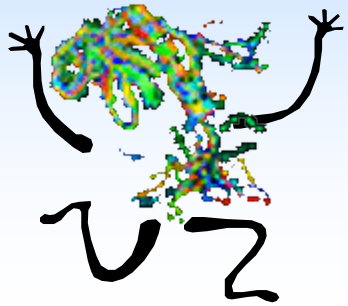
GET THAT PROTEIN!



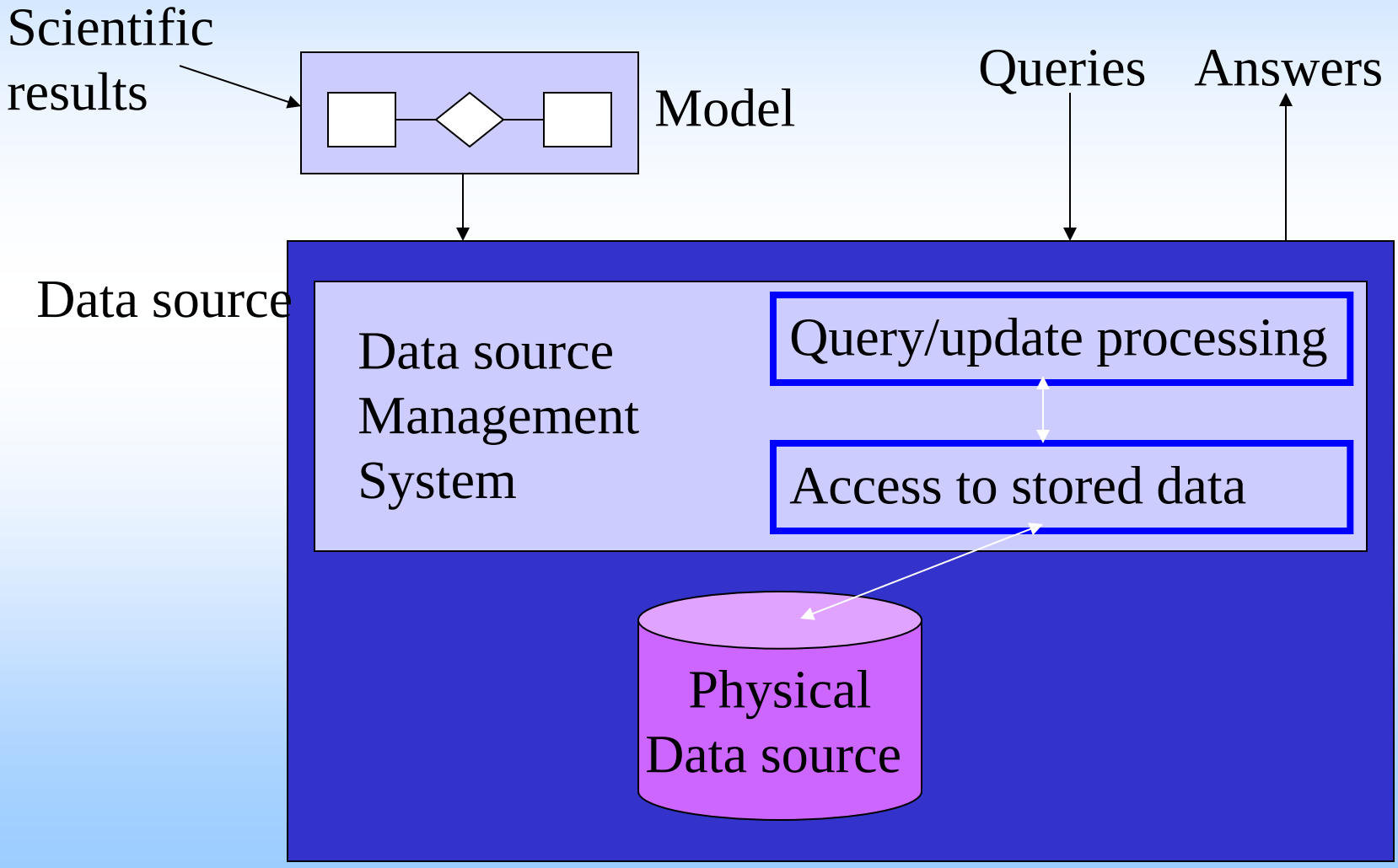
Information retrieval

Patrick Lambrix
Department of Computer and Information Science
Linköpings universitet

Electronic Data Sources

- Data in electronic form
- Used in every day life and research

Data sources



Storing and accessing textual information

- What information is stored?
- How is the information stored?
  - high level
- How is the information retrieved?

