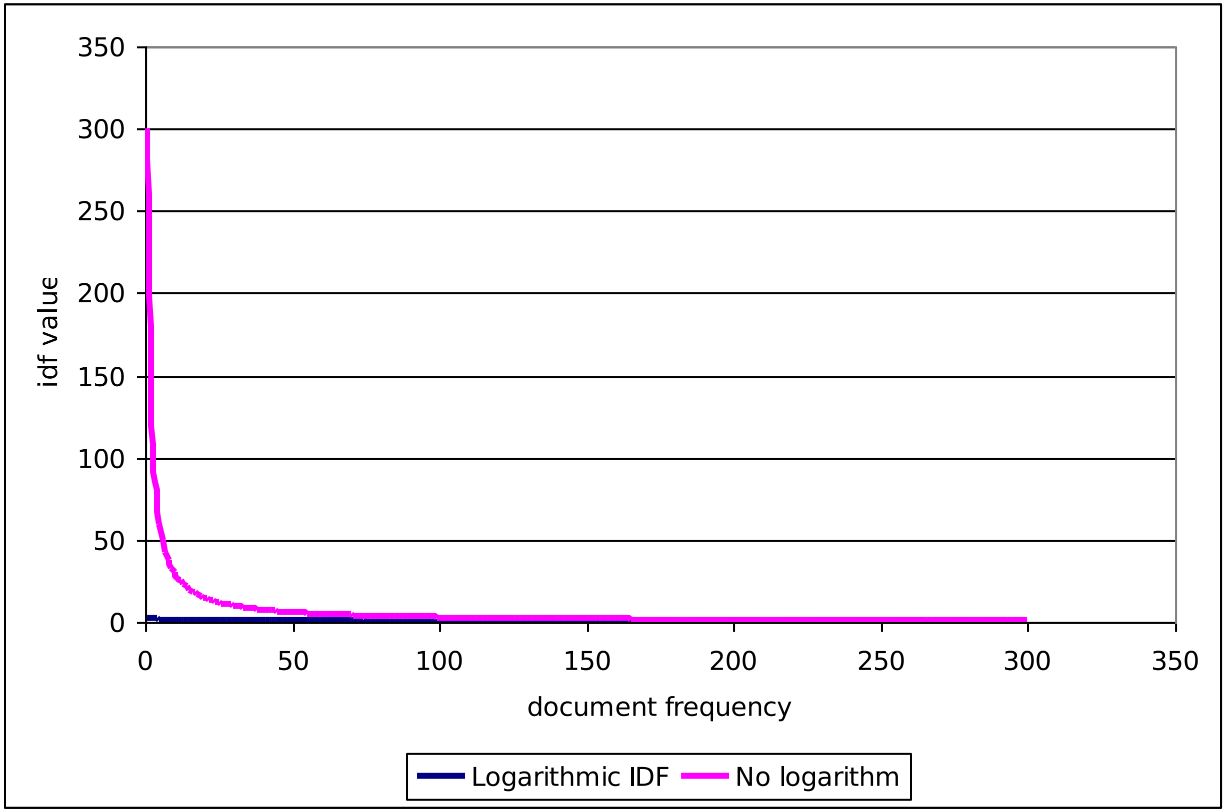
$$idf_i = \log \frac{N}{n_i} \text{ ... inverse document frequency for term } i$$

$N$ ... 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

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

# Weighting of query terms

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

- 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
  - Cosine coefficient allows **ranked list** output
- Disadvantages
  - Term are assumed to be mutually independent

# 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.

# Simple example (ii)

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

# Information Retrieval Basics: Agenda

- Vector Retrieval Model
  - **Exercise 01**
- Other Retrieval Models
- Common Retrieval Methods
  - Query Modification
  - Co-Occurrence
  - Relevance Feedback
- Exercise 03

# Exercise 01

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

- Some hints:
  - Excel:
    - Sheet on homepage
    - Use functions “Summenprodukt” & “Quadratesumme”

