Multimedia Retrieval

2 Fundamentals of Information Retrieval

Prof. Dr. Marcin Grzegorzek

Research Group for Pattern Recognition
www.pr.informatik.uni-siegen.de

Institute for Vision and Graphics
University of Siegen, Germany

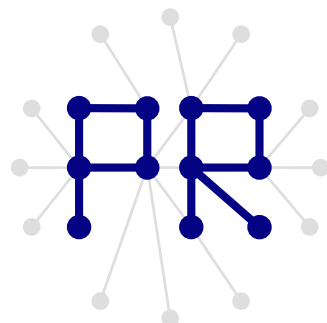


Table of Contents

Table of Contents

Table of Contents

Table of Contents

Overview

Overview

DB-, IR-, und MMDB-Systeme

- IR- und DB-Systeme verwalten Daten, unterscheiden sich jedoch erheblich im Zugriff auf die Daten.

- Datenbankanfrage ist scharf:

  select ISBN
  from Buch
  where Titel = "Multimedia-Datenbanken"

- IR-Anfrage ist in der Regel unscharf formuliert:
  Finde alle Text-Dokumente, die sich mit dem Thema "Multimedia-Datenbanken" beschäftigen.

- MMDB-Systeme kombinieren Konzepte von DB- und IR-Systemen.

Informationsbedarf in einem IR-System

Der Informationsbedarf in einem IR-System kann unterschiedlich verstanden werden:

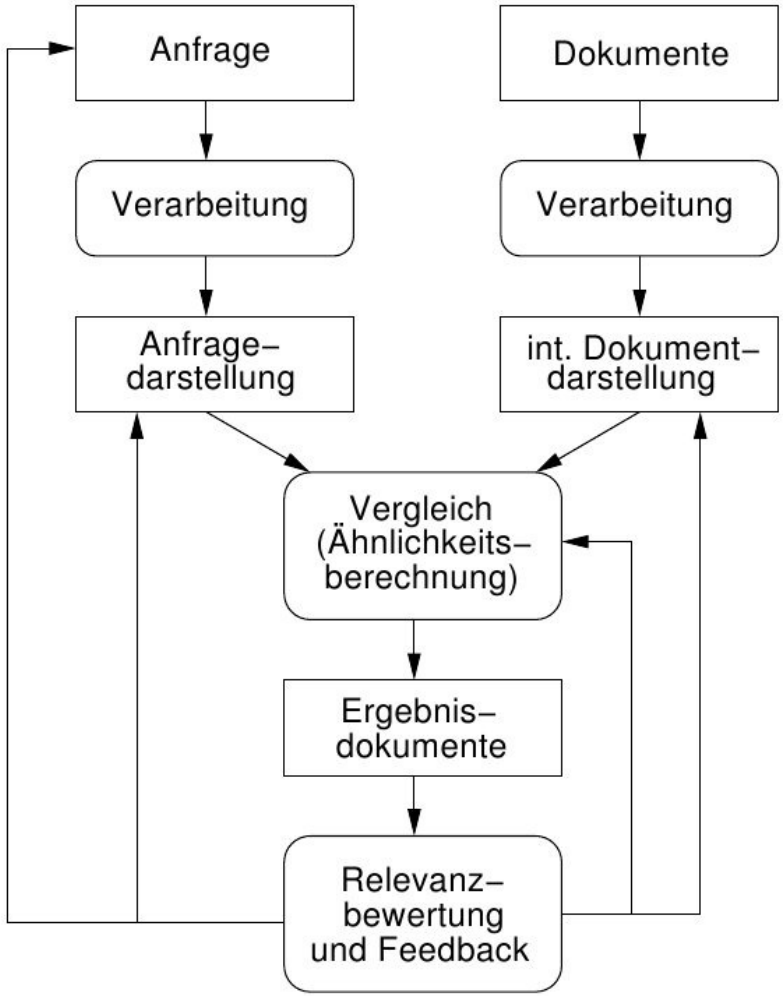
- als Dokument:
  Liefere alle Text-Dokumente, die ähnlich zum Text-Dokument #0821 sind

- als Anfrage:
  Datenbank `and` (Bild `or` Video)

Daten Retrieval versus Information Retrieval

Merkmal	**Daten Retrieval**	**Inf. Retrieval**
Information	explizit	implizit
Ergebnisse	exakt	unscharf
Anfrage	einmalig	iterativ verfeinernd
Fehlertoleranz	keine	vorhanden
Ergebniskollektion	Menge	Liste

Schritte des IR-Prozesses



Overview

Klassifikation der IR-Modelle

Boolesches Modell

- Dokumente werden als Indexterme repräsentiert.
- Die Suche erfolgt über einfache Mengenoperationen.
- Die Anfragen lassen sich über boolesche Junktoren verknüpfen.

