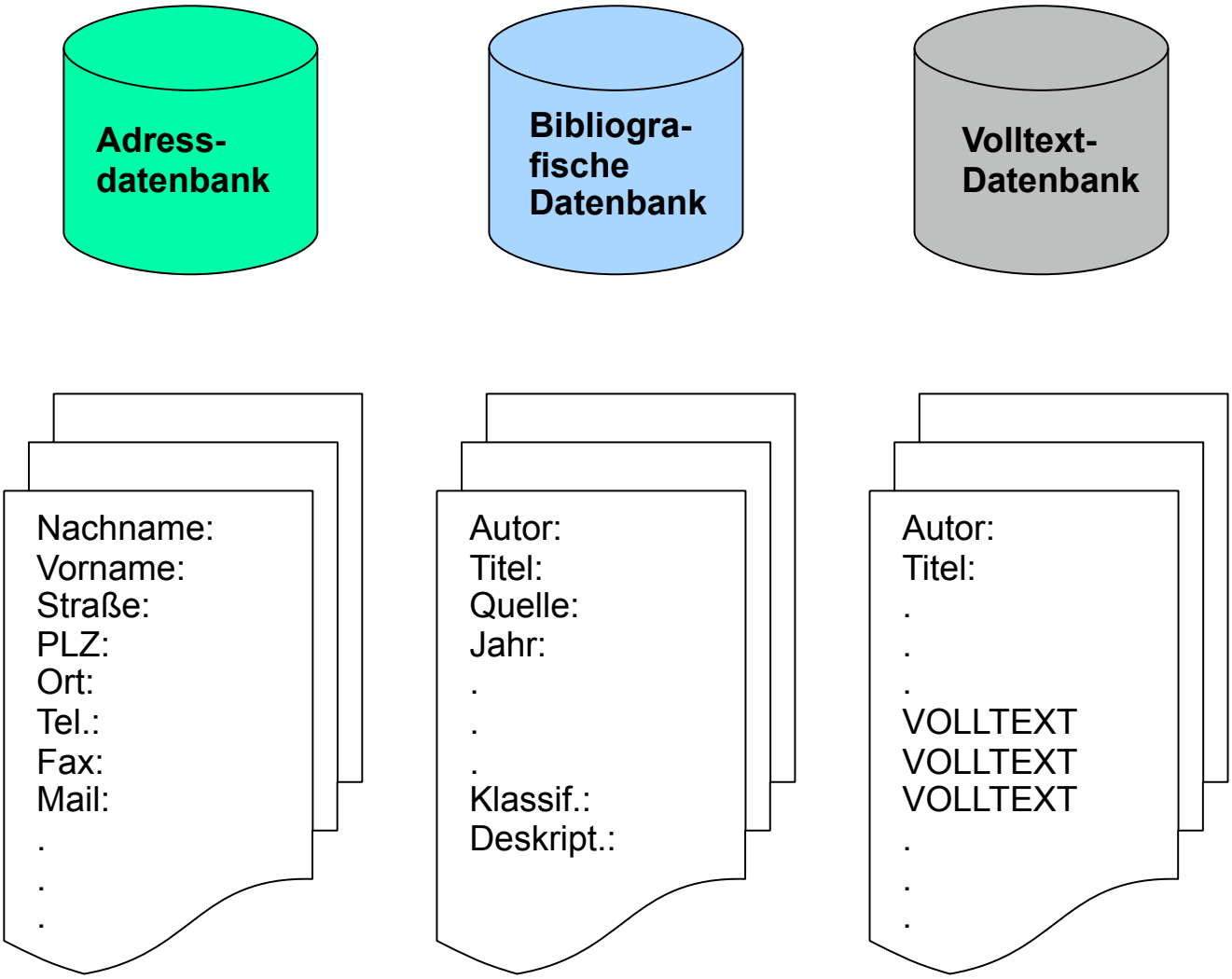
Methoden und Verfahren des Information Retrieval

1. Einführung in die Thematik

Informationssuche, Informationssysteme, Datenbanken

Drei Datenbanktypen



☞ Welcher *Datentypus* ist für den jeweiligen *Datenbanktypus* charakteristisch?

☞ Wie lassen sich jeweils die *Datenbankinhalte* sortieren/strukturiert ausgeben?

☞ Welcher *Suchtypus* ist für den jeweiligen *Datenbanktypus* charakteristisch?

Eigenschaften von Datenbanktypen

Adressdatenbank Information: Telefonnummer von Kurt Mayer

Nachname:
Vorname:
Straße:
PLZ:
Ort:
Tel.:
Fax:
Mail:
etc.

- feststehende Feldinhalte
- begrenzbare Feldlänge
- eindeutige Strukturierung der Feldinhalte
- kategorisierte Sortier- bzw. Ausgabemöglichkeit
- Suche ist zielgerichtete *Daten*suche

Bibl. Datenbank Information: Aufsätze von Kurt Mayer in 1996

Autor:
Titel:
Quelle:
Jahr:
.
Klassif.:
Deskript.:

- feststehender Typus von Feldinhalt
- Feldlänge (im Prinzip) variabel
- Feldinhalte teilweise strukturiert
- kategorisierte Sortier- bzw. Ausgabemöglichkeit
- Suche ist teilweise zielgerichtete Datensuche, teilweise unspezifische Textsuche

Volltext-Datenbank Information: In welchen Aufsichtsräten sitzt Kurt Mayer

Autor:
Titel:
.
VOLLTEXT
VOLLTEXT
VOLLTEXT

- Feldinhalte größtenteils vage
- Feldlänge variabel
- Feldinhalte nicht strukturiert
- keine kategorisierte Sortier- bzw. Ausgabemöglichkeit
- Suche ist **Information Retrieval**

Definitionen für Information Retrieval

An Information Retrieval System is a system that is capable of storage, retrieval and maintenance of information.
Kowalski 1997, 2

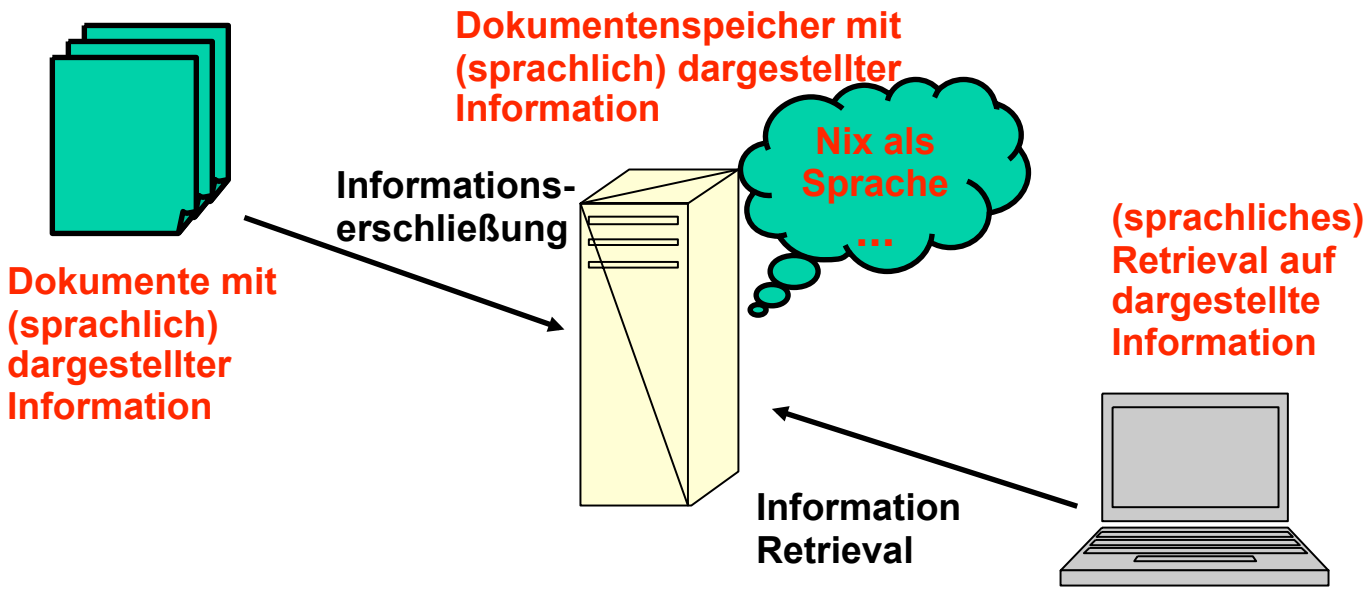
Informationsspeicherung | **Informationssuche / Informationswieder-gewinnung** | **Informationsverwaltung**

☞ Information ist gebunden an einen **Informationsträger**, ein Medium – Text, Bild, Film, Musik, Tabellen, Formeln etc. –, das die Basis für ein Information Retrieval ist.

☞ **80%** (geschätzt) der weltweit verfügbaren Information liegt in Textform vor, d.h. Information Retrieval ist fast immer **Text Retrieval**.

☞ Dabei steigt die **Bedeutung des Information Retrieval** mit der enorm wachsenden Verfügbarkeit von Information kontinuierlich.

Information-Retrieval-Systeme (IRS) sind interaktive Informations-systeme für vage Anfragen und unsicheres Wissen.
Norbert Fuhr



Themenübersicht

2. Funktionen von IR-Systemen I
2.1 Elementare Suchfunktionen

Autor(en) (5):	Salton, G.
Titel (10):	¬The state of retrieval system evaluation
Quelle (25):	Information processing and management. 28(1992) no.4, S.441-449.
Jahr (30):	1992
Abstract (40):	Substatioal misgivings have been voiced over the years about the methodologies used to evaluate IR procedures and about the credibility of many of the available test results. In this note, an attempt is made to review the state of retrieval evaluation and to separate certain misgivings about the design of retrieval tests from conclusions that can legitimately be drawn from the evaluation results
Systematik (50):	Retrievalstudien

☞ Welche Suche(n) führte(n) zu diesem Dokument?

☞ Für welche Suchthemen sollte dieses Dokument ein Treffer sein?