# Exercise 01

- Document collection (6 documents)
  - spatz, amsel, vogel, drossel, fink, falke, flug
  - spatz, vogel, flug, nest, amsel, amsel, amsel
  - kuckuck, nest, nest, ei, ei, ei, flug, amsel, amsel, vogel
  - amsel, elster, elster, drossel, vogel, ei
  - falke, katze, nest, nest, flug, vogel
  - spatz, spatz, konstruktion, nest, ei
- Queries:
  - spatz, vogel, nest, konstruktion
  - amsel, ei, nest

# Exercise

| | d1 | d2 | d3 | d4 | d6 | d6 | idf |
|---|---|---|---|---|---|---|---|
| **amsel** | 1 | 3 | 2 | 1 | | | |
| **drossel** | 1 | | | 1 | | | |
| **ei** | | | 3 | 1 | | 1 | |
| **elster** | | | | 2 | | | |
| **falke** | 1 | | | | 1 | | |
| **fink** | 1 | | | | | | |
| **flug** | 1 | 1 | 1 | | 1 | | |
| **katze** | | | | | 1 | | |
| **konstruktion** | | | | | | 1 | |
| **kuckuck** | | | 1 | | | | |
| **nest** | | 1 | 2 | | 2 | 1 | |
| **spatz** | 1 | 1 | | | | 2 | |
| **vogel** | 1 | 1 | 1 | 1 | 1 | | |

# Information Retrieval Basics: Agenda

- Vector Retrieval Model
  - Exercise 01
- **Other Retrieval Models**
- Common Retrieval Methods
  - Query Modification
  - Co-Occurrence
  - Relevance Feedback
- Exercise 02

# Other Retrieval Models: Set Theoretic Models

- Fuzzy Set Model
  - Each query term defines a fuzzy set
  - Each document has a **degree of membership**
  - Done e.g. with query expansion (co-occurrence or thesaurus)
- Extended Boolean Model
  - Incorporates non binary weights
  - Geometric interpretation: Distance between document vector and desired Boolean state (query)

# Algebraic Models: Generalized Vector Space M.

- Term independence not necessary
- Terms (as dimensions) are not orthogonal and may be linear dependent.
- Smaller linear independent units exist.
  - m … minterm
  - Constructed from co-occurrence: $2^t$ minterms
- Dimensionality a problem
  - Number of active minterms (which actually occur in a document)
  - Depends on the number of documents

# Algebraic Models: Latent Semantic Indexing M.

- Introduced 1988, LSI / LSA
- Concept matching vs. term matching
- Mapping documents & terms to concept space:
  - Fewer dimensions
  - Like clustering

# Algebraic Models: Latent Semantic Indexing M.

- Let $M_{ij}$ be the document term matrix
  - with $t$ rows (terms) and $N$ cols (docs)
- Decompose $M_{ij}$ into $K*S*D^t$
  - $K$ .. matrix of eigenvectors from term-to-term (co-occurence) matrix
  - $D^t$ .. matrix of eigenvectors from doc-to-doc matrix
  - $S$ .. $r$ x $r$ diagonal matrix of singular values with $r=min(t,N)$, the rank of $M_{ij}$

# Algebraic Models: Latent Semantic Indexing M.

- With $M_{ij} = K{*}S{*}D^t$ …
- Only the $s$ largest singular values from $S$:
  - Others are deleted
  - Respective columns in $K$ and $D^t$ remain
- $M_s = K_s{*}S_s{*}D^t_s$ …
  - $s < r$ is new rank of $M$
  - $s$ large enough to fit in all data
  - $s$ small enough to cut out unnecessary details

# Algebraic Models: Latent Semantic Indexing M.

- Reduced doc-to-doc matrix:
  - $M^t{}_s * M_s$ is $N$x$N$ Matrix quantifying the relationship between documents
- Retrieval is based on pseudo-document
  - Let column *0* in $M_{ij}$ be the query
  - Calculate $M^t{}_s * M_s$
  - First row (or column) gives the relevance

# Algebraic Models: Latent Semantic Indexing M.

- Advantages
  - *M* even more sparse
  - Retrieval on a “conceptual” level
- Disadvantages
  - Doc-to-doc matrix might be quite big
  - Therefore: Processing time