What information is stored?

- Model the information
  - Entity-Relationship model (ER)
  - Unified Modeling Language (UML)

What information is stored? - ER

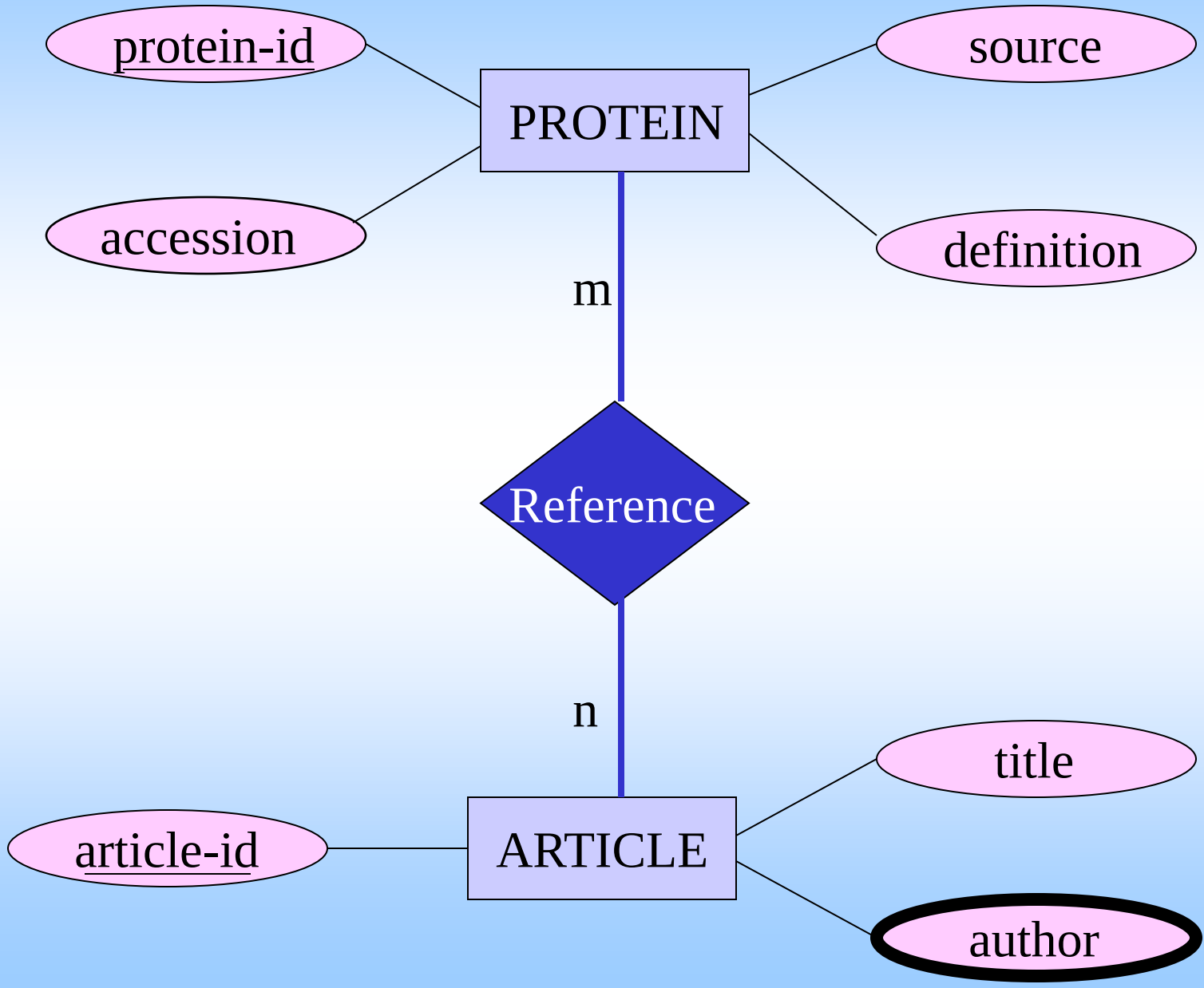
- entities and attributes
- entity types
- key attributes
- relationships
- cardinality constraints