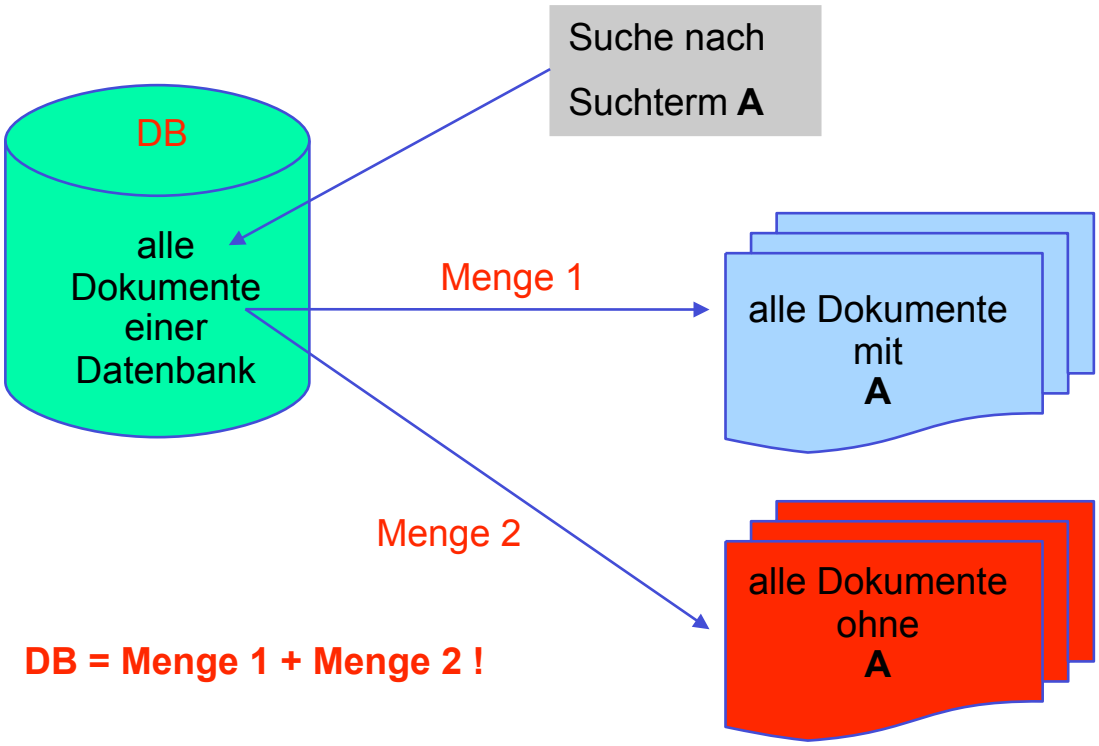
☞ Wie muss man dann suchen (können)?

A. Boolesches Retrieval

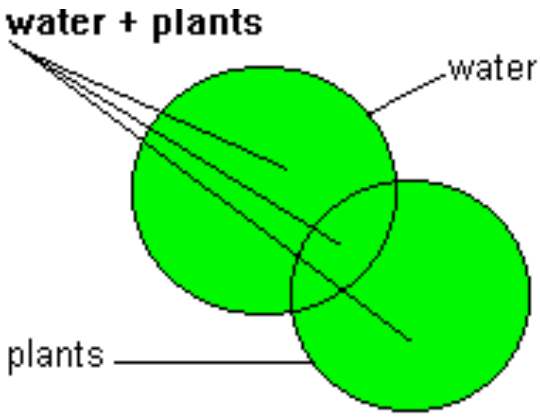
Übersicht

- erlaubt die Abfrage **komplexer Suchthemen**, d.h. Themen, die aus der Verbindung von mehreren Sachverhalten bestehen
- stellt für die Abwicklung komplexer Suchen **logische Operatoren** zur Verfügung:
    - **AND** – logisches UND
    - **OR** – logisches ODER
    - **(AND) NOT** – logisches NICHT
- ist ein **matching-orientiertes Suchverfahren** auf der Basis von Zeichenketten
- verknüpft mehrere Suchbegriffe durch **Mengenoperationen**
- ist als Standardsuche Bestandteil aller IR-Systeme

Einfaches Matching (mit einem Suchbegriff)



Logisches ODER



Findet alle Dokumente, die

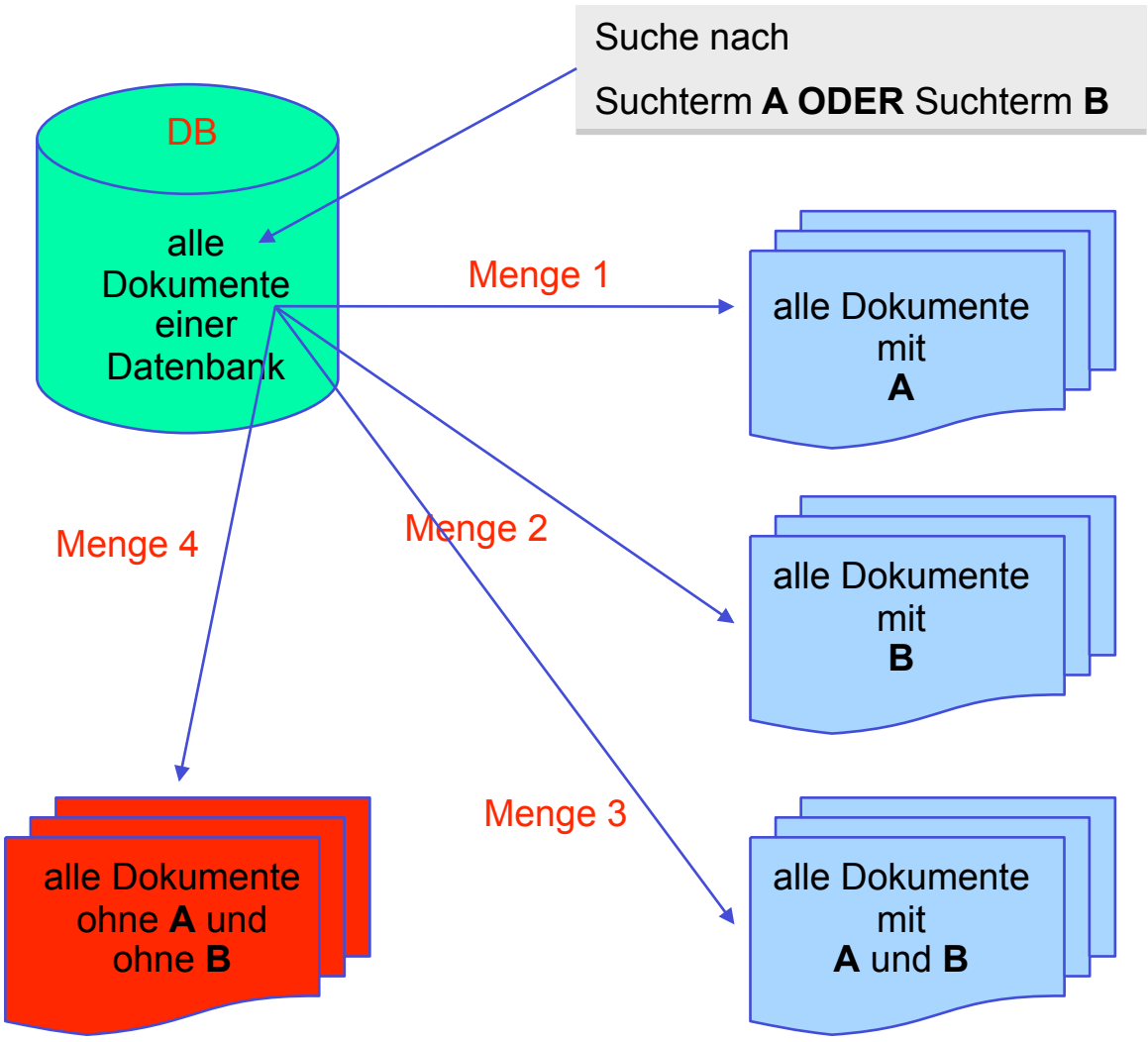
- **entweder** Suchterm A (water)
- **oder** Suchterm B (plants)
- **oder** Suchterm A und Suchterm B

  enthalten, d.h. nicht gefunden werden alle Dokumente, die

- **weder** Suchterm A **noch** Suchterm B enthalten

Achtung: Das logische **ODER** entspricht dem sprachlichen "und"!
"Halteverbot an Sonn- und Feiertagen" gilt an Sonn- **ODER** Feiertagen

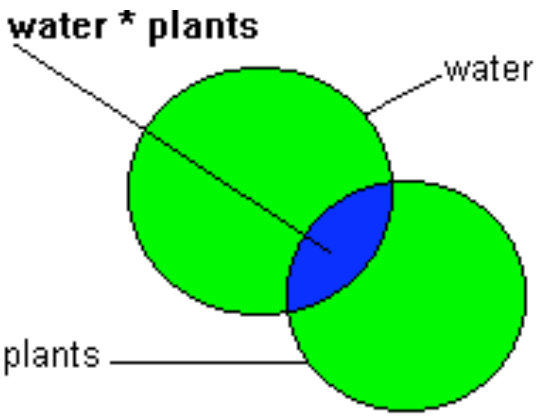
Grundsätzlich vergrößert ein logisches **ODER** die Menge der gefundenen Dokumente, weil es zusätzliche Matching-Möglichkeiten liefert.



Verwendung des booleschen **ODER**

- **breiter bzw. vager Einstieg** in eine Suche, z.B.
  - weil die in den Dokumenten verwendete Terminologie nicht bekannt ist,
  - weil sich das Thema nur schwer auf *einen Begriff* bringen lässt,
  - weil nur sehr wenige Treffer vermutet werden
- **Ausweitung einer vorhandenen Treffermenge**, die zu klein ist
- **Berücksichtigung sprachlicher Varianten** für die Suche, z.B.
  - Tür **ODER** Tor **ODER** Portal

Logisches UND

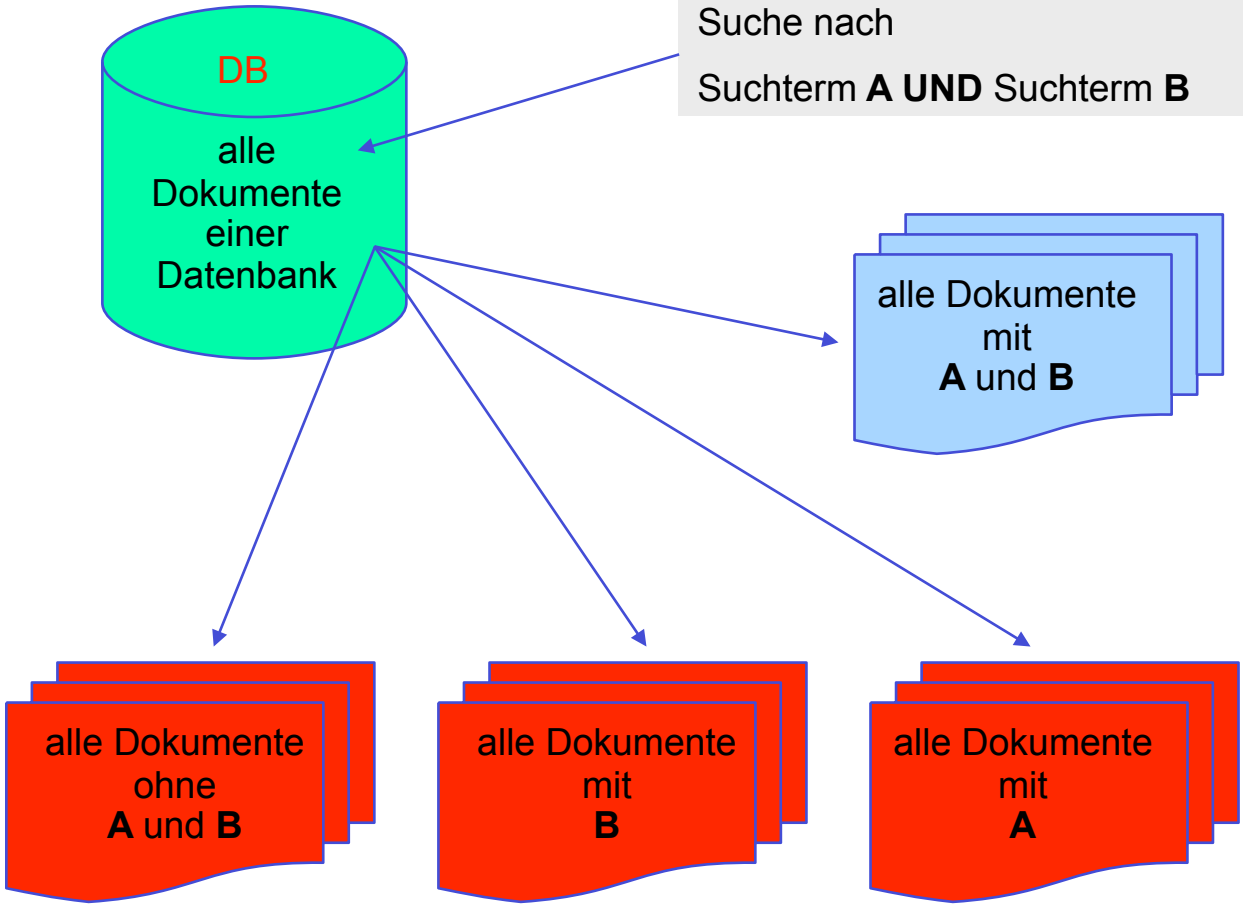


Findet alle Dokumente, die

- Suchterm A (water) **und** Suchterm B (plants)