Fuzzy-Modell

- Eine Erweiterung des booleschen Modells auf unscharfe Mengen.

Vektorraummodell

- Jedes Dokument wird als ein Vektor aufgefasst.
- Eine Anfrage wird auch als Vektor in einem Vektorraum behandelt.
- Die Suche basiert auf Bestimmung von Vektorähnlichkeiten.

Boolesches Modell - Allgemeines

Binäres Termgewicht

- Das Gewicht eines Terms bezogen auf ein Text-Dokument ist binär ("1" - das Dokument beinhaltet den Term, "0" - das Dokument beinhaltet den Term nicht).

Boolesche Junktoren

- In der Anfrage werden Terme durch boolesche Junktoren (and, or, not) kombiniert.

Vergleichsfunktion

- Innerhalb der Vergleichsfunktion werden die durch die Anfrage spezifizierten Anfrageterme in den jeweiligen Dokumenten auf Enthaltensein getestet.

Boolesches Modell - Beispiel

Terme:

Indexvokabular = {Korsika, Sardinien, Strand, Ferienwohnung, Gebirge}

Dokumente:

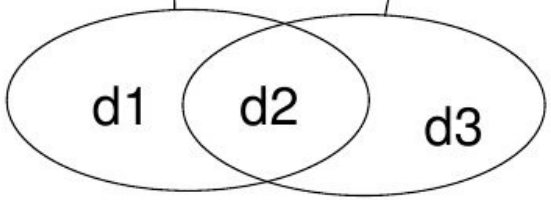
Dokument 1 :	{Sardinien, Strand, Ferienwohnung}
Dokument 2 :	{Korsika, Strand, Ferienwohnung}
Dokument 3 :	{Korsika, Gebirge}

Ergebnisse:

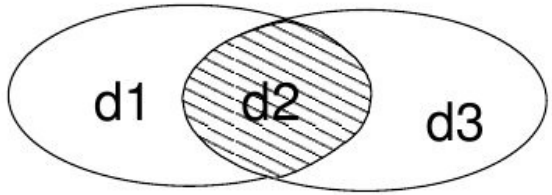
Korsika	liefert {d2,d3}
Ferienwohnung	liefert {d1,d2}
Ferienwohnung and Korsika	liefert {d2}
Ferienwohnung or Korsika	liefert {d1,d2,d3}
Ferienwohnung and not Korsika	liefert {d1}

Boolesches Modell - Beispiel

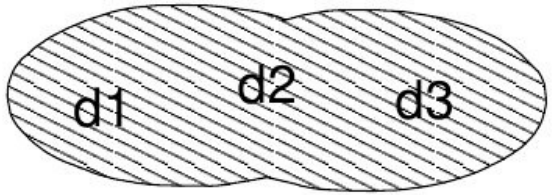
Ferienwohnung *Korsika*



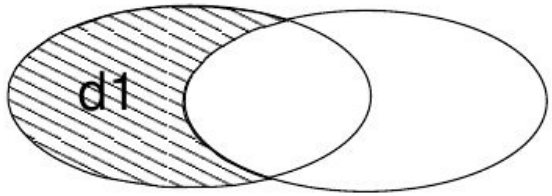
Ferienwohnung **and** *Korsika*



Ferienwohnung **or** *Korsika*



Ferienwohnung **and not** *Korsika*



Boolesches Modell - Weiteres

“but”-Junktor

- Die Anfrage *Ferienwohnung but Korsika* liefert alle die Dokumente, die den Term “Ferienwohnung” aber nicht den Term “Korsika” enthalten.

“of”-Konstrukt

- Die Anfrage *2 of (Korsika, Strand, Ferienwohnung)* sucht nach allen Dokumenten, die mindestens zwei der drei vorgegebenen Terme enthalten.

Fuzzy-Modell, Allgemeines

- Das Fuzzy-Modell ist eine Erweiterung des booleschen Modells.
- Das Problem der zu scharfen Enthaltenseinsbedingung von Termen in Dokumenten wird behoben.
- Die Grundidee liegt in der Verwendung einer graduellen Zugehörigkeit von Dokumenten zu Termen.
- Es wird auf das Konzept einer Fuzzy-Menge zurückgegriffen.

Fuzzy-Modell, Definitionen

Fuzzy-Menge
Eine Fuzzy-Menge $A = \{\langle u, \mu_A(u)\rangle\}$ über ein Universum U ist durch eine Zugehörigkeitsfunktion $\mu_A : U \rightarrow [0, 1]$ charakterisiert, welche jedem Element u des Universums U einen Wert $\mu_A(u)$ aus dem Intervall $[0, 1]$ zuordnet.

Term als Fuzzy-Menge
In unserem Retrieval-Szenario entspricht die Menge aller gespeicherten Dokumente dem Universum und ein Term einer Fuzzy-Menge.