- EER: sub-types

1 tgctacccgc gcccgggctt ctggggtgtt ccccaaccac ggcccagccc tgccacaccc
61 cccgcccccg gcctccgcag ctcggcatgg gcgcgggggt gctcgtcctg ggcgcctccg
121 agcccggtaa cctgtcgtcg gccgcaccgc tccccgacgg cgcggccacc gcggcgcggc
181 tgctggtgcc cgcgtcgccg cccgcctcgt tgctgcctcc cgccagcgaa agccccgagc
241 cgctgtctca gcagtggaca gcgggcatgg gtctgctgat ggcgctcatc gtgctgctca
301 tcgtggcggg caatgtgctg gtgatcgtgg ccatcgccaa gacgccgcgg ctgcagacgc
361 tcaccaacct cttcatcatg tccctggcca gcgccgacct ggtcatgggg ctgctggtgg
421 tgccgttcgg ggccaccatc gtggtgtggg gccgctggga gtacggctcc ttcttctgcg
481 agctgtggac ctcagtggac gtgctgtgcg tgacggccag catcgagacc ctgtgtgtca
541 ttgccctgga ccgctacctc gccatcacct cgcccttccg ctaccagagc ctgctgacgc
601 gcgcgcgggc gcggggcctc gtgtgcaccg tgtgggccat ctcggccctg gtgtccttcc
661 tgcccatcct catgcactgg tggcgggcgg agagcgacga ggcgcgccgc tgctacaacg
721 accccaagtg ctgcgacttc gtcaccaacc gggcctacgc catcgcctcg tccgtagtct
781 ccttctacgt gcccctgtgc atcatggcct tcgtgtacct gcgggtgttc cgcgaggccc
841 agaagcaggt gaagaagatc gacagctgcg agcgccgttt cctcggcggc ccagcgcggc
901 cgccctcgcc ctcgccctcg cccgtccccg cgcccgcgcc gccgcccgga cccccgcgcc
961 ccgccgccgc cgccgccacc gccccgctgg ccaacgggcg tgcgggtaag cggcggccct
1021 cgcgcctcgt ggccctacgc gagcagaagg cgctcaagac gctgggcatc atcatgggcg
1081 tcttcacgct ctgctggctg cccttcttcc tggccaacgt ggtgaaggcc ttccaccgcg
1141 agctggtgcc cgaccgcctc ttcgtcttct tcaactggct gggctacgcc aactcggcct
1201 tcaaccccat catctactgc cgcagccccg acttccgcaa ggccttccag ggactgctct
1261 gctgcgcgcg cagggctgcc cgccggcgcc acgcgaccca cggagaccgg ccgcgcgcct
1321 cgggctgtct ggcccggccc ggacccccgc catcgcccgg ggccgcctcg gacgacgacg
1381 acgacgatgt cgtcggggcc acgccgcccg cgcgcctgct ggagccctgg gccggctgca
1441 acggcggggc ggcggcggac agcgactcga gcctggacga gccgtgccgc cccggcttcg
1501 cctcggaatc caaggtgtag ggcccggcgc ggggcgcgga ctccgggcac ggcttcccag
1561 gggaacgagg agatctgtgt ttacttaaga ccgatagcag gtgaactcga agcccacaat
1621 cctcgtctga atcatccgag gcaaagagaa aagccacgga ccgttgcaca aaaaggaaag
1681 tttgggaagg gatgggagag tggcttgctg atgttccttg ttg

DEFINITION	Homo sapiens adrenergic, beta-1-, receptor
ACCESSION	NM_000684
SOURCE ORGANISM	human
REFERENCE	1
AUTHORS	Frielle, Collins, Daniel, Caron, Lefkowitz, Kobilka
TITLE	Cloning of the cDNA for the human beta 1-adrenergic receptor
REFERENCE	2
AUTHORS	Frielle, Kobilka, Lefkowitz, Caron
TITLE	Human beta 1- and beta 2-adrenergic receptors: structurally and functionally related receptors derived from distinct genes