  enthalten, d.h. nicht gefunden werden alle Dokumente, die

- Suchterm A **und nicht** Suchterm B enthalten
- Suchterm B **und nicht** Suchterm A enthalten
- **weder** Suchterm A **noch** Suchterm B enthalten



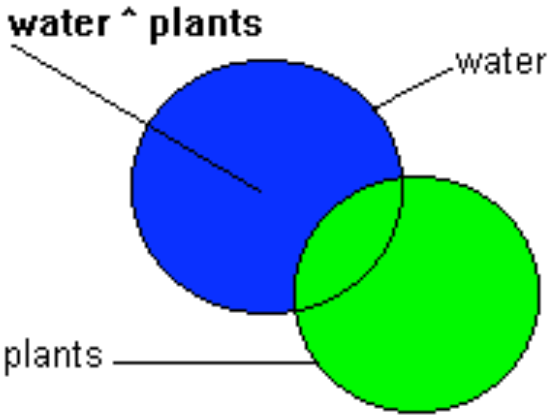
Verwendung des booleschen **UND**

- **Suche eines komplexen Themas**, das sich durch eine Kombination von Begriffen gut beschreiben lässt
- **enger bzw. genauer Einstieg** in eine Suche, z.B.
    - weil die in den Dokumenten verwendete Terminologie bekannt ist,
    - weil sich das Thema leicht auf *Begriffe* bringen lässt,
    - weil viele Treffer vermutet werden
- **Einschränkung einer vorhandenen Treffermenge**, die zu groß ist

Das boolesche **UND** entpsricht dem sprachlichen "sowohl als auch"

Das logische **UND** verkleinert Treffermengen (rapide)

Logisches (AND) NOT



Findet alle Dokumente, die

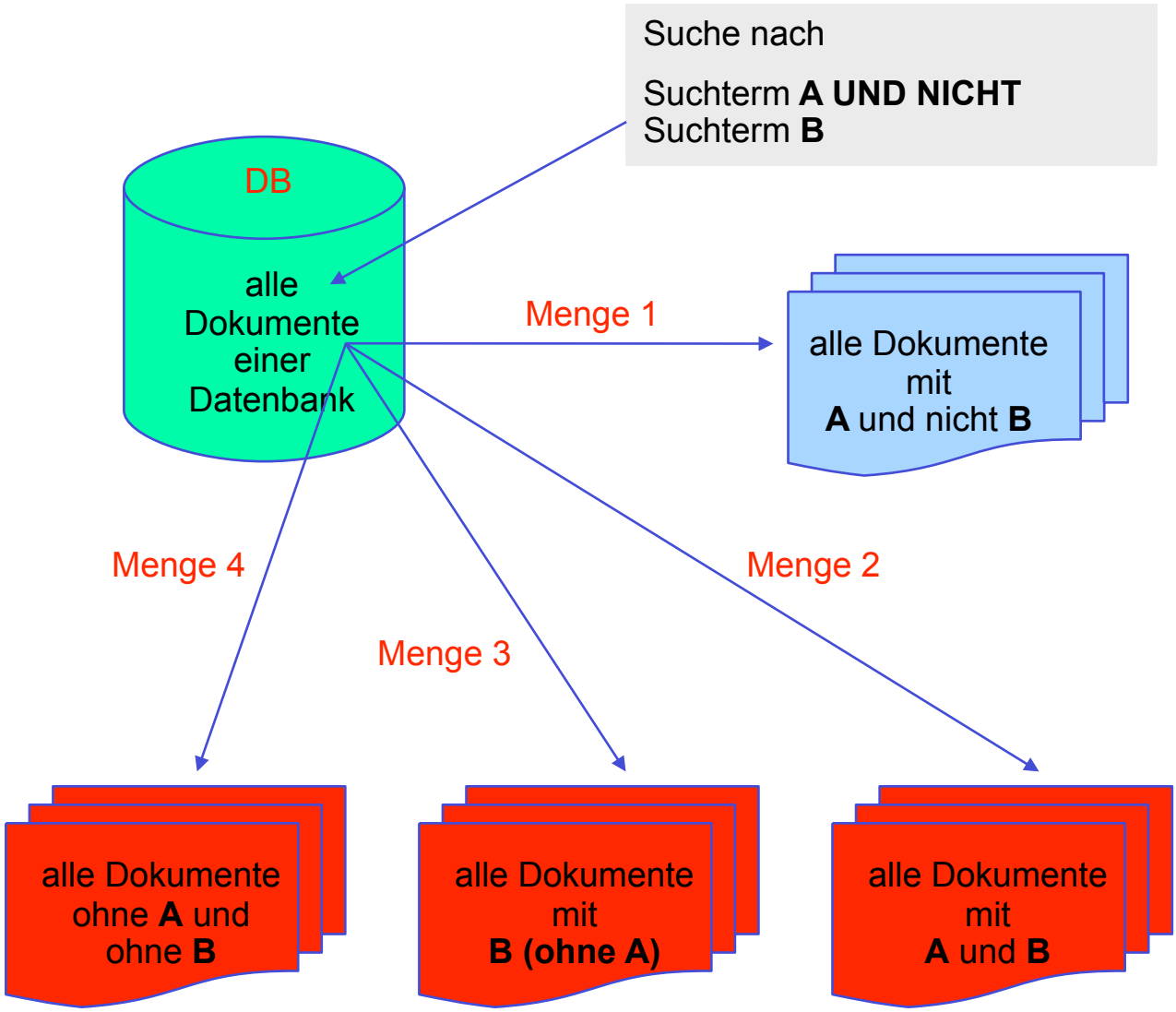
- Suchterm A (water)
- **und nicht** Suchterm B (plants)

enthalten, d.h. nicht gefunden werden alle Dokumente, die

- Suchterm A **und** Suchterm B enthalten
- **nur** Suchterm B enthalten
- **weder** Suchterm A **noch** Suchterm B enthalten

Das logische **NOT** entspricht dem sprachlichen "und nicht"

Achtung: Das boolesche NOT führt zu einem strengen Ausschluss von Begriffen von der Suche und verkleinert Treffermengen rapide, auch mit unerwünschten Folgen



Verwendung des booleschen **NOT**

- **Ausschluss eine Begriffs** von der Suche zur Spezifizierung des Suchergebnisses, z.B.
    - KFZ **NOT** Ford
- **Reduzierung einer Treffermenge** durch Ausschluss nicht gewünschter Dokumente

Kombination boolescher Operatoren

Die Verknüpfung von mehr als zwei Suchtermen erfordert die Festlegung einer Reihenfolge bei der Interpretation der booleschen Operatoren:

• KFZ **NOT** Ford **OR** Fiat

ist logisch mehrdeutig, d.h. zwei Lesarten sind möglich:

• (KFZ **NOT** Ford) **OR** Fiat oder KFZ **NOT** (Ford **OR** Fiat)

Lösung

1. Festlegung einer Reihenfolge bei der Abarbeitung, allg.

NOT vor **AND** vor **OR**

1. Klammerung (s.o.)

Übung: Boolesche Logik

1. **Titel (10)**: ¬The state of retrieval system evaluation.
   Quelle (25): Information processing and management.
2. **Titel (10)**: Order-theoretical ranking.
   Quelle (25): Journal of the American Society for Information Science.
3. **Titel (10)**: Introduction to information storage and retrieval systems.
   Quelle (25): Information retrieval: data structures and algorithms.
4. **Titel (10)**: Machine learning for information retrieval: neural networks, symbolic learning, and genetic algorithms.
   Quelle (25): Journal of the American Society for Information Science.
5. **Titel (10)**: Evaluation of information retrieval systems: approaches, issues, and methods.
   Quelle (25): Annual review of information science and technology.

- Formulieren Sie eine Suchanfrage mit 5 Suchbegriffen, die nur Titel 2 und 4 findet.
- Formulieren Sie eine Suchanfrage mit AND, OR und NOT, die nur Titel 4 findet.
- Formulieren Sie eine Suchanfrage mit 5 verknüpften, aber beliebigen booleschen Operatoren, die Titel 2, 3 und 5 findet.

B. Umgebungssuche (Proximity Search)

Autor(en) (5):	Crestani, F. (Hrsg.)\AUT Pase, G. (Hrsg.)
Titel (10):	Soft computing in information retrieval: techniques and applications
Bibl. Angaben (15):	Heidelberg: Springer, 2000. XII,394 S.
ISBN (20):	3-7908-1299-4
Jahr (30):	2000
Serie (35):	Studies in fuzziness and soft computing; vol.50
Abstract (40):	Presented are a number of advanced models for the representation and retrieval of information originating from the application of soft computing techniques to information retrieval. The book is a collection of articles from some of the most outstanding and well known researchers in the area of information retrieval

Die sog. Umgebungssuche ist eine Restriktion des logischen AND, indem zusätzlich ein maximaler Abstand zwischen zwei (oder mehr) Suchtermen festgelegt wird.