Fuzzy-Wert
Fuzzy-Wert $\mu_t(d_1)$ des Dokuments d_1 bezüglich des Terms t drückt aus, wie stark der Term das Dokument charakterisiert.

Fuzzy-Modell, Definitionen

Mengendurchschnitt
Der Mengendurchschnitt $A \cap B$ (Konjunktion) wird durch die Min-Funktion realisiert $\mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u))$

Mengenvereinigung
Die Mengenvereinigung $A \cup B$ (Disjunktion) wird durch die Max-Funktion realisiert $\mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u))$

Komplementbildung
Die Komplementbildung $\overline{A}$ (Negation) bezüglich des Universums entspricht einer Subtraktion von 1
$\mu_{\overline{A}}(u) = 1 - \mu_A(u)$

Fuzzy-Modell, Beispiel

Anfrage	μ	d_1	d_2	d_3
	μ_{Korsika}	0,1	0,6	1
	μ_{Strand}	0,3	0,2	0,8
1	$\mu_{\text{Korsika} \cap \text{Strand}}$			
2	$\mu_{\text{Korsika} \cup \text{Strand}}$			
3	$\mu_{\overline{\text{Korsika}}}$			

Fuzzy-Modell, Beispiel

Anfrage	μ	d_1	d_2	d_3
	μ_{Korsika}	0,1	0,6	1
	μ_{Strand}	0,3	0,2	0,8
1	$\mu_{\text{Korsika} \cap \text{Strand}}$	0,1	0,2	0,8
2	$\mu_{\text{Korsika} \cup \text{Strand}}$	0,3	0,6	1
3	$\mu_{\overline{\text{Korsika}}}$	0,9	0,4	0

Fuzzy-Modell, Term-zu-Term-Korrelationsmatrix

- $c_{i,j}$ in Zeile t_i und Spalte t_j drückt aus, wie stark die Terme t_i und t_j in den Dokumenten der Datenbank korrelieren, also in den Dokumenten gemeinsam auftreten

$$c_{i,j} = \frac{n_{i,j}}{n_i + n_j - n_{i,j}}$$

- Zugehörigkeitswert eines Dokuments d_j zu einem Term t_j

$$\mu_{t_i}(d_j) = 1 - \prod_{t_k \in d_j} (1 - c_{i,k})$$

Vektorraummodell - Allgemeines

- Die Dokumente werden wie Vektoren eines Vektorraums aufgefasst.
- Vektoren aus Termgewichten oder Merkmalswerten
- Anfrage als Vektor
- Ähnlichkeits- und Distanzmaße

Vektorraummodell - Beispiel

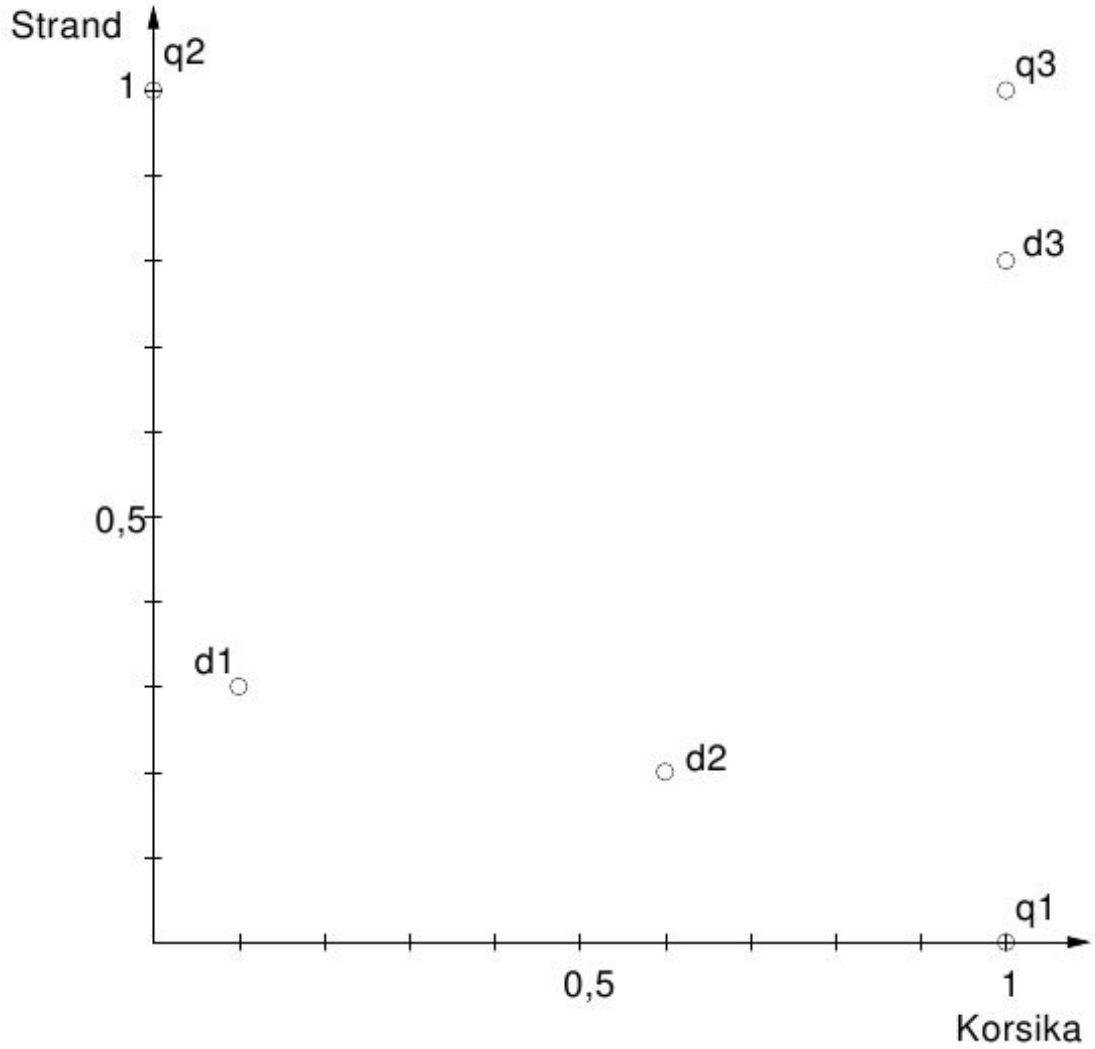
Dokumente:

Dimension	d_1	d_2	d_3
Korsika	0,1	0,6	1
Strand	0,3	0,2	0.8

Anfragen:

Dimension	q_1	q_2	q_3
Korsika	1	0	1
Strand	0	1	1

Vektorraummodell - Beispiel



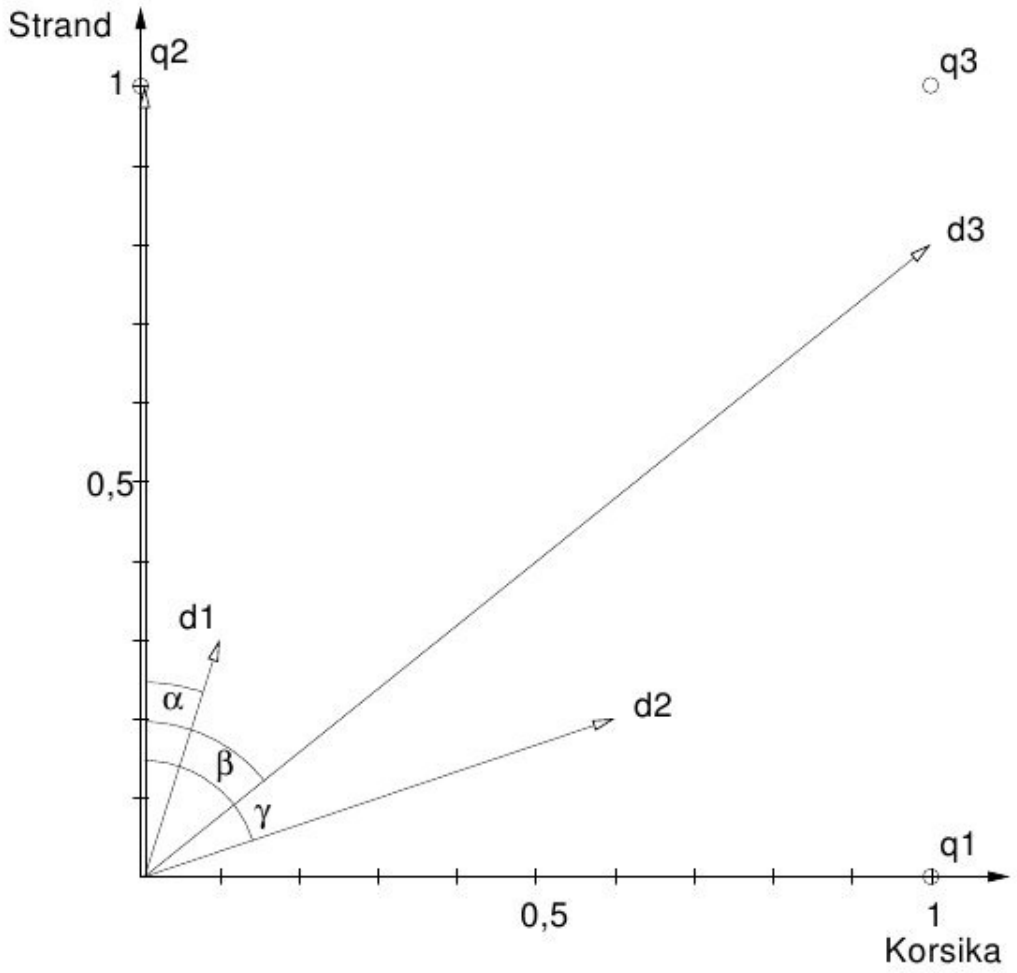
Vektorraummodell - Beispiel

Ähnlichkeitswerte nach Kosinusmaß

$\mathrm{sim}_{\cos}$	d_1	d_2	d_3
q_1	0,3162	0,9487	0,7809
q_2	0,9487	0,3162	0,6247
q_3	0,8944	0,8944	0,9939

Vektorraummodell - Beispiel

$\cos \alpha = 0,9487$
$\cos \beta = 0,6247$
$\cos \gamma = 0,3162$