# Example LSA …

Example of text data: Titles of Some Technical Memos

| | |
|---|---|
| c1: | *Human* machine *interface* for ABC *computer* applications |
| c2: | A *survey* of *user* opinion of *computer system response time* |
| c3: | The *EPS user interface* management *system* |
| c4: | *System* and *human system* engineering testing of *EPS* |
| c5: | Relation of *user* perceived *response time* to error measurement |
| m1: | The generation of random, binary, ordered *trees* |
| m2: | The intersection *graph* of paths in *trees* |
| m3: | *Graph minors* IV: Widths of *trees* and well-quasi-ordering |
| m4: | *Graph minors*: A *survey* |

*from Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. Discourse Processes, 25, 259-284.*

# Example LSA ...

$$\{X\} =$$

| | c1 | c2 | c3 | c4 | c5 | m1 | m2 | m3 | m4 |
|---|---|---|---|---|---|---|---|---|---|
| human | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| interface | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| computer | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| user | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| system | 0 | 1 | 1 | 2 | 0 | 0 | 0 | 0 | 0 |
| response | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| time | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| EPS | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| survey | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| trees | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| graph | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| minors | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

*from Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. Discourse Processes, 25, 259-284.*

# Example LSA ...

$\{W\} =$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.22 | -0.11 | 0.29 | -0.41 | -0.11 | -0.34 | 0.52 | -0.06 | -0.41 |
| 0.20 | -0.07 | 0.14 | -0.55 | 0.28 | 0.50 | -0.07 | -0.01 | -0.11 |
| 0.24 | 0.04 | -0.16 | -0.59 | -0.11 | -0.25 | -0.30 | 0.06 | 0.49 |
| 0.40 | 0.06 | -0.34 | 0.10 | 0.33 | 0.38 | 0.00 | 0.00 | 0.01 |
| 0.64 | -0.17 | 0.36 | 0.33 | -0.16 | -0.21 | -0.17 | 0.03 | 0.27 |
| 0.27 | 0.11 | -0.43 | 0.07 | 0.08 | -0.17 | 0.28 | -0.02 | -0.05 |
| 0.27 | 0.11 | -0.43 | 0.07 | 0.08 | -0.17 | 0.28 | -0.02 | -0.05 |
| 0.30 | -0.14 | 0.33 | 0.19 | 0.11 | 0.27 | 0.03 | -0.02 | -0.17 |
| 0.21 | 0.27 | -0.18 | -0.03 | -0.54 | 0.08 | -0.47 | -0.04 | -0.58 |
| 0.01 | 0.49 | 0.23 | 0.03 | 0.59 | -0.39 | -0.29 | 0.25 | -0.23 |
| 0.04 | 0.62 | 0.22 | 0.00 | -0.07 | 0.11 | 0.16 | -0.68 | 0.23 |
| 0.03 | 0.45 | 0.14 | -0.01 | -0.30 | 0.28 | 0.34 | 0.68 | 0.18 |

$\{S\} =$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 3.34 | | | | | | | | |
| | 2.54 | | | | | | | |
| | | 2.35 | | | | | | |
| | | | 1.64 | | | | | |
| | | | | 1.50 | | | | |
| | | | | | 1.31 | | | |
| | | | | | | 0.85 | | |
| | | | | | | | 0.56 | |
| | | | | | | | | 0.36 |

$\{P\} =$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.20 | 0.61 | 0.46 | 0.54 | 0.28 | 0.00 | 0.01 | 0.02 | 0.08 |
| -0.06 | 0.17 | -0.13 | -0.23 | 0.11 | 0.19 | 0.44 | 0.62 | 0.53 |
| 0.11 | -0.50 | 0.21 | 0.57 | -0.51 | 0.10 | 0.19 | 0.25 | 0.08 |
| -0.95 | -0.03 | 0.04 | 0.27 | 0.15 | 0.02 | 0.02 | 0.01 | -0.03 |
| 0.05 | -0.21 | 0.38 | -0.21 | 0.33 | 0.39 | 0.35 | 0.15 | -0.60 |
| -0.08 | -0.26 | 0.72 | -0.37 | 0.03 | -0.30 | -0.21 | 0.00 | 0.36 |
| 0.18 | -0.43 | -0.24 | 0.26 | 0.67 | -0.34 | -0.15 | 0.25 | 0.04 |
| -0.01 | 0.05 | 0.01 | -0.02 | -0.06 | 0.45 | -0.76 | 0.45 | -0.07 |
| -0.06 | 0.24 | 0.02 | -0.08 | -0.26 | -0.62 | 0.02 | 0.52 | -0.45 |