"**information . . retrieval**" entspricht

"information AND retrieval (+ Term A höchstens zwei Wörter vor oder nach Term B)"; ohne Festlegung der Reihenfolge

Vor- und Nachteile

- **Erhöhung der Genauigkeit der Suchanfrage** gemäß der Hypothese, dass zwei (oder mehr) Suchterme in enger Nachbarschaft mit größerer Wahrscheinlichkeit ein Hinweis auf das Thema des Dokuments sind
- **Vermeidung des feldübergreifenden Matchings**
- **Gefahr der zu engen Suche**, wenn zufällig mehr Begriffe zwischen den Suchtermen A und B stehen

Varianten

- **Nachbarschaftssuche (Adjaceny Search)**
  schränkt Proximity auf unmittelbare Nachbarschaft und eindeutige Richtung ein: "information retrieval" (vgl. C. Phrasensuche)
- **Feldbezogene Umgebungssuche**
  Suchterme müssen im gleichen Feld stehen
- **Satzbezogene Umgebungssuche**
  Suchterme müssen im gleichen Satz stehen

C. Phrasensuche

Autor(en) (5):	Blair, D.C.
Titel (10):	STAIRS Redux: thoughts on the STAIRS evaluation, ten years after
Quelle (25):	Journal of the American Society for Information Science. 47(1996) no.1, S.4-22.
Jahr (30):	1996
Abstract (40):	The test of retrieval effectiveness performed on IBM's STAIRS and reported in 'Communications of the ACM' 10 years ago, continues to be cited frequently in the information retrieval literature. The reasons for the study's continuing pertinence to today's research are discussed, and the political, legal, and commercial aspects of the study are presented. In addition, the method of calculating recall that was used in the STAIRS study is discussed in some detail, especially how it reduces the 5 major types of uncertainty in recall estimations. It is also suggested that this method of recall estimation may serve as the basis for recall estimations that might be truly comparable between systems

Die sog. Phrasensuche sucht mehrere Begriffe als exakte Wortfolge. Die Eingabe

"information retrieval literature"

sucht alle Zeichenketten (Strings) mit exakt dieser Wortfolge. IR-Systeme realisieren/simulieren die Phrasensuche manchmal mit der wiederholten Adjacency Search:

information ADJ retrieval ADJ literature

Vor- und Nachteile der Phrasensuche

- **Erhöhung der Genauigkeit** der Suchanfrage gegenüber der Umgebungssuche
- **Suchmöglichkeit für feststehende Wendungen** ("Regeln für den Schlagwortkatalog")
- **Gefahr des Ausschlusses von potenziellen Treffern**, weil z.B. Varianten einer Phrase existieren ("Deutsche Bundesbank" – der "Deutsche**n** Bundesbank")

Grundsätzlich ist die Phrasensuche ein Sonderfall des einfachen **Matchings**, indem die Zeichenkette mehrere Wörter und Leerzeichen umfassen kann.

Die Simulation von Phrasensuche durch mehrfaches Adjacency ist eine **Operatorsuche** mit mehreren Kriterien.

3. Bedingungen des Information Retrieval
3.1 Datenstrukturen und Indexaufbau

Begriffe und Bedeutungen

Index	→	Suchregister einer Datenbank
Indexierung		Aufbau eines Suchregisters
Indexierung		Inhaltserschließung
Indexing !	→	Inhaltserschließung (Vergabe von Deskriptoren)
Volltextindexierung		Aufbau eines Suchregisters über den gesamten Quelltext einer Datenbank
Invertierung		s. Indexierung
Freitextsuche		Suche im wahlfreien Zugriff in einer Datenbank
Volltextsuche		Suche im gesamten Quelltext einer Datenbank (Indexsuche oder freie Suche)

Suchverfahren in IR-Systemen (Zeichenkettensuche)

- **Sequenzielle Suche**
  Durchsuchen der gesamten Datenbank (vom 1. Wort des 1. Datensatzes bis zum letzten Wort des letzten Datensatzes) nach einer Zeichenkette
    - **schlechte Performance bei wachsendem Volumen!**
- **Indexsuche**
  Durchsuchen eines alphabetisch sortierten Suchregisters, das alle oder eine Teilmenge aller Zeichenketten aller Datensätze der Datenbank enthält
    - sehr gute Performance durch Zugriff auf sortierte Menge
    - ggf. eingeschränkte Suchmöglichkeiten (z.B. wenn Dokumente nicht vollständig indexiert sind)
    - bedarf Techniken für Indexaufbau und -aktualisierung

Autor: van de Rak, Jan Willem
Titel: Zwischen Pleonasmus und Fassette: Das Regelwerk als sinnstiftendes Element in Zeiten erschließerischer Verrohung.
Ort: Normstett
Jahr: 1998.

Schlagworte: *Regelwerk ; Norm ; soziokulturelle Studie*

Abstract: Die Arbeit untersucht die Bedeutung von Regelwerken vor dem Hintergrund des weltweit zu beobachtenden Niedergangs der Erschließungskultur im späten 20. Jh. Lösungsmöglichkeiten sieht der Autor in einer deutlich weitergehenden Reglementierung aller Bereiche des täglichen Lebens.

20		Niedergangs
aller		*Norm*
als		Pleonasmus
Arbeit		Regelwerk
Autor		*Regelwerk*
Bedeutung		Regelwerken
beobachtenden		Reglementierung
Bereiche		sieht
das		sinnstiftendes
dem		*soziokulturelle*
der	(2)	späten
des	(2)	*Studie*
deutlich		täglichen
die	(2)	und
einer		untersucht
Element		Verrohung
erschließerischer		von
Erschließungskultur		vor
Fassette		weitergehenden
Hintergrund		weltweit
im		Zeiten
in	(2)	zu
Jh		Zwischen
Lebens		
Lösungsmöglichkeiten		
...		

Prinzip der invertierten Liste (Inverted File)

Indexterm	Dok-Nr.	[Treffer]	[Position]	[Feld]
Autor	24, 35, 476, 8790	4	24 (67) 35 (6, 5543) 476 (20) 8790 (854)	Abs Tit, Abs Tit Abstract
Bedeutung	7, 17432	2	7 (17, 45, 96) 17432 (9)	Tit, Abs, Abs Tit
beobachtenden	35, 97, 3425	3	35 (5) 97 (3345) 3425 (17)	Tit VT Tit
Bereiche	2100, 4526	2	2100 (54) 4526 (345)	Abs VT

Eigenschaften des Inverted File

- ermittelt die **Zahl** der mit einem Indexterm (Suchbegriff) verknüpften Dokumente

Autor → 4 Treffer

- und deren **Dokumentnummern**

Bedeutung → Dok. 7 u. 17432

- erlaubt eine **feldspezifische Suche**

Bereiche (Abs) → Dok. 2100

- erlaubt eine **Umgebungssuche** durch Speicherung der Position von Indextermen

beobachtenden ADJ Autor → Dok. 35, Pos. 5 u. 6

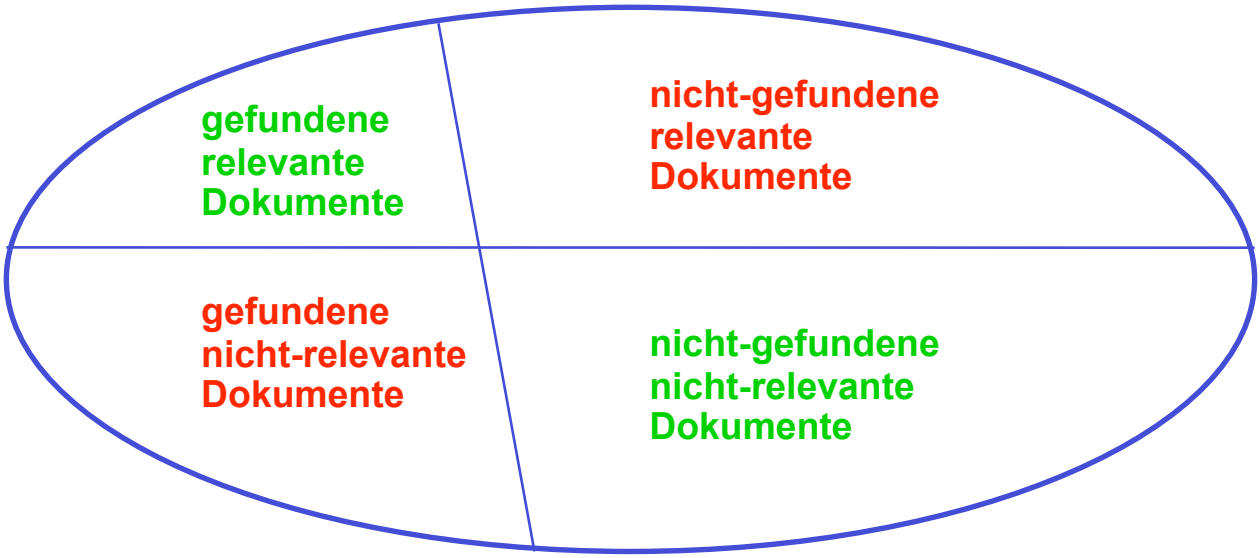
- ist allerdings den gleichen Bedingungen und Restriktionen des String-Matching unterworfen wie das Freitextretrieval

3.2 Bestimmung des Sucherfolgs – Recall und Precision

Zwei (oder drei) Ziele des Information Retrieval

1. Ermittlung aller relevanten Informationen/Dokumente durch die Suche in einer Datenbank
2. Reduzierung der nicht benötigten Informationen/Dokumente in einer Suche
3. Realistisch: Ermittlung so vieler relevanter Dokumente wie möglich und so wenig nicht-relevante Dokumente wie möglich/ nötig

Effekt einer Suche auf den gesamten Dokumentenraum



Das Maß für 1., d.h. das Maß für die Ausbeute einer Suche heißt

$$\textbf{Recall} = \frac{\text{gefundene relevante Dokumente}}{\text{alle relevanten Dokumente*}}$$

* = gefundene relevante Dokumente + nicht-gefundene relevante Dokumente

Das Maß für 2., d.h. das Maß für den Ballast bei einer Suchanfrage heißt

$$\textbf{Precision} = \frac{\text{gefundene relevante Dokumente}}{\textbf{alle gefundenen Dokumente*}}$$

* = gefundene relevante Dokumente + gefundene nicht-relevante Dokumente

Die Ergebnisse von Retrievaltests belegen ein typisches Verhältnis, eine **gegenseitige Abhängigkeit** zwischen den beiden Werten **Recall** und **Precision**, die

Inverse Relation zwischen Recall und Precision, d.h.

- Maßnahmen zur Erhöhung des Recalls bewirken eine Reduktion der Precision,
- Maßnahmen zur Erhöhung der Precision bewirken ein Absinken des Recalls,

Beispiele:

- eine Suche, die alle Dokumente einer Datenbank findet, erzielt 100% Recall bei einer Precision nahe bei 0,
- eine Suche, die ein relevantes Dokument von 100 in der Datenbank vorhandenen relevanten Dokumenten findet, erzielt 100% Precision bei einem Recall von 1%.