Entity-relationship

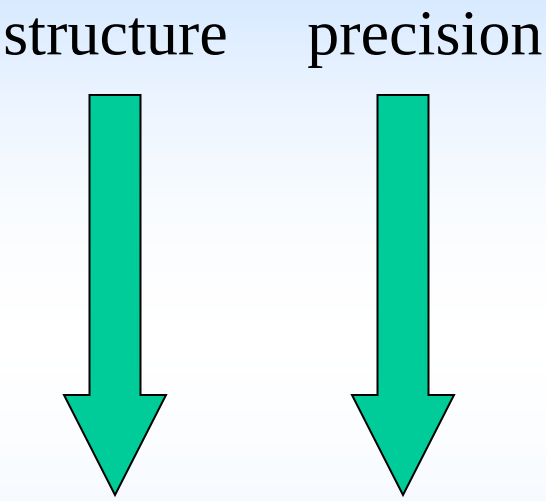


Storing and accessing textual information

- What information is stored?
- How is the information stored?
  - high level
- How is the information retrieved?

Storing textual information

- Text (IR)
- Semi-structured data
- Data models (DB)
- Rules + Facts (KB)



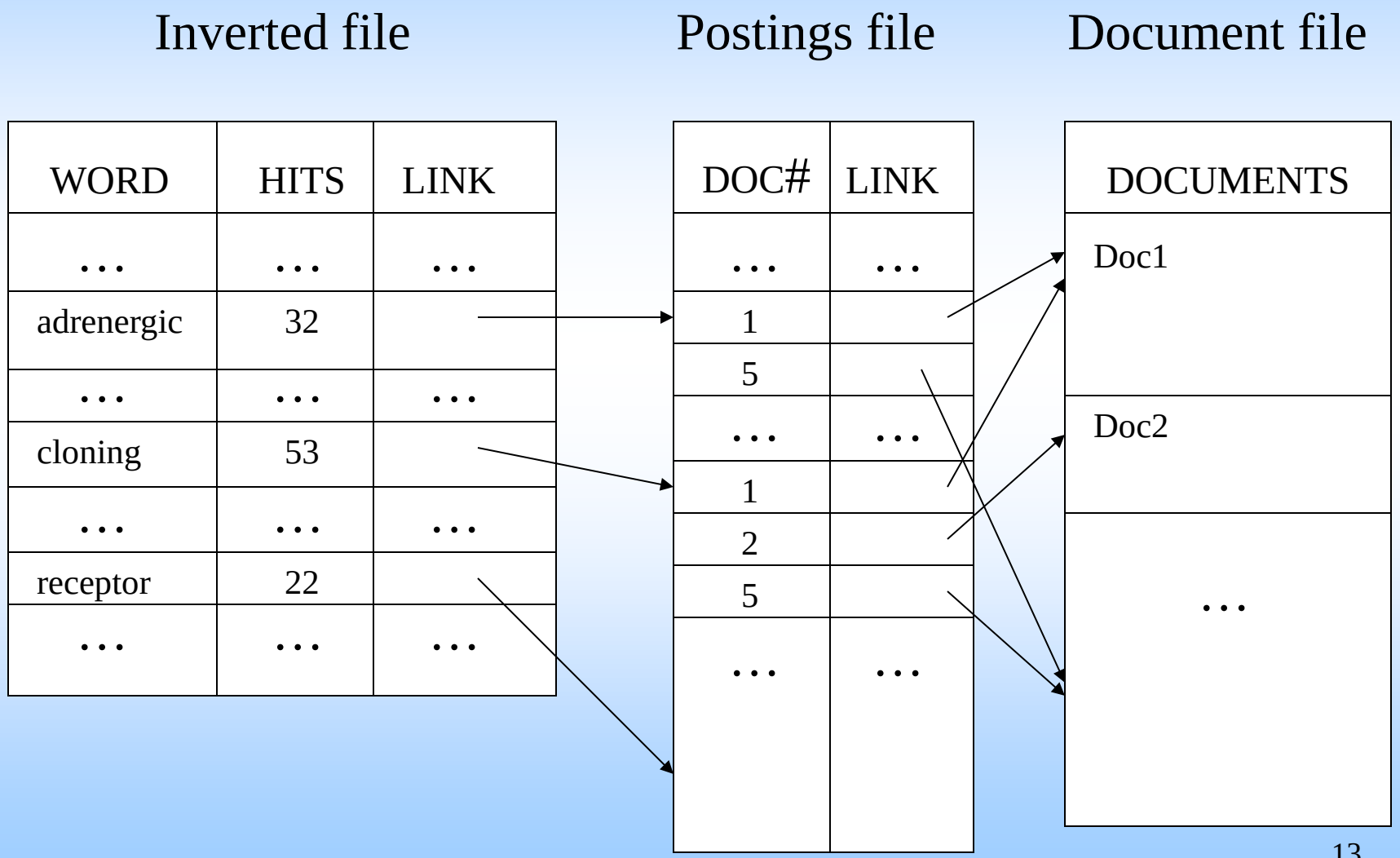
Storing textual information - Text - Information Retrieval