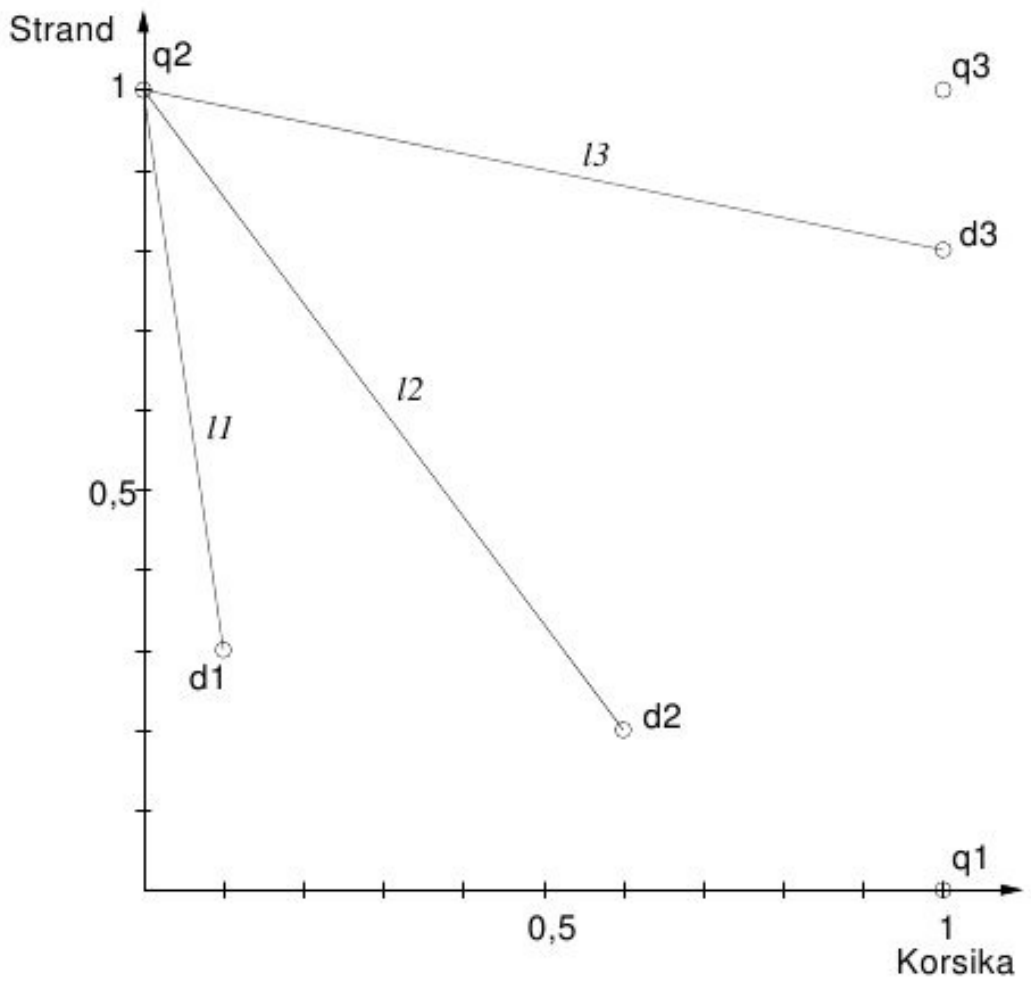
Vektorraummodell - Beispiel

Unähnlichkeitswerte anhand Euklidischer Distanz

dissim_{L_2}	d_1	d_2	d_3
q_1	0,9487	0,4472	0,8
q_2	0,7071	1	1,0198
q_3	1,1402	0,8944	0,2

Vektorraummodell - Beispiel

$l_1 = 0,7071$
$l_2 = 1$
$l_3 = 1,0198$



Vektorraummodell - Beispiel

- Kosinusmaß und Euklidische Distanz erzeugen unterschiedliche Ergebnisse
  - Kosinusmaß erzeugt $\langle d_1, d_3, d_2 \rangle$
  - Euklidische Distanz erzeugt $\langle d_1, d_2, d_3 \rangle$