Folgerungen:

- Ziel der Entwicklung sollten IR-Systeme sein, die für eine möglichst große Zahl von Suchen guten Recall bei akzeptabler Precision ermöglichen,
- Maßnahmen zur Verbesserung der Suchmöglichkeiten lassen sich entweder als Recall- oder als Precision-verbessernde Maßnahmen kennzeichnen (beides gleichzeitig geht nicht!):
  - Trunkierung (s.u.) = Recall-erhöhend
  - Phrasensuche = Precision-erhöhend

Übung: Bedingungen des Information Retrieval

1. Welche Vorteile bietet die Vollinvertierung einer Datenbank? Welches Problem löst ein solcher Index nicht?
2. Bei einer Suche werden 60 Dokumente gefunden. 150 zum Thema relevante Dokumente befinden sich in der Datenbank. Die Precision beträgt 25%.
    - Wieviele nicht-relevante Dokumente befinden sich in der Treffermenge?
    - Wie hoch ist der Recall?
    - Welche Werte verändern sich in welcher Form (mit großer Wahrscheinlichkeit), wenn mit einer zweiten Suche der Recall verdoppelt wird?
3. In welcher Beziehung stehen die Ihnen bisher bekannten Suchfunktionen zu den Kriterien "Recall" und "Precision"?

Übung: Einführung Suchfunktionen II

Eine textbasierte Datenbank bietet für die Dokumentensuche folgende Suchmöglichkeiten:

- Boolesche Suche
- Proximity Search
- Phrasensuche

Für eine verbesserte Version sollen folgende Suchmöglichkeiten funktional unterstützt werden:

1. Eine Suche findet "recognize" **und** "recognise"
2. Die Suche nach "Haus" findet
    1. "Haus", "hausen", "Hauswirtschaft", "Hausierer"
    2. nur "Haus", "Hauses"
3. Die Suche nach "Bibliographie" findet auch "Bibliografie" und "bibliography" **aber nicht** "Biographie"
4. Die Suche nach "KFZ" findet auch "Auto"
5. Die Suche nach "Lohnnebenkosten" findet auch "Lohnpolitik"
6. Die Suche nach "Gewerkschaft" findet auch "Lohnrunde"

- Charakterisieren Sie die in 1. bis 6. gezeigten Probleme beim Information Retrieval.
- Entwickeln Sie jeweils eine Lösungsmöglichkeit und beschreiben Sie die Wirkungsweise der Funktion.
- Stellen Sie Vor- und Nachteile der "neuen" Funktionen gegenüber.

4. Funktionen von IR-Systemen II
4.1 Suchfunktionen II

Maskierung

lockert die Bedingungen des exakten Matchings auf Zeichenebene, indem das Maskierungszeichen an beliebiger Stelle im Wort einen (oder keinen) Buchstaben ersetzt:

aufw#ndig	findet	"**aufwendig"** und "**aufwändig"**
Bibliogra##ie	findet	**"Bibliographie, Bibliografie"**

Vor- und Nachteile

- abweichende Schreibweisen können bei der Suche berücksichtigt werden
- starke Nähe zum ursprünglichen Suchbegriff
- erfordert Kenntnisse über Wortalternativen

Trunkierung

lockert die Bedingungen des exakten Matchings auf der Ebene der Zeichenkette, indem das Trunkierungszeichen am Wortende und (seltener) am Wortanfang Zeichenketten beliebiger Länge ersetzt:

Haus?	findet u.a.	**Haus, Hauses, Hausmann, Hauswirtschaftslehrbuch**
?haus	findet u.a.	**Reihenhaus, Mietshaus, Mehrfamilienhaus, Lebkuchenhaus, Frauenhaus**
?haus?	findet u.a.	**Reihenhaustür, Chaussee**

Vor- und Nachteile

- Einbeziehung von Varianten durch Wortbildung am Wortende
- Einbeziehung von Komposita im Deutschen
- erfordert *sehr gute* Kenntnisse der Sprache

Fuzzy Search

erweitert das exakte Matching um die Möglichkeit, beliebige Abweichungen vom Suchbegriff (in definierbarer Zahl) zuzulassen

Wirtschafft	findet auch	**Wirtschaft**
Wortscafft	findet auch	**Wirtschaft**
Wort	findet auch	**Wirt**

Vor- und Nachteile

- Schreibfehler werden zuverlässig abgefangen
- Maskierung ist nicht mehr nötig
- erfordert kein Nachdenken
- Wörter mit "ähnlichen" Zeichenketten werden in die Suche einbezogen

Thesaurussuche

erlaubt die Einbeziehungen von Wortrelationen in die Suche, indem auf ein zuvor festgelegtes Vokabular zurückgegriffen wird

Bücherei	findet auch	**Bibliothek, Leihbücherei**	**[Synonym]**
Bibliothek	findet auch	**Hochschulbibliothek,Spezialbibliothek**	**[Unterbegriff]**

Vor- und Nachteile

- überwindet die engen Grenzen des exakten Matchings durch Vokabularunterstützung
- ermöglicht kontrollierte Einengung und Ausweitung von Suchergebnissen
- erfordert Vokabularfestlegung / Terminologiearbeit
- sollte idealerweise auf der Dokumentenebene unterstützt werden (Erschließung)

Konzeptsuche (concept search)

überwindet die Beschränkungen des exakten Matchings auf Wortebene durch Ausweitung auf ein umfassenderes "Suchthema" (Begriff, Konzept, Topic).

Realisierung erfolgt über zuvor definierte Themen mit zugeordneten Wortfeldern:

Wirtschaft: Ökonomie (Syn.), Volkswirtschaft (UB), Wirtschaftskriminalität (UB) ...

Vor- und Nachteile

- bietet Vokabularunterstützung
- erlaubt systematische Ausweitung und Einschränkung der Suche
- bedarf vorheriger Wortfelddefinition
- Wortfelder sind datenbankabhängig
- sollte auf der Dokumentenseite unterstützt werden

Natürlichsprachige Suche

vermeidet die Schwieirigkeiten, die beim exakten Matching durch strenge Suchsyntax entstehen. Natürlichsprachige Suche soll eine nahe an der Nutzersprache liegende Sucheingabe erlauben.

Beispiel:

"Ich interessiere mich für Literatur zum Thema **Information Retrieval**, die sich mit **Suchmaschinen** befasst, allerdings suche ich nichts zu **Northern Light**"

Sucheingabe wird dann linguistisch analysiert und (in der Regel) in eine Suchanfrage für exaktes Matching umgesetzt:

(**"Information Retrieval"** and **Suchmaschine**) not **Northern Light**

Vor- und Nachteile

- echte Hilfe insb. für unerfahrene Nutzer
- erfordert linguistische Komponente auf Retrieval- und idealerweise auch auf Dokumentenebene

5. Von der Zeichenkette zum Volltext

Dem Retrieval mit exaktem Matching unterliegt (implizit) **Hypothese I**:

"Das Vorkommen einer Zeichenkette in einem Datensatz ist ein hinreichendes Kriterium für seine Relevanz im Hinblick auf das durch die Such-Zeichenkette formulierte Thema."

Die Hypothese ist für das Retrieval in bibliografischen Datenbanken plausibel, weil

- bibliografische Datenbanken (vorwiegend) kategorisierte Inhalte in (mehr oder weniger streng) normierter Form enthalten,
- formale und inhaltliche Dokumentbeschreibungen in bibliografischen Datenbanken für die Zwecke des Wiederauffindens optimiert sind,
- formale und inhaltliche Dokumentbeschreibungen die im Dokument vorliegende Information extrem verdichten (Monografie > Katalogisat + Notation + Schlagwörter)

Lancaster-Retrievaltest (1991)

Rahmenbedingungen:

- Online-Katalog mit 4,5 Mio Nachweisen
- 51 Themen (v.a. komplexe (d.h. verknüpfte) Sachverhalte)
- zu findende (relevante) Dokumente wurden über umfassende Bibliografienarbeit bzw. Expertenbefragung vorher festgelegt
- Suchen wurden von LCSH-Experten durchgeführt!
- 607 relevante Nachweise insgesamt in der Datenbank
- 327 gefundene relevante Nachweise über aller Suchen
- Recall 53,9% über alle Suchfragen bei Suche mit LCSH (Achtung: systembedingt zu hoher Wert)
- Precision wurde nicht gemessen

Ergebnisse des Lancaster-Tests

- Suche über Erschließung (LCSH) 53,9%
- Einbeziehung eng verwandter Suchbegriffe 60,1%
- Einbeziehung verwandter Suchbegriffe 62,3%
- Einbeziehung von Titelstichwörtern 55,5%

Erweiterung der Titelaufnahme um Begriffe aus

- Sachregistern 74,5%
- Inhaltsverzeichnissen 68,0%
- Volltexten 63,4%

? Diskutieren Sie die Ergebnisse des Lancaster-Tests hinsichtlich

- Erschließungsqualität
- Retrievalqualität in OPACs
- der dem Matching unterliegenden Hypothese

The conclusion that emerges most clearly is that, if one wants to know the best things to read on some topic, there is no substitute for consulting an expert, either directly or indirectly (e.g. through an expert-compiled bibliography).

Lancaster u.a.: Identifying Barriers to Effective Subject Access in Library Catalogs, LRTS 35(1991), S. 388.

Hitzenberger (1981): Vergleich von Schlagwörtern und Titelstichwörtern bei 1163 Titeln des Bayerischen Verbunds

Formale Analyse

- 44,9%: Übereinstimmung von Hauptschlagwort und Stichwort
- 12,5%: Übereinstimmung von HSW und Grundform des Stichworts
- 25%: teilweise Übereinstimmung von Schlagwort und Stichwort
- 17,6%: keine Übereinstimmung zwischen Schlag- und Stichwort

Inhaltliche Analyse

- 17,8%: mehr Information durch Schlagwort als durch Stichwort
- 36,9%: gleiche Information durch Schlag- und Stichwort
- 45,3%: mehr Information durch Stichwort

? Diskutieren Sie die Ergebnisse der Untersuchung von Hitzenberger v.a. im Hinblick auf die Rolle von Stichwörtern für das Retrieval.

? Welche Konsequenzen zur Verbesserung des Retrievals legen die Ergebnisse von Lancaster und Hitzenberger nahe?

Ein Volltext(fragment):

Informationssuche im Internet

Angesichts der im Internet verfügbaren Datenmengen ist die Art und Weise des Zugriffs auf die Information der entscheidende Faktor bei der Nutzung von Internet-Ressourcen. Die zur Zeit verfügbaren Alternativen sind allgemein bekannt: Unter dem Sammelbegriff Suchmaschinen versuchen diverse Indexierungs- und Retrieval-Softwares, dem Problem der Quantität mit brutaler Rechenleistung zu begegnen. Suchmaschinen durchsuchen - so vollständig wie möglich -die verfügbaren Internetquellen und indexieren diese - mehr oder weniger vollständig -, d.h. stellen im Text vorkommende Begriffe für eine Suche zur Verfügung. Der Vorteil dieser Methode ist die prinzipielle Verfügbarkeit des gesamten Datenbestandes, denn jede indexierte Quelle kann, eine richtige Suche vorausgesetzt, gefunden werden. Der Vorteil der Suchmaschinen ist jedoch gleichzeitig ihr Nachteil, denn die große Zahl verfügbarer Internetquellen sorgt bei vielen Suchen für nicht mehr praktikable Ergebnismengen, die nicht selten mehr als 10.000 Nachweise anbieten und dadurch die Trennung zwischen Treffer und Nicht-Treffer in einen wenig erfolgversprechenden intellektuellen Suchprozeß münden lassen.