- search using words
- conceptual models:

  boolean, vector, probabilistic, …
- file model:

  flat file, inverted file, ...

IR - File model: inverted files



IR – File model: inverted files

- Controlled vocabulary
- Stop list
- Stemming

IR - formal characterization

Information retrieval model: (D,Q,F,R)

- D is a set of document representations
- Q is a set of queries
- F is a framework for modeling document representations, queries and their relationships
- R associates a real number to document-query-pairs (ranking)

IR - conceptual models

Classic information retrieval

- Boolean model
- Vector model
- Probabilistic model

Boolean model

Document representation

	adrenergic	cloning	receptor		
Doc1	yes	yes	no	-->	(1 1 0)
Doc2	no	yes	no	-->	(0 1 0)

Boolean model

queries : boolean (and, or, not)

Q1: cloning and (adrenergic or receptor)

Queries are translated to disjunctive normal form (DNF)

DNF: disjunction of conjunctions of terms with or without ‘not’

Rules: not not A --> A

not(A and B) --> not A or not B

not(A or B) --> not A and not B

(A or B) and C --> (A and C) or (B and C)

A and (B or C) --> (A and B) or (A and C)

(A and B) or C --> (A or C) and (B or C)

A or (B and C) --> (A or B) and (A or C)

Boolean model

Q1: cloning and (adrenergic or receptor)

--> (cloning and adrenergic) or (cloning and receptor)

DNF is completed

+ translated to same representation as documents

(cloning and adrenergic) or (cloning and receptor)

--> (cloning and adrenergic and receptor)

or (cloning and adrenergic and not receptor)

or (cloning and receptor and adrenergic)

or (cloning and receptor and not adrenergic)

--> (1 1 1) or (1 1 0) or (1 1 1) or (0 1 1)

--> (1 1 1) or (1 1 0) or (0 1 1)

Boolean model

	adrenergic	cloning	receptor		
Doc1	yes	yes	no	-->	(1 1 0)
Doc2	no	yes	no	-->	(0 1 0)

Q1: cloning and (adrenergic or receptor)

--> (1 1 0) or (1 1 1) or (0 1 1) Result: Doc1

Q2: cloning and not adrenergic

--> (0 1 0) or (0 1 1) Result: Doc2

Boolean model

Advantages

- based on intuitive and simple formal model (set theory and boolean algebra)