# Example LSA ...

| | c1 | c2 | c3 | c4 | c5 | m1 | m2 | m3 | m4 |
|---|---|---|---|---|---|---|---|---|---|
| human | 0.16 | 0.40 | 0.38 | 0.47 | 0.18 | -0.05 | -0.12 | -0.16 | -0.09 |
| interface | 0.14 | 0.37 | 0.33 | 0.40 | 0.16 | -0.03 | -0.07 | -0.10 | -0.04 |
| computer | 0.15 | 0.51 | 0.36 | 0.41 | 0.24 | 0.02 | 0.06 | 0.09 | 0.12 |
| user | 0.26 | 0.84 | 0.61 | 0.70 | 0.39 | 0.03 | 0.08 | 0.12 | 0.19 |
| system | 0.45 | 1.23 | 1.05 | 1.27 | 0.56 | -0.07 | -0.15 | -0.21 | -0.05 |
| response | 0.16 | 0.58 | 0.38 | 0.42 | 0.28 | 0.06 | 0.13 | 0.19 | 0.22 |
| time | 0.16 | 0.58 | 0.38 | 0.42 | 0.28 | 0.06 | 0.13 | 0.19 | 0.22 |
| EPS | 0.22 | 0.55 | 0.51 | 0.63 | 0.24 | -0.07 | -0.14 | -0.20 | -0.11 |
| survey | 0.10 | 0.53 | 0.23 | 0.21 | 0.27 | 0.14 | 0.31 | 0.44 | 0.42 |
| trees | -0.06 | 0.23 | -0.14 | -0.27 | 0.14 | 0.24 | 0.55 | 0.77 | 0.66 |
| graph | -0.06 | 0.34 | -0.15 | -0.30 | 0.20 | 0.31 | 0.69 | 0.98 | 0.85 |
| minors | -0.04 | 0.25 | -0.10 | -0.21 | 0.15 | 0.22 | 0.50 | 0.71 | 0.62 |

| | c1 | c2 | c3 | c4 | c5 | m1 | m2 | m3 | m4 |
|---|---|---|---|---|---|---|---|---|---|
| human | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| interface | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| computer | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| user | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| system | 0 | 1 | 1 | 2 | 0 | 0 | 0 | 0 | 0 |
| response | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| time | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| EPS | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| survey | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| trees | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| graph | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| minors | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

# Example LSA …

Correlations between titles in raw data:

| | c1 | c2 | c3 | c4 | c5 | m1 | m2 | m3 |
|---|---|---|---|---|---|---|---|---|
| c2 | -0.19 | | | | | | | |
| c3 | 0.00 | 0.00 | | | | | | |
| c4 | 0.00 | 0.00 | 0.47 | | | | | |
| c5 | -0.33 | 0.58 | 0.00 | -0.31 | | | | |
| m1 | -0.17 | -0.30 | -0.21 | -0.16 | -0.17 | | | |
| m2 | -0.26 | -0.45 | -0.32 | -0.24 | -0.26 | 0.67 | | |
| m3 | -0.33 | -0.58 | -0.41 | -0.31 | -0.33 | 0.52 | 0.77 | |
| m4 | -0.33 | -0.19 | -0.41 | -0.31 | -0.33 | -0.17 | 0.26 | 0.56 |

| | |
|---|---|
| 0.02 | |
| -0.30 | 0.44 |

Correlations in two dimensional space:

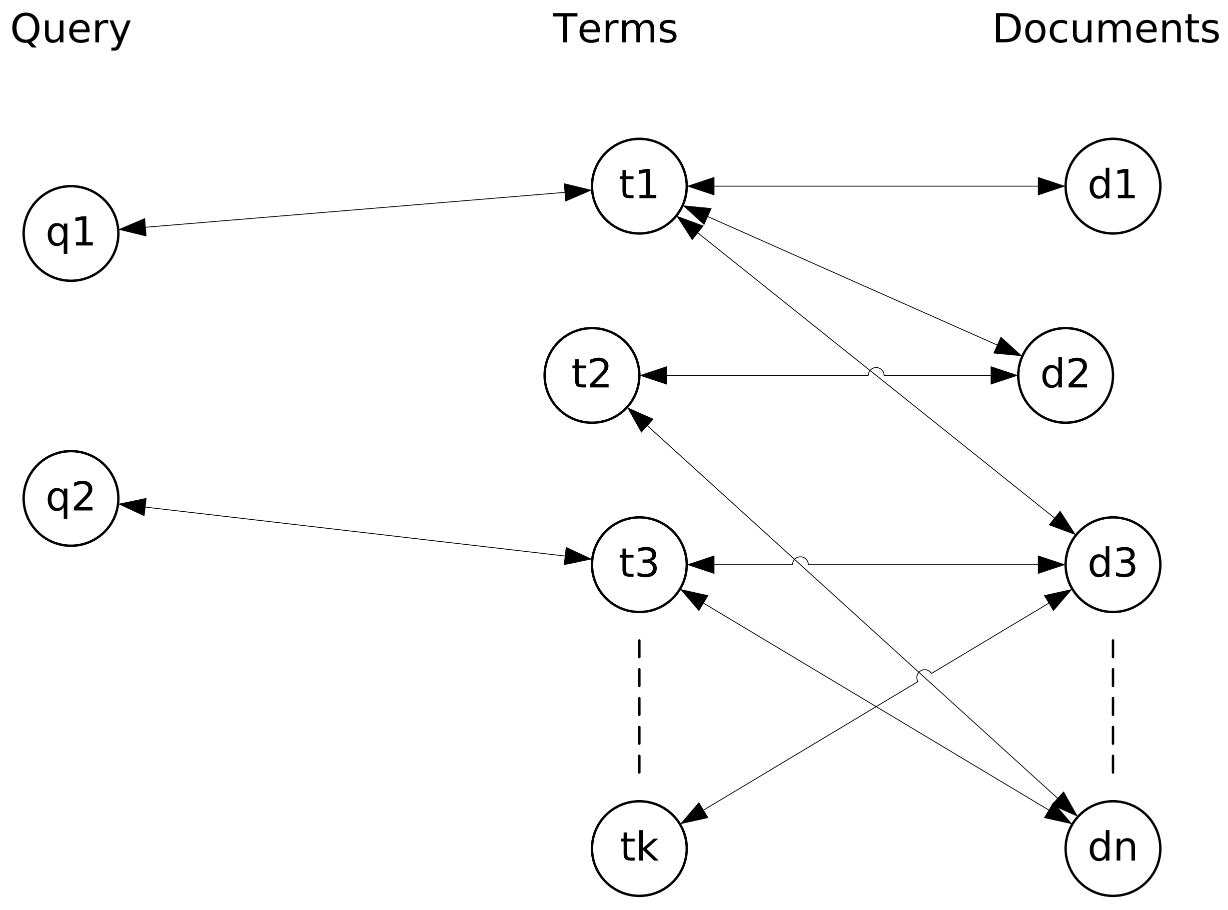
| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| c2 | 0.91 | | | | | | | |
| c3 | 1.00 | 0.91 | | | | | | |
| c4 | 1.00 | 0.88 | 1.00 | | | | | |
| c5 | 0.85 | 0.99 | 0.85 | 0.81 | | | | |
| m1 | -0.85 | -0.56 | -0.85 | -0.88 | -0.45 | | | |
| m2 | -0.85 | -0.56 | -0.85 | -0.88 | -0.44 | 1.00 | | |
| m3 | -0.85 | -0.56 | -0.85 | -0.88 | -0.44 | 1.00 | 1.00 | |
| m4 | -0.81 | -0.50 | -0.81 | -0.84 | -0.37 | 1.00 | 1.00 | 1.00 |

| | |
|---|---|
| 0.92 | |
| -0.72 | 1.00 |

# Algebraic Models: Neural Network M. / Associative Retrieval

- Neural Network:
  - Neurons emit signals to other neurons
  - Graph interconnected by synaptic connections
- Three levels:
  - Query terms, terms & documents