? Analysieren Sie die im Volltext vorkommenden Begriffe auf ihre Tauglichkeit als Suchbegriff im Sinne der Matching-Hypothese.

5.1 Wörter und deren Häufigkeiten

Einige Überlegungen:

- Information Retrieval auf Volltexte kann nicht von der Matching-Hypothese ausgehen, weil nicht alle vorkommenden Zeichenketten Dokumentrelevanz haben (können).
- Es besteht ein Zusammenhang zwischen der Auftretenshäufigkeit von Wörtern und deren Bedeutung für das Retrieval.
- Wichtig sind diejenigen Wörter, die
    - Dokumente hinreichend signifikant vertreten und gleichzeitig
    - von nicht-relevanten Dokumenten trennen.

Verteilung der Worthäufigkeit in Textkorpora: "Zipf's Law"

Worthäufigkeit * Häufigkeitsrang = Konstante

Worthäufigkeit: Auftretenshäufigkeit eines Wortes/Kollektion

Häufigkeitsrang: Position im Ranking nach Häufigkeit

Beispiel:

1. Häufigstes Wort	10.000
2. Zweithäuf. Wort	5.000
3. Dritthäuf. Wort	3.300
10.000. Zehn...	1

Wortverteilung in den Kollektionen von TREC-1 (1993)

Quelle	WSJ	AP	ZIFF	FR	DOE
Größe in MB	295	266	251	258	190
Mittelwert: Wörter/DS	182	353	181	313	82
verschiedende Wörter	156.000	198.000	174.000	126.000	186.000
einmaliges Auftreten	65.000	90.000	86.000	59.000	96.000
Auftreten > 1	199	174	165	106	159

5.2 Informationsstatistik

Hypothese II

"Die Häufigkeit eines Wortes ist über das reine Auftreten hinaus entscheidendes Kriterium für die Relevanz eines Dokuments in Bezug zum Suchterm."

? Betrachten Sie "Zipf's Law" und die Wortverteilung von TREC-1 und stellen Sie plausible Regeln für ein Retrievalmodell auf, das Worthäufigkeiten berücksichtigt.

Vermutungen

- hochfrequente Wörter sind schlechte Suchbegriffe
- niedrigfrequente Wörter sind schlechte Suchbegriffe, weil sie wahrscheinlich nicht zum Vokabular des Nutzers gehören und/oder autorenspezifisch sind

(1) Einfache Termhäufigkeit

Termhäufigkeit = Häufigkeit Term je Dokument

(2) Relative Termhäufigkeit (WDF)

WDF = Häufigkeit Term je Dokument / Gesamtzahl Terme

(3) Dokumenthäufigkeit

Dokumenthäufigkeit = Häufigkeit Dokumente je Term

(4) Inverse Dokumenthäufigkeit (IDF)

IDF = Termhäufigkeit bzw. WDF / Dokumenthäufigkeit

? Analysieren Sie das Verhalten der vier Berechnungsmodelle für seltene und häufige Terme und beurteilen Sie deren Fähigkeit im Hinblick auf Relevanzurteile.

Übung: Vergleich von Termgewichtungsverfahren

5 Dokumente aus einer Kollektion von 10.000

d1 = Anwendung des Prinzips Thesaurus für das Retrieval im OPAC

d2 = Zusammenhang zwischen Thesaurus und Klassifikation

d3 = Klassifikation und OPAC: verbesserter Sucherfolg durch Einsatz einer Klassifikation im Retrieval

d4 = Thesaurus für die Physik und Thesaurus für die physikalische Chemie

d5 = Klassifikation für die Chemie

Anzahl der Dokumente mit den Suchtermen

Anwendung	2000
Chemie	200
Einsatz	100
Klassifikation	100
OPAC	600
Physik	300
Prinzip	1500
Retrieval	400
Sucherfolg	50
Thesaurus	200
Zusammenhang	3000

? Berechnen Sie die inverse Dokumenthäufigkeit IDF für alle Suchterme (nur Substantive) in den Dokumenten.

Beispiel:
d1 = Anwendung (Gewicht), Prinzip (Gewicht), ...

? Berechnen Sie die Retrievalergebnisse für folgende Suchanfragen:

- **Thesaurus im Retrieval**
- **Klassifikation in der Chemie**

? Diskutieren Sie den Nutzen der Gewichtung im Hinblick auf das Retrieval, insb. die Ergebnisdarstellung

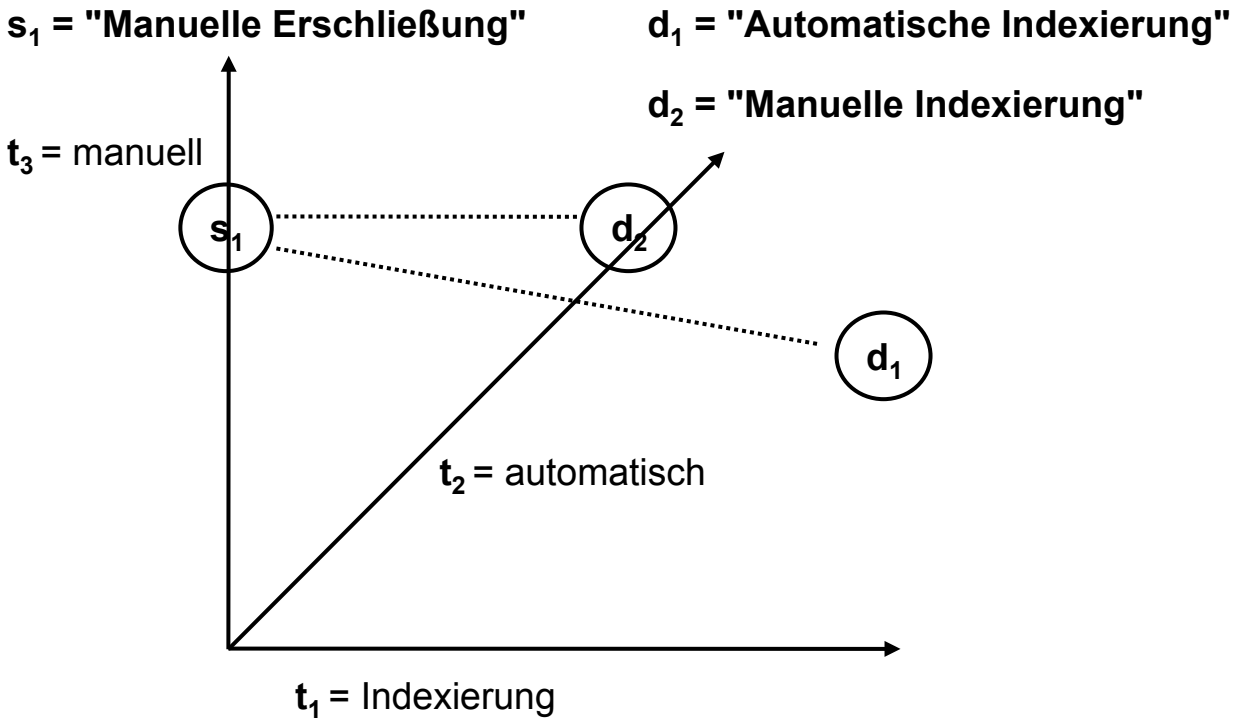
5.3 Ähnlichkeit als Relevanzmerkmal: Das Vektorraummodell

Hypothese III

"Relevanz eines Dokuments lässt sich als Ähnlichkeit zwischen dem Dokument und der Suchanfrage auffassen."

Modell

- Die (Index-)Terme eines Dokuments werden jeweils als Koordinaten in einem n-dimensionalen Vektorraum aufgefasst;
- Dokumente werden dadurch zu Punkten in diesem Vektorraum denen n Koordinaten zugeordnet sind;
- Die Terme der Suchfragen im Retrieval werden auch als Koordinaten aufgefasst, für die spezielle Frage ergibt sich dadurch ebenfalls ein Punkt im Vektorraum;
- Relevanz ergibt sich nun aus dem Abstand zwischen der Suchfrage und den Dokumenten – großer Abstand, wenig Relevanz, geringer Abstand, höhere Relevanz.



Abstrakte Form

d/s = (Termgewicht1, Termgewicht2, Termgewicht3)

Für das Beispiel:

d1 = (Termgewicht "Indexierung", Termgewicht "automatisch", 0)

d2 = (Termgewicht "Indexierung", 0, Termgewicht "manuell")

s1 = (0, 0, Termgewicht "manuell")

Für die Berechnung der Ähnlichkeit zwischen dem **Anfragevektor** und den **Dokumentvektoren** wird eine **Ähnlichkeitsfunktion** benötigt.

Bei Verwendung des **Skalarprodukts:** wenn *(a,b)* Anfragevektor ist und *(x,y)* Dokumentvektor ist, berechnet sich die Ähnlichkeit *c* zwischen beiden als

$$c = ax + by.$$

Bezogen auf das Beispiel:

*c von s1 und d2 = 0 + 0 + Termgewicht "manuell" * Termgewicht "manuell"*

? Berechnen Sie für die Dokumente aus 5.2 und die Suchanfrage

Thesaurus im Retrieval

die Ähnlichkeiten unter Verwendung des Skalarprodukts.

? Wie sieht das entsprechende Ranking aus?

? Lassen Sie für die Termgewichte nur 0 und 1 zu für Nicht-Vorkommen bzw. Vorkommen eines Terms. Es werden zwei Fälle unterschieden. Die Ähnlichkeit ist dabei wie folgt definiert:

(1) Falls das Skalarprodukt >= 1 ist, ist die Ähnlichkeit = 1, sonst ist die Ähnlichkeit = 0.

(2) Falls alle Werte des Skalarprodukts den Wert 1 haben, ist die Ähnlichkeit = 1, sonst ist die Ähnlichkeit = 0.

6. Automatisches Indexieren

Unter automatischem Indexieren versteht man ganz allgemein die Ermittlung (Extraktion) und/oder Zuordnung von Indextermen zu Dokumenten.

Mögliche Verfahrensweisen sind entweder

- **linguistisch basiert** oder
- **statistisch basiert**

Die Indexterme als Ergebnis linguistischer oder statistischer Analyse können entweder

- aus dem Dokument stammen, d.h. aus dem Text des Dokuments **extrahiert** sein
- oder aus einem getrennt vorliegenden Vokabular dem Dokument **zugeteilt** sein.

6.1 Linguistisch basierte automatische Indexierung

Matchingbasierte IR-Systeme haben auf der Ebene der Zeichenkette zwei Probleme:

- grammatikalische Varianten **eines** Wortes sind verschiedene Indexterme,
- für Wortgewichtungsverfahren sind grammatikalische Varianten ebenfalls verschiedene Terme, die dadurch getrennt gezählt werden.

Lösung bieten automatische Verfahren zur sprachlichen Vereinheitlichung; dabei unterscheidet man

1. **regelbasierte Verfahren**,
   d.h. Verfahren, die auf der Basis eines Regelsystems aus den im Text vorkommenden Wörtern normierte Indexterme generieren
2. **lexikonbasierte Verfahren**,
   die auf der Basis umfangreicher Wortlisten (Wörterbücher) im Text vorkommende Wörter identifizieren und ggf. grammatikalisch vereinheitlichen.