Disadvantages

- binary decisions

  - words are relevant or not

  - document is relevant or not, no notion of partial match

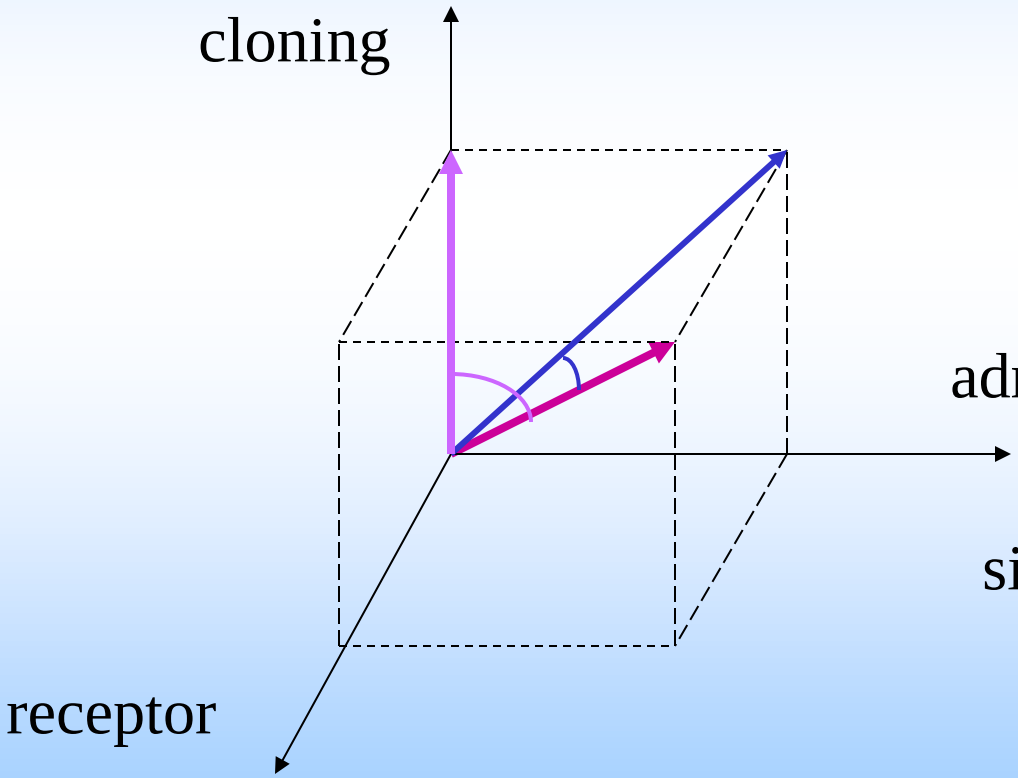
Boolean model

	adrenergic	cloning	receptor		
Doc1	yes	yes	no	-->	(1 1 0)
Doc2	no	yes	no	-->	(0 1 0)

Q3: adrenergic and receptor

--> (1 0 1) or (1 1 1) Result: empty

Vector model (simplified)



Doc1 (1,1,0)

Doc2 (0,1,0)

Q (1,1,1)

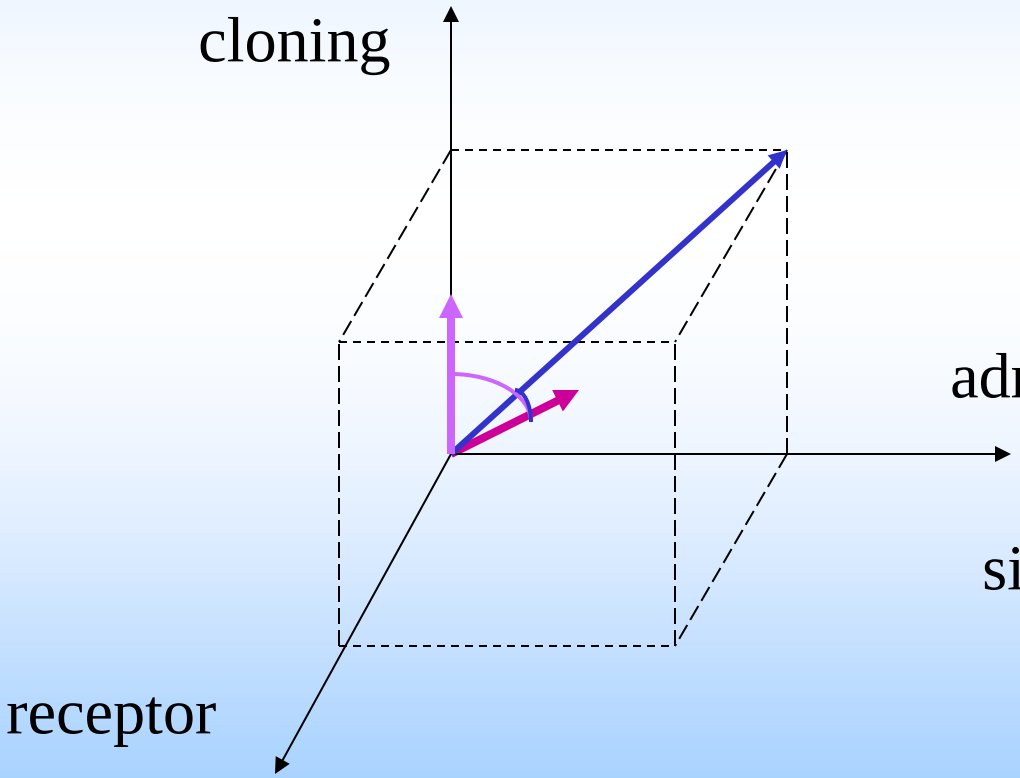
$$sim(d,q) = \frac{d \cdot q}{|d| \times |q|}$$