# Algebraic Models: Neural Network M. / Associative Retrieval



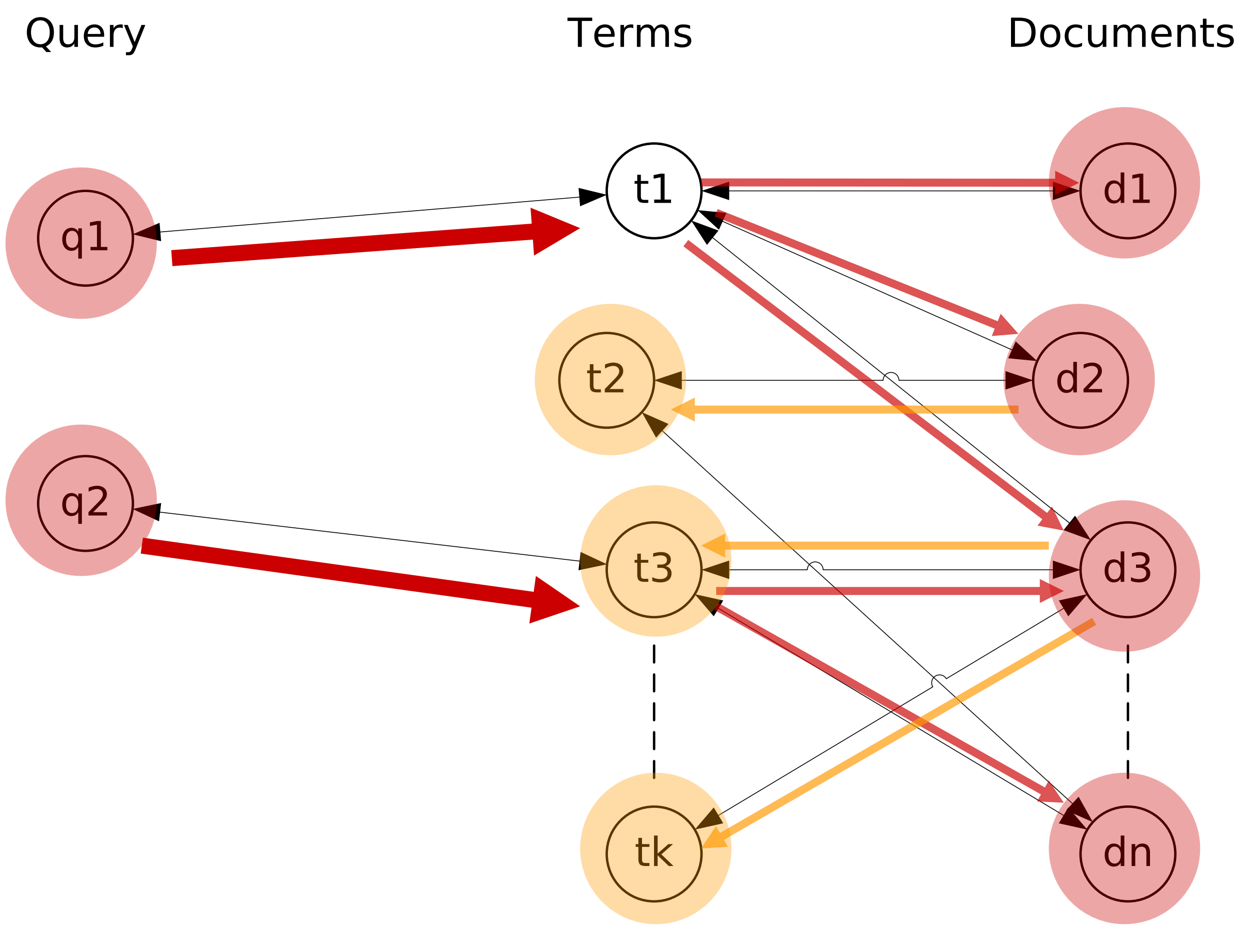
# Algebraic Models: Neural Network M. / Associative Retrieval

- Query term is "activated"
  - Usually with weight 1
  - Query term weight is used to "weaken" the signal
- Connected terms receive signal
  - Term weight "weakens" the signal
- Connected documents receive signal
  - Different activation sources are "combined"

# Algebraic Models: Neural Network M. / Associative Retrieval

- First round query terms -> terms -> docs
  - Equivalent to vector model
- Further rounds increase retrieval performance

# Algebraic Models: Neural Network M. / Associative Retrieval



# Information Retrieval Basics: Agenda

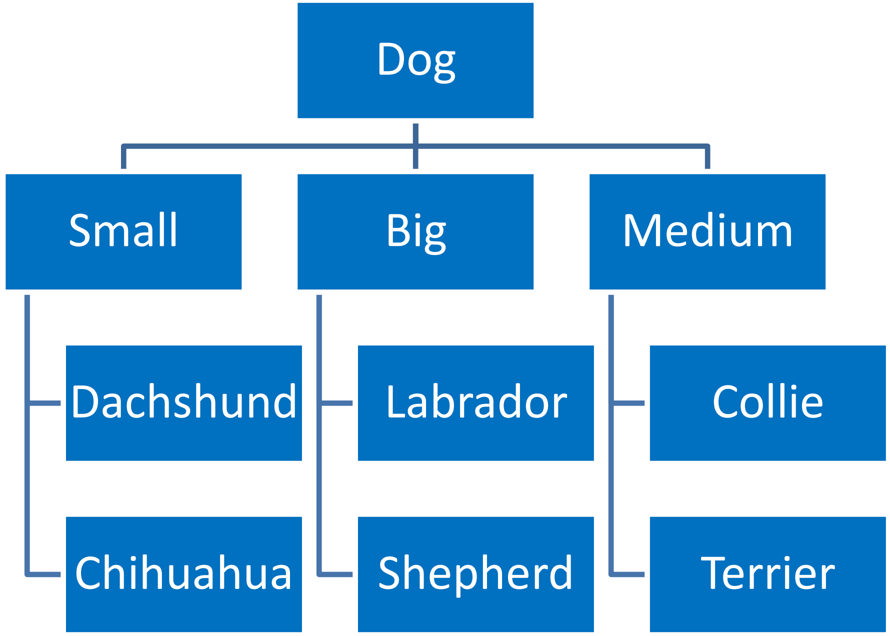
- Vector Retrieval Model
  - Exercise 02
- Other Retrieval Models
- Common Retrieval Methods
  - **Query Modification**
  - Co-Occurrence
  - Relevance Feedback
- Exercise 03

# Query Modification

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

# Query Expansion

- Integrate existing knowledge
  - Taxonomies
  - Ontologies
- Modify query
  - Related terms
  - Narrower terms
  - Broader terms



# Term Reweighting

- To improve accuracy of ranking
- Query term weights are changed
  - Note: no terms are added / removed
  - Result ranking changes

# Information Retrieval Basics: Agenda

- Vector Retrieval Model
  - Exercise 02
- Other Retrieval Models
- Common Retrieval Methods
  - Query Modification
  - **Co-Occurrence**
  - Relevance Feedback
- Exercise 03

# Co-Occurrence

- Quantify relations between terms
  - Based on how often they occur together
  - Not based on the position
- Let $M_{ij}$ be the document term matrix
  - with $t$ rows (terms) and $N$ cols (docs)
- $M*M^t$ is the “co-occurrence” matrix

# Co-Occurrence: Example

| | d1 | d2 | d3 | d4 | d5 |
|---|---|---|---|---|---|
| **computer** | 7 | 7 | 0 | 8 | 3 |
| **pda** | 5 | 1 | 4 | 0 | 3 |
| **cellphone** | 0 | 1 | 5 | 0 | 0 |
| **wlan** | 6 | 1 | 0 | 0 | 4 |
| **network** | 1 | 2 | 0 | 6 | 0 |

| | | | | |
|---|---|---|---|---|
| 7 | 5 | 0 | 6 | 1 |
| 7 | 1 | 1 | 1 | 2 |
| 0 | 4 | 5 | 0 | 0 |
| 8 | 0 | 0 | 0 | 6 |
| 3 | 3 | 0 | 4 | 0 |

# Co-Occurrence: Example

| | computer | pda | cellphone | wlan | network |
|---|---|---|---|---|---|
| **computer** | 171 | 51 | 7 | 61 | 69 |
| **pda** | 51 | 51 | 21 | 43 | 7 |
| **cellphone** | 7 | 21 | 26 | 1 | 2 |
| **wlan** | 61 | 43 | 1 | 53 | 8 |
| **network** | 69 | 7 | 2 | 8 | 41 |

# Co-Occurrence & Query Expansion

| | computer | pda | cellphone | wlan | network |
|---|---|---|---|---|---|
| **computer** | 171 | 51 | 7 | 61 | 69 |
| **pda** | 51 | 51 | 21 | 43 | 7 |
| **cellphone** | 7 | 21 | 26 | 1 | 2 |
| **wlan** | 61 | 43 | 1 | 53 | 8 |
| **network** | 69 | 7 | 2 | 8 | 41 |

Query: *cellphone*

Query: *cellphone OR pda*

# Information Retrieval Basics: Agenda

- Vector Retrieval Model
  - Exercise 02
- Other Retrieval Models
- Common Retrieval Methods
  - Query Modification
  - Co-Occurrence
  - **Relevance Feedback**
- Exercise 03

# 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

# Relevance Feedback

- $D_r \subset D$... set of relevant docs identified by the user
- $D_n \subset D$ ... set of non relevant docs
- $C_r \subset D$ ... set of relevant docs
- $\alpha$, $\beta$, $\gamma$ ... tuning parameters

# Relevance Feedback

- Considering an optimal query
  - Unlikely and therefore hypothetical
- Which vector retrieves $C_r$ best?

$$\vec{q}_{OPT} = \frac{1}{|C_r|} \cdot \sum_{\forall \vec{d}_j \in C_r} \vec{d}_j - \frac{1}{N - |C_r|} \cdot \sum_{\forall \vec{d}_j \notin C_r} \vec{d}_j$$

# Relevance Feedback

$$\text{Rochio: } \vec{q}_m = \alpha \cdot \vec{q} + \frac{\beta}{|D_r|} \cdot \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \frac{\gamma}{|D_n|} \cdot \sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$$

$$\text{Ide: } \vec{q}_m = \alpha \cdot \vec{q} + \beta \cdot \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \gamma \cdot \sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$$

$$\text{Ide-Dec-Hi: } \vec{q}_m = \alpha \cdot \vec{q} + \beta \cdot \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \gamma \max_{non-relevant}(\vec{d}_j)$$

# Relevance Feedback

- Rochio
  - Based on $q_{OPT}$, $\alpha$ was 1 in original idea
- Ide
  - $\alpha=\beta=\gamma=1$ in original idea
- Ide-Dec-Hi
  - $\max_{\text{non-relevant}}$ … highest ranked doc of $D_n$

- All three techniques yield similar results …

# Relevance Feedback

- Evaluation issues:
  - Boosts retrieval performance
  - Relevant documents are ranked top
  - But: Already marked by the user
- Evaluation remains complicated issue

# Information Retrieval Basics: Agenda

- Vector Retrieval Model
  - Exercise 01
- Other Retrieval Models
- Common Retrieval Methods
  - Query Modification
  - Co-Occurrence
  - Relevance Feedback
- **Exercise 02**

ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

# Exercise 02

Install R: http://www.r-project.org/

- Co-Occurrence
  - Document-term matrix from exercise 01
    - x <- cbind(1, 3, 2, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 3, 1, 0, 1,0, 0, 0, 2, 0, 0,1, 0, 0, 0, 1, 0,1, 0, 0, 0, 0, 0,1, 1, 1, 0, 1, 0,0, 0, 0, 0, 1, 0,0, 0, 0, 0, 0, 1,0, 0, 1, 0, 0, 0,0, 1, 2, 0, 2, 1,1, 1, 0, 0, 0, 2,1, 1, 1, 1, 1, 0)
    - x <- matrix(x, ncol=6)
  - Compute term-term co-occurrence
  - Find the most 3 relevant terms for "*kuckuck*" and "*ei*"
- Apply LSA to Exercise 02 before computing the term-term co-occurrence
  - ?svd // helps with svd, %*% is matrix multiplication, use diag() for d

# Thanks …

for your attention!