6.1.1 Regelbasierte Verfahren – Stemming

Der Einsatz eines regelbasierten Verfahrens macht nur dann Sinn, wenn die Quellsprache über eine im hohen Maße **regelhafte Wortbildung** verfügt, d.h.

- die Zahl der benötigten Regeln nicht zu hoch ist,
- die Zahl der zu erfassenden Ausnahmefälle nicht zu hoch ist.

Beide Bedingungen sind für das Englische erfüllt, für das Deutsche dagegen nicht.

Prinzipien regelbasierter Verfahren

- über ein **Set von Regeln** werden unterschiedliche Fälle von Flexionsendungen unterschieden mit dem Ziel, Endungen zu modifizieren oder zu entfernen,
- alle nicht über das Regelwerk erfassten Fälle werden explizit als Ausnahme in einer **Ausnahmeliste** geführt,
- für den Prozess entsteht dadurch die **Abarbeitungsreihenfolge**
  1. Versuch einer Identifizierung über Ausnahmeliste
  2. Anwendung des Regelwerks

Ziele

- Generierung von **grammatikalischen Grundformen** als Indextermen; Flexionsendungen werden entfernt, die Wortklasse bleibt erhalten (Lexikoneintrag):

retrieval, retrieve

- Generierung von **Wortstämmen** als Indextermen; Wortbildungsbestandteile (Derivate) werden entfernt, die Wortklasse geht verloren:

retriev

[Wortstämme und Grundformen können in manchen Fällen auch identisch sein: *sea*]

Einfaches Stemming-Regelwerk (Kuhlen/Knorz)

1.	**IES**	⇨	**Y**
2.	**ES**	⇨	**_** [wenn *O / CH / SH / SS / ZZ / X vorausgehen]
3.	**S**	⇨	**_** [wenn * / E / %Y / %O / OA / EA vorausgehen]
4.	**IES'**	⇨	**Y**
	ES'	⇨	**_**
	S'	⇨	**_**
5.	**'S**	⇨	**_**
	'	⇨	**_**
6.	**ING**	⇨	**_** [wenn ** / % / X vorausgehen]
	ING	⇨	**E** [wenn %* vorausgehen]
7.	**IED**	⇨	**Y**
8.	**ED**	⇨	**_** [wenn ** / % / X vorausgehen]
	ED	⇨	**E** [wenn %* vorausgehen]

% **= alle Vokale und Y**

***** **= alle Konsonanten**

_ **= Tilgung**

/ **= Oder**

Der vollständige Kuhlen-Algorithmus erreicht eine Fehlerquote unter 3 Prozent!

? Testen Sie das Regelwerk für folgende Beispiele; welche Regeln werden jeweils angewandt:

algorithms, associated, indexing, inverted, ladies', mother's, properties, satisfied, searches, using

? Entwerfen Sie einen Stemming-Algorithmus für Pluralendungen deutscher Substantive.

6.1.2 Lexikonbasierte Verfahren

Prinzipien

- die im Text vorkommenden Wörter werden über **Einträge in einem Wörterbuch** identifiziert; die Generierung von Stämmen bzw. Grundformen erfolgt ausschließlich auf der Basis dieser Einträge (Kein Eintrag ⇨ keine Aktion!)
- lexikonbasierte Verfahren arbeiten also mit **Positivlisten**, die erstellt und gepflegt werden müssen
- Aufwand für lexikonbasierte Verfahren ist dann angemessen, wenn
  - Regelwerke zu umfangreich und
  - Ausnahmelisten zu umfangreich würden,

  weil die zu bearbeitende Sprache zu geringe Regelhaftigkeit aufweist wie z.B. das Deutsche.

Funktionsweise

- **Lexikon** als
  1. **Vollformenlexikon**, d.h Lexikon enthält alle grammatikalischen Varianten und Verweise auf die Stamm-/Grundform;
  2. **Stamm-/Grundform-Lexikon**, d.h. es sind nur Stämme bzw. Grundformen mit möglichen/erlaubten Endungsformen im Lexikon verzeichnet.
- **Identifizierungsstrategie**
  - entfällt bei Vollformenlexika, da hier einfaches Matching zwischen Wortform im Text und Lexikoneintrag möglich ist;
  - für Stamm-/Grundform-Lexika z.B. sog. **"Longest-Matching-Strategie"**, d.h. lange Lexikoneinträge werden vor kurzen identifiziert.

Funktionen

- Grundformerzeugung (Lemmatisierung)

  Häuser ⇨ Haus

- Stoppworterkennung
- Zerlegung von Komposita (Dekomposition)

Haustürgriff ⇨ Haus, Tür, Griff

- Bildung von Wortableitungen

philosophisch ⇨ Philosophie

- Erkennung von Mehrwortgruppen

Zweites Deutsches Fernsehen

Das Morphologieprogramm MORPHY

- lexikonbasiertes System für das Deutsche
- Einträge als Stamm/Grundform + Merkmale

Beispiel: *Kuß, Klasse 4, ss/ß-Wechsel: JA, Plural: JA*

- Klassenzugehörigkeit regelt Endungsverhalten; 62 Klassen für Substantive
- Strategie: Einlesen von Rechts, Longest-Matching
  1. Abschneiden von Endungen bis zur Identifizierung
  2. Testen der Endungen für Wortstamm
- Beispiel:

```
@

Flüssen

Fluß SUB DAT PLU MAS
```

1. Identifizierung von "Fluß" durch Abschneiden von "-en"
2. Lexikoneintrag "Fluß" mit ss/ß-Wechsel und Umlautung im Plural

? Vergleichen Sie Vor- und Nachteile regelbasierter bzw. lexikonbasierter Indexierungssysteme

6.2 Statistisch basierte automatische Indexierung
6.2.1 Das Verfahren AIR/PHYS

Umgebung

Fachdatenbank PHYS (inzw. Bestandteil von INSPEC) mit **englisch-sprachiger** Erschließung durch **normiertes Vokabular** (Deskriptoren) und **Abstracts**

Ziel von AIR/PHYS

automatische Indexierung der Dokumente mit **Deskriptoren** des PHYS-Thesaurus

Realisierung

1. **statistische Auswertung** der intellektuell erschlossenen Dokumente: v.a. Untersuchung der Beziehung

$$\textit{Term} \Rightarrow \mathbf{z} \Rightarrow \textit{Deskriptor},$$

wobei **z** ein Maß für die Wahrscheinlichkeit ist, mit der ein *Deskriptor* einem Dokument (intellektuell) zugeteilt ist, wenn *Term* im Dokument vorhanden ist:

$$z = \frac{h(t,s)}{f(t)}$$

h(t,s) = Anzahl der Dokumente, in denen Term t vorkommt und Deskriptor s vergeben wurde

f(t) = Anzahl der Dokumente, in denen Tem t vorkommt

1. (automatischer) **Aufbau eines Indexierungswörterbuchs** unter Ausnutzung der Gewichte aus 1., echter Thesaurusrelationen (Synonym) und Deskriptor-Deskriptor-Relationen als gewichtetes Maß für das gemeinsame Auftreten von Deskriptoren

2. **Automatische Indexierung** in zwei Phasen

   - **Rohindexierung** mit regel- und lexikonbasierter Textanalyse und statistischer Relationierung
   - **Abgestimmte Indexierung** unter Einbeziehung von Deskriptor-Deskriptor-Relationen

Pilotanwendung AIR/PHYS

- Wörterbuchaufbau auf der Basis von 400.000 intellektuell erschlossenen Dokumente
    - 20.000 Deskriptoren
    - 190.000 Wörter
    - 350.000 statistische Regeln mit z > 0,3
    - 70.000 Synonym-Relationen
    - 200.000 Deskriptor-Deskriptor-Relationen
- Erschließung von 10.000 Dokumenten / Monat
- Zuteilung von im Schnitt 12 Deskriptoren je Dokument
- intellektuelle Nachbearbeitung mit durchschnittlich einem Drittel Korrekturbedarf, d.h. **semi-automatisches Verfahren**

Ergebnisse der AIR/PHYS-Indexierung

- **Retrievaltest** mit 15.000 Dokumenten und 300 (Original-)Fragen
    - automatische Indexierung
      Precision: 0.46
      Recall: 0.57
    - intellektuelle Indexierung
      Precision: 0.53
      Recall: 0.51
- **intellektuelle Bewertung** der Erschließungsqualität durch Experten
    - 1/3 intellektuelle Erschließung besser
    - 1/3 automatische Indexierung besser
    - 1/3 qualitativ gleichwertig