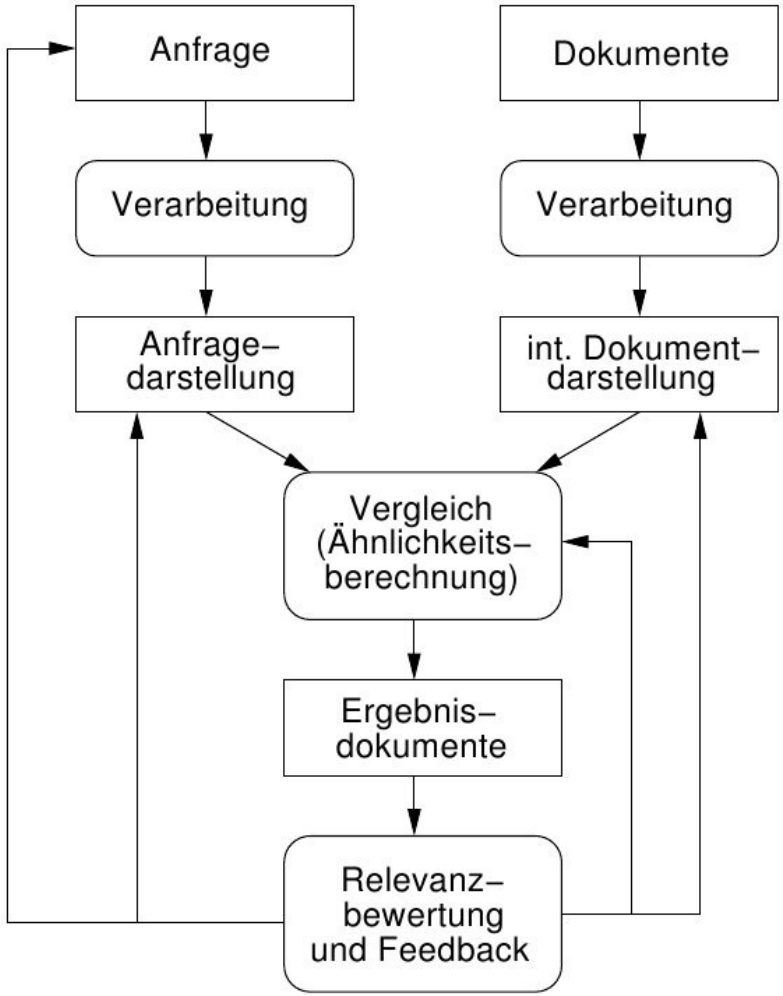
- Wahl der geeigneten Ähnlichkeitsfunktion abhängig von
  - subjektivem Ähnlichkeitsempfinden
  - Anwendungsszenario

Vektorraummodell - Zusammenfassung

- Vektorraummodell ist sehr verbreitet.
- VR-Modell setzt feste Anzahl von nummerischen Merkmalswerten pro Dokument voraus.
- Probleme:
  - Merkmale als orthogonale Dimensionen aufgefasst (unrealistisch)
  - Problem bei hoher Anzahl von Merkmalswerten bzgl. Effektivität und Effizienz
  - Anfrage ist Vektor, also keine Junktoren

Overview

Vereinfachter IR-Prozess



RF - Beispiel

Anfrage	Ergebnisdokumente			
	1	2	3	...
q	d_0	d_1	d_2	...
q_0	d_4	d_1 (+)	d_5 (−)	...
q_1	d_1 (+)	d_3 (+)	d_4 (−)	...
q_2	d_3	d_1	d_0	...