Vector model

- Introduce weights in document vectors

  (e.g. Doc3 (0, 0.5, 0))
- Weights represent importance of the term for describing the document contents
- Weights are positive real numbers
- Term does not occur -> weight = 0

Vector model



Doc1 (1,1,0)

Doc3 (0,0.5,0)

Q4 (0.5,0.5,0.5)

$$sim(d,q) = \frac{d \cdot q}{|d| \times |q|}$$

Vector model

- How to define weights? tf-idf

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

$w_{i,j}$ = weight for term k_i in document d_j

$= f_{i,j} \times idf_i$

Vector model

- How to define weights? tf-idf

term frequency $freq_{i,j}$: how often does term k_i occur in document d_j?

normalized term frequency:

$f_{i,j} = freq_{i,j} / max_l freq_{l,j}$

Vector model

- How to define weights? tf-idf

document frequency : in how many documents does term k_i occur?

N = total number of documents

n_i = number of documents in which k_i occurs

inverse document frequency idf_i: $\log (N / n_i)$

Vector model

- How to define weights for query?

recommendation:

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

$w_{i,q}$ = weight for term k_i in q

$= (0.5 + 0.5\, f_{i,q)} \times idf_i$

Vector model

- Advantages

- term weighting improves retrieval performance
- partial matching
- ranking according to similarity

Disadvantage

- assumption of mutually independent terms?

Probabilistic model

weights are binary ($w_{i,j} = 0$ or $w_{i,j} = 1$)

R: the set of relevant documents for query q

Rc: the set of non-relevant documents for q

$P(R|d_j)$: probability that dj is relevant to q

$P(Rc|d_j)$: probability that dj is not relevant to q

$sim(d_j,q) = P(R|d_j) / P(Rc|d_j)$

Probabilistic model

$sim(d_j,q) = P(R|d_j) / P(Rc|d_j)$

(Bayes' rule, independence of index terms,
take logarithms, $P(k_i|R) + P(\text{not } k_i|R) = 1$)

--> SIM(dj,q) ==

$$SUM_{i=1}^{t} \; w_{i,q} \times w_{i,j} \times$$

$$(\log(P(k_i|R) / (1- P(k_i|R))) +$$

$$\log((1- P(k_i|Rc) / P(k_i|Rc)))$$

Probabilistic model

- How to compute $P(k_i|R)$ and $P(k_i|Rc)$?

  - initially: $P(k_i|R) = 0.5$ and $P(k_i|Rc) = n_i/N$

  - Repeat: retrieve documents and rank them

  V: subset of documents (e.g. r best ranked)

  V_i: subset of V, elements contain k_i

  $P(k_i|R) = |V_i| / |V|$

  and $P(k_i|Rc) = (n_i - |V_i|) / (N - |V|)$

Probabilistic model

- Advantages:

- ranking of documents with respect to probability of being relevant

- Disadvantages:

- initial guess about relevance

- all weights are binary

- independence assumption?