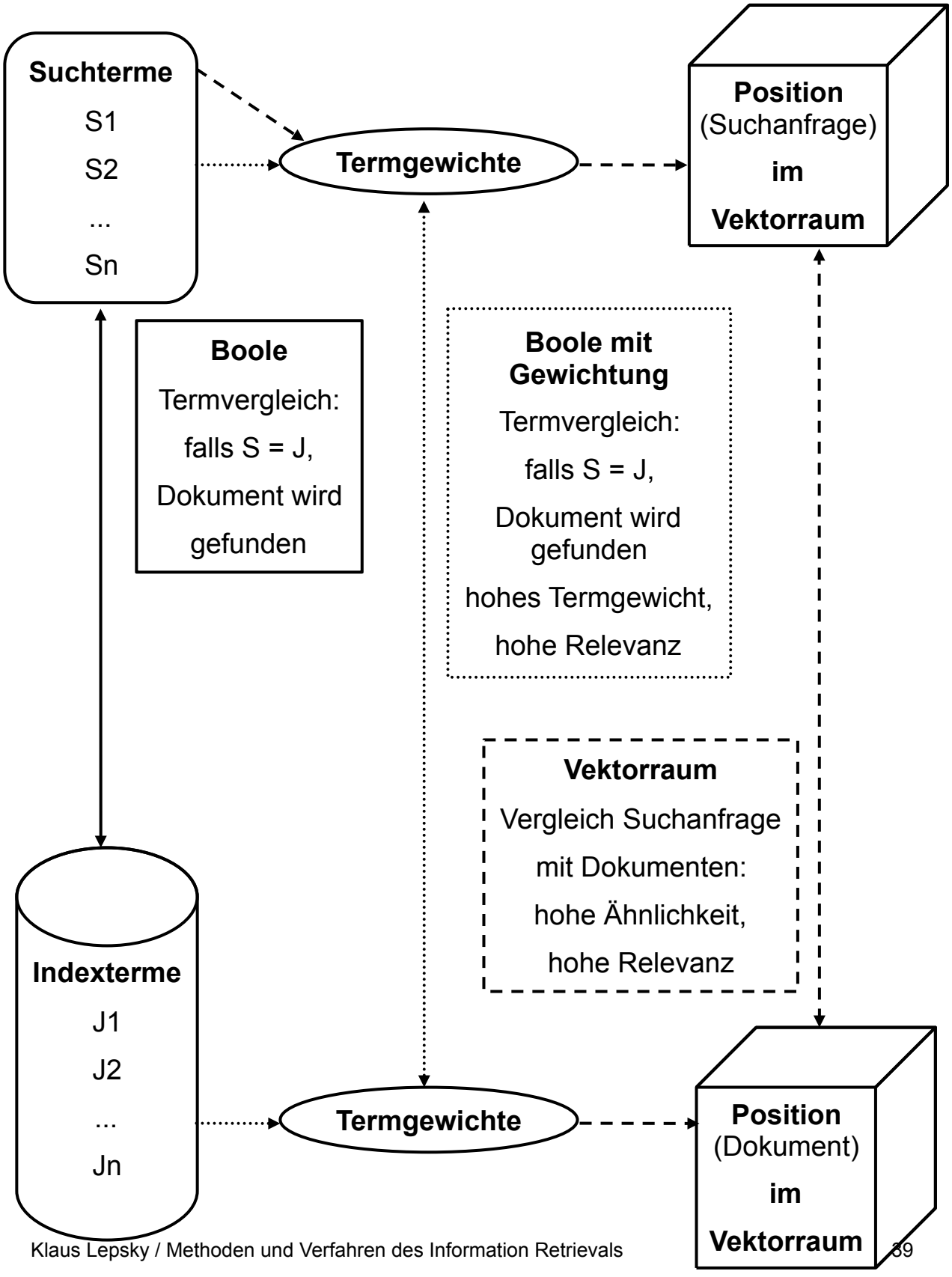
Literatur

Knorz, Gerhard: Automatische Indexierung. In: Wissensrepräsentation und Information Retrieval. Potsdam 1994. S. 138-198.

Nohr, Holger: Automatische Indexierung: Einführung in betriebliche Verfahren, Systeme und Anwendungen. Potsdam 2001. S.71-77.

7. Probabilistisches Information Retrieval
Relevance Ranking und Relevance Feedback

7.1 Retrieval-Modelle



7.2. Das probabilistische Retrievalmodell

Hypothese IV

„Ein Dokument ist hinsichtlich einer Suchanfrage relevant, wenn ein Nutzer dieses Dokument als Ergebnis der Suchanfrage als relevant einschätzen würde.“ (nach Maron/Kuhns)

Bestandteile des Modells

- Suchanfrage
- Dokumente
- Relevanz von Dokumenten hinsichtlich der Suchanfrage
- Wahrscheinlichkeit eines (positiven) Relevanzurteils durch den Nutzer
- gewichtete Indexterme
- ggf. gewichtete Suchterme

Modell:

„Die Wahrscheinlichkeit, dass ein bestimmtes Dokument *d* hinsichtlich einer bestimmten Suchanfrage *q* als relevant eingeschätzt wird, entspricht dem Verhältnis zwischen der Zahl der Nutzer, die *q* gesucht haben und *d* als relevant einschätzen und der Zahl der Nutzer, die insgesamt *q* gesucht haben.“ (nach Maron/Kuhns)

Verfahren

- **Ermittlung von Relevanzurteilen**, d.h. Aussagen darüber, welche Dokumente in Bezug auf welche Suchanfragen wie häufig als relevant eingestuft wurden
- Berechnung der Wahrscheinlichkeit, mit der ein Dokument hinsichtlich einer Suchanfrage relevant ist, d.h. Ermittlung von **Relevanzschätzwerten**

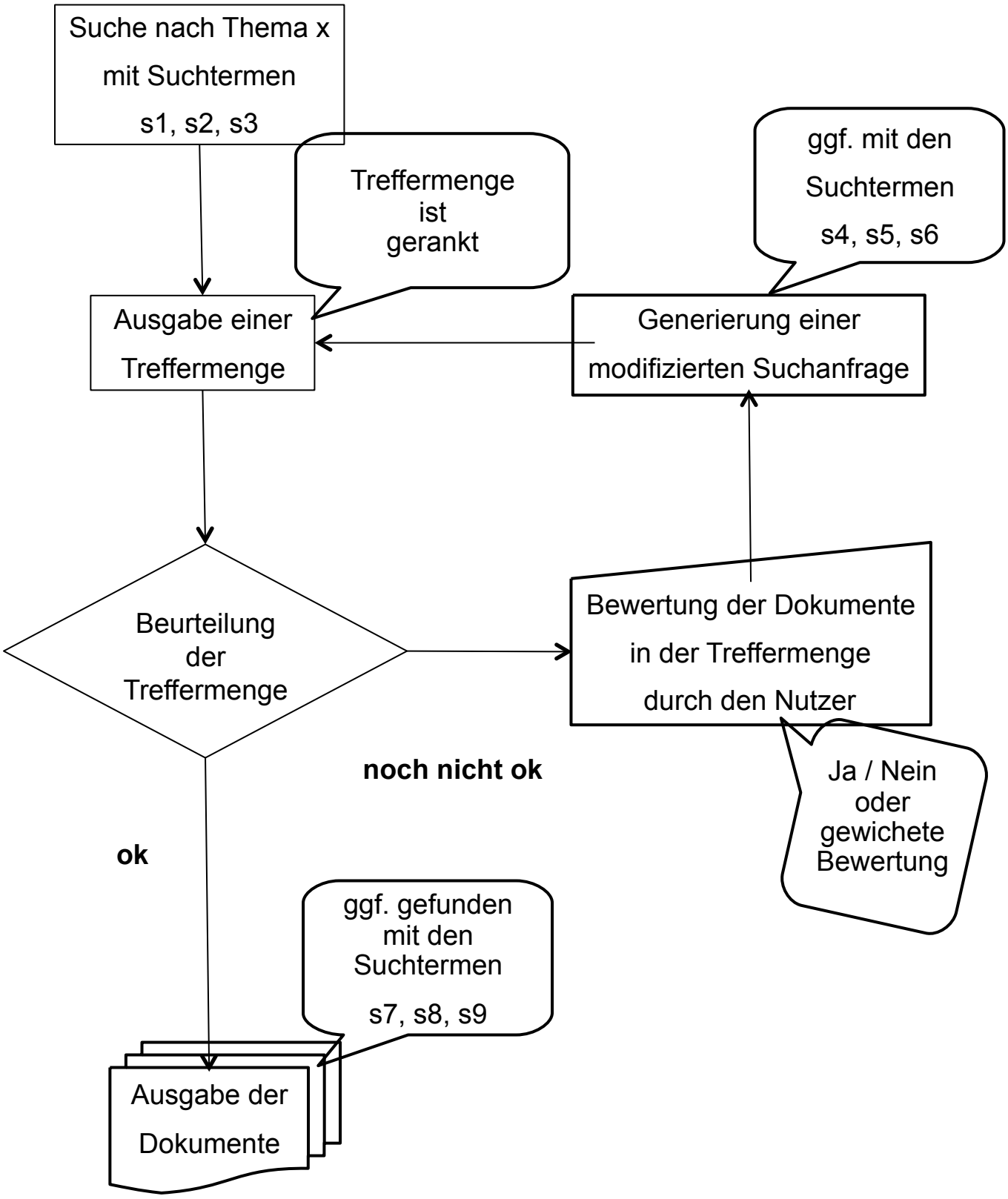
Problem

Relevanzurteile liegen für umfangreiche Dokumentsammlungen

in der Regel nicht vor!

7.3 Relevance Feedback

Relevance Feedback verwendet Relevanzurteile des Nutzers für die Suche nach relevanten Dokumenten.



Retrieval-Systeme und ihre Bestandteile

1. Die Kollektion *besteht aus entweder*

Einzeldokumenten

(z.B. HTML-Seiten, ASCII-Texten)

Dokumente sind für das IR-System alle Dateien eines oder mehrerer Verzeichnisse

oder aus

Datenbankfiles

(z.B. Grunddateien bibliografischer Datenbanksysteme)

Dokumente sind für das IR-System alle Einzeldokumente der Datenbankdatei, d.h. Beginn und Ende eines Dokuments müssen eindeutig bestimmt und dem System deklariert sein

2. Der Index

entsteht durch die Auswertung der Dokumente hinsichtlich der in ihnen enthaltenen Wörter bzw. Zeichenketten (Indexterme).
*Das Retrievalsystem ist in seiner Funktionalität unabhängig von der Kollektion. Basis für das Retrieval ist der Index (die **invertierte Liste**). Die Beziehung zur Kollektion wird über die Adresse in der invertierten Liste gewahrt.*

3. Funktionalitäten *sind zu unterscheiden in*

Vergleichsfunktionen

für den Vergleich von Zeichenketten (= Suchfunktionalitäten)

und

Indexbezogene Funktionen *(Automatische Indexierung)*

als Methoden zur Verbesserung von Indextermen hinsichtlich der Vergleichsfunktionen:

linguistisch basierte automatische Indexierung

zur sprachlichen Normalisierung von Indextermen

und

statistisch basierte automatische Indexierung

zur Gewichtung von Indextermen

8. Automatische Klassifizierung / Clustering

Ziel: **Strukturierung großer Dokumentmengen**

Zwei Ansätze:

- **Automatisches Klassifizieren**
  als Zuweisen von Dokumenten in vorgegebene Themen
- **Clustering**
  als Unterteilung einer Dokumentkollektion in Gruppen ähnlicher Dokumente (Cluster)

Automatisches Klassifizieren

Ausgangspunkt

Systematisch geordnete Themen / Klassifikation

Ziel

Zuordnung aller Dokumente einer Kollektion zu den Themen der Ordnung / Klassen der Klassifikation

Verfahren

Erstellen einer Testkollektion, d.h. intellektuelle Zuweisung von Dokumenten zu den Themen / Klassen

Analyse der Termbeziehungen in den Dokumenten einer Klasse, z.B. auf der Basis einer **Dokument-Term-Matrix** der gewichteten Terme:

	Term1	**Term2**	**Term3**	**Term4**	**Term5**	**Term6**	**Term7**	**Term8**
Dok 1	0	4	0	0	0	2	1	3
Dok 2	3	1	4	3	1	2	0	1
Dok 3	3	0	0	0	3	0	3	0
Dok 4	0	1	0	3	0	0	2	0
Dok 5	2	2	2	3	1	4	0	2

- Ermittlung der häufigsten gemeinsamen Terme einer Klasse
- Ermittlung der Häufigkeit dieser Terme in anderen Klassen
- Zuweisung der Terme zur Klasse, falls Terme in der Klasse häufig, in anderen Klassen jedoch selten sind

Ergebnis

Zuordnung von Termen zu Klassen

Klassifikationsverfahren

- Festlegung der Bedingungen, die zur Zuweisung eines Dokuments zu einer Klasse führen:
  - wie viele Terme einer Klasse müssen mindestens im Dokument enthalten sein
  - welche Gewichte müssen diese haben
- Termgewichtung für neue Dokumente
- Anwendung der Regeln
- Zuordnung eines Dokuments zu einer Klasse

Clustering

Ausgangspunkt

unstrukturierte, in der Regel sehr große Dokumentkollektion

Ziel

Strukturierung der