q korrekte aber unbekannte Anfrage

q_0 initiale Anfrage

q_1 Anfrage nach 1. Iteration

q_2 Anfrage nach 2. Iteration

RF - Anfragemodifikation im Vektorraummodell

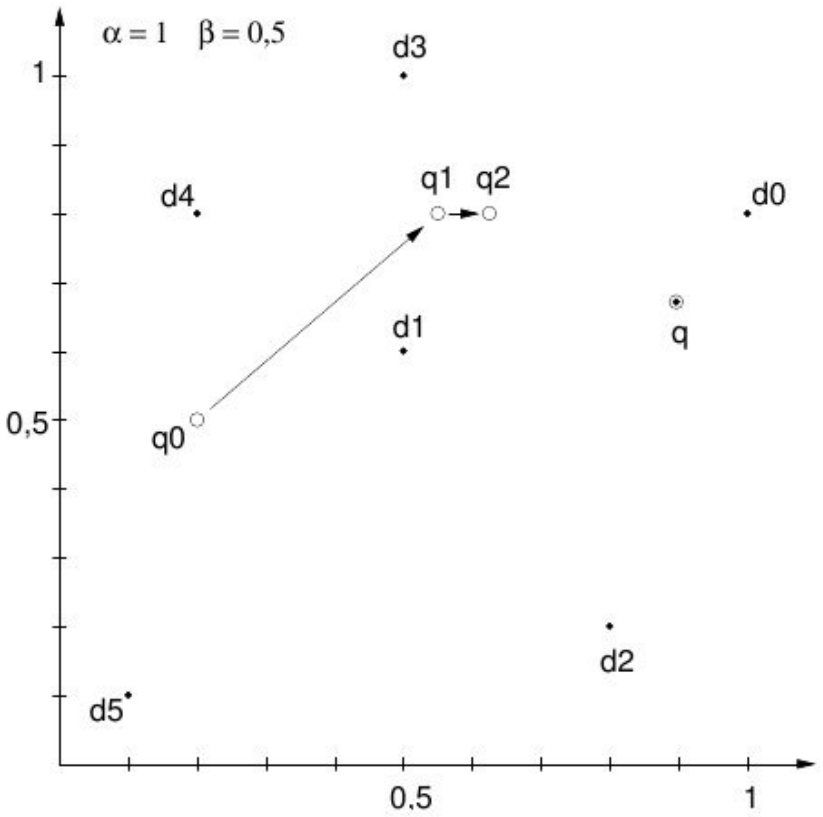
Verschiebung des Anfragevektors

- in Richtung der als relevant bewerteten Dokumente
- weg von als irrelevant bewerteten Dokumenten
- Anfrage: q_{alt}, relevant (irrelevant) bewertete Dokumente: D_r (D_i)
- α und β gewichten Einfluss relevanter und irrelevanter Dokumente

$$q_{\text{neu}} = q_{\text{alt}} + \frac{\alpha}{|D_r|} \sum_{d_r \in D_r} d_r - \frac{\beta}{|D_i|} \sum_{d_i \in D_i} d_i$$

RF - Beispiel zur Anfragemodifikation

An-frage	Ergebnisdokumente			
	1	2	3	...
q	d_0	d_1	d_2	...
q_0	d_4	d_1 (+)	d_5 (−)	...
q_1	d_1 (+)	d_3 (+)	d_4 (−)	...
q_2	d_3	d_1	d_0	...



Overview

Bewertung - Allgemeines

- Bewertung (Qualitätsvergleich) verschiedener Retrieval-Systeme
- Quantitative Maße vonnöten

Bewertung - Precision, Recall und Fallout

- Zwei verschiedene Fehlentscheidungen
  - *false alarms* (f_a) bezeichnet diejenigen Dokumente, die vom Retrieval-System irrtümlicherweise als relevant zurückgeliefert wurden (auch: *false positives*)
  - *false dismissals* (f_d) sind Dokumente, die fälschlicherweise vom Retrieval-System als irrelevant eingestuft wurden (auch: *false negatives*)

- Zwei korrekte Entscheidungen
  - *correct alarms* (c_a)
  - *correct dismissals* (c_d)

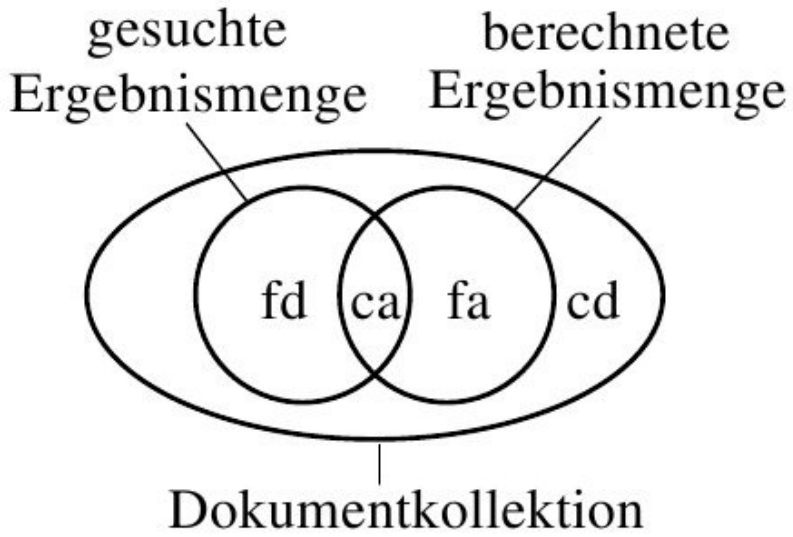
- f_a, f_d, c_a, c_d stehen für entsprechende Dokumentanzahlen bzgl. einer Anfrage.

Bewertung - Precision, Recall and Fallout

Nutzer-bewertung	Systembewertung	
	relevant	irrelevant
relevant	ca	fd
irrelevant	fa	cd

Bewertung - Precision, Recall and Fallout

$$
\begin{aligned}
|\text{gesuchte Ergebnismenge}| &= \text{fd} + \text{ca} \\
|\text{berechnete Ergebnismenge}| &= \text{ca} + \text{fa} \\
|\text{Dokumentkollektion}| &= \text{fd} + \text{ca} + \text{fa} + \text{cd}
\end{aligned}
$$



Bewertung - Precision

Precision P

Wie viele (als Verhältnis) Ergebnisdokumente sind tatsächlich relevant?

$$P = \frac{c_a}{c_a + f_a} \qquad P \in [0, 1]$$

Bewertung - Recall

Recall R

Wie viele (als Verhältnis) tatsächlich relevante Dokumente erscheinen im Ergebnis?

$$R = \frac{c_a}{c_a + f_d} \qquad R \in [0, 1]$$

Bewertung - Fallout

Fallout F

Verhältnis falsch gefundener zur Gesamtzahl irrelevanter Dokumente

$$F = \frac{f_a}{f_a + c_d} \qquad F \in [0, 1]$$

Bewertung - Precision, Recall und Fallout

- Precision, Recall und Fallout sind definiert bzgl. einer Anfrage
- Es ist besser, mehrere Anfragen zu betrachten und entsprechende Durchschnittswerte zu berechnen.

Bewertung - Precision, Recall and Fallout

20 Dokumente, 2 Anfragen, jeweils 10 Ergebnisdokumente

Anfrage	fa	ca	fd	cd	Precision	Recall	Fallout
q_1	8	2	6	4	20%	25%	66%
q_2	2	8	2	8	80%	80%	20%
Durchschnitt	–	–	–	–	50%	52,5%	43%

Overview

Nutzerprofile - Allgemeines

- Bis jetzt keine Unterscheidung von Anwender und Anwendergruppen
- Verhalten bzw. Suchbedarf verschiedener Nutzer differiert oft
- Idee: Subjektivität wird als Nutzerprofil modelliert und bei Suche berücksichtigt

Nutzerprofile - Retrieval mit Profilen

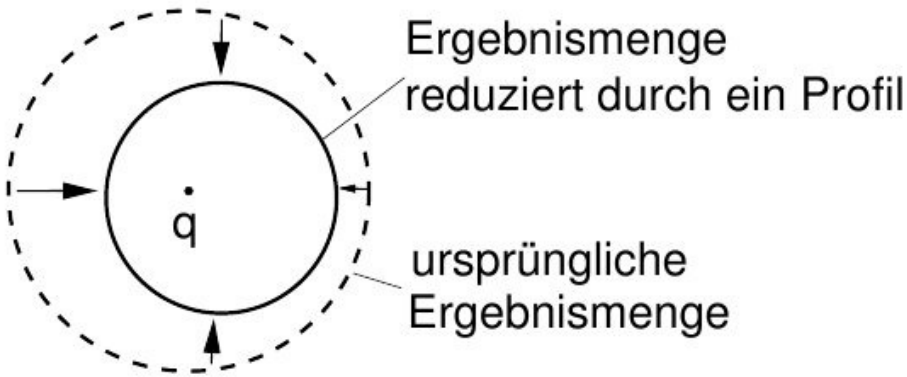
Nachfiltern

- Filterung auf Anfrageergebnis
- hoher Berechnungsaufwand durch u. U. großem Zwischenergebnis
- reduzieren von nur false alarms

Vorfiltern

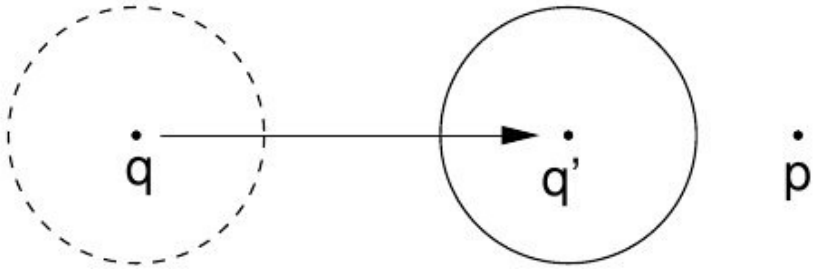
- Nutzerprofil beeinflusst Retrieval-Prozess direkt
- Reduzierung von false alarms und false dismissals

Nutzerprofile - Nachfiltern



Nutzerprofile - Vorfiltern

- Annahme: Anfrage als Profil
- einfache Realisierung: Verschiebung Anfragepunkt q in Richtung Profilanfragepunkt p



- Problem: relevante Dokumente bzgl. q können irrelevant bzgl. q' werden
- gewünscht: Reduzierung false dismissals statt Reduzierung false alarms