IR - measures

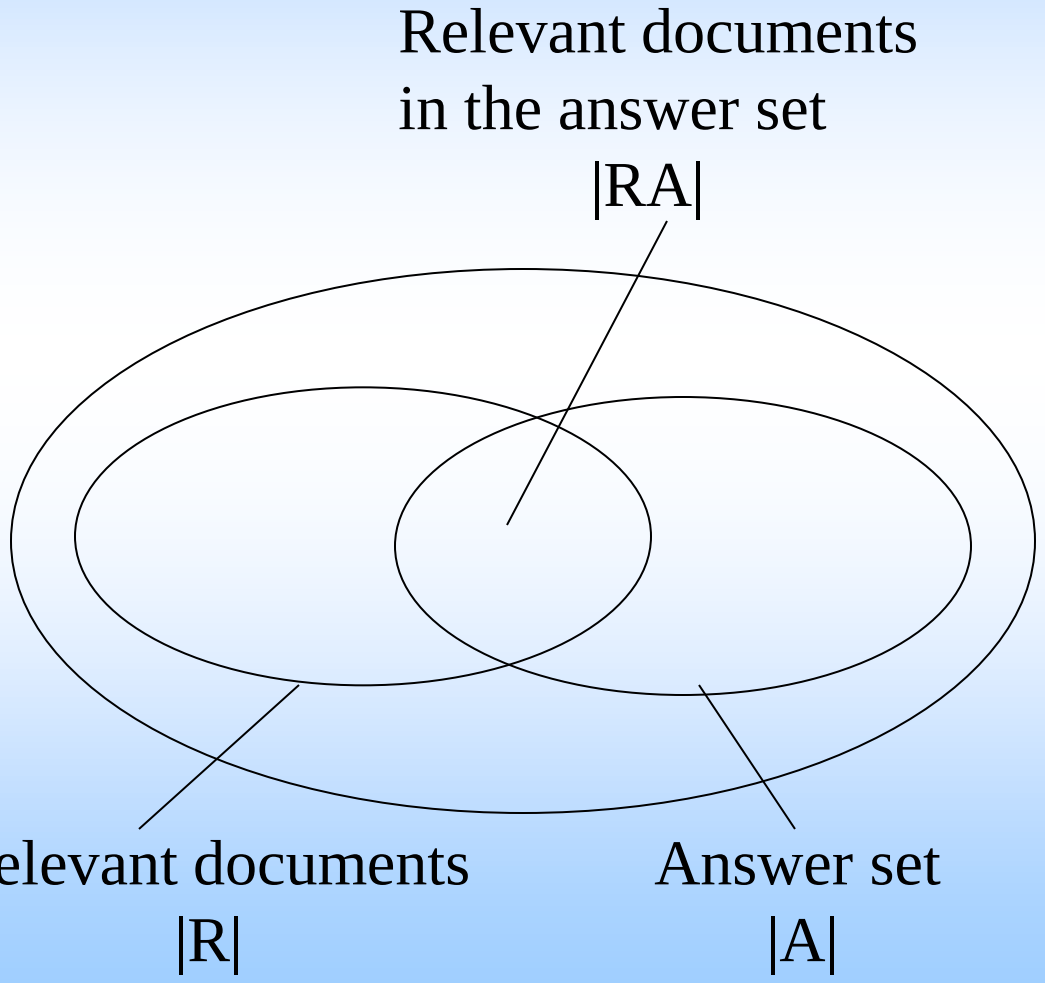
$$\text{Precision} = \frac{\text{number of found relevant documents}}{\text{total number of found documents}}$$

$$\text{Recall} = \frac{\text{number of found relevant documents}}{\text{total number of relevant documents}}$$

IR - measures

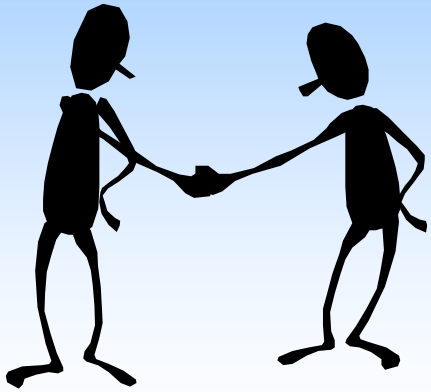
Precision =
|RA| / |A|

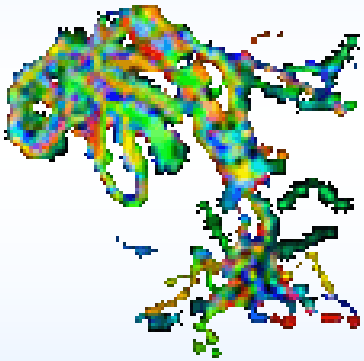
Recall =
|RA| / |R|



Related work at IDA/ADIT

- Use of IR/text mining in
  - Ontology engineering
    - Defining similarity between concepts (OA)
    - Defining relationships between concepts (OD)
- Semantic Web
- Databases





Literature

Baeza-Yates, R., Ribeiro-Neto, B., *Modern Information Retrieval*, Addison-Wesley, 1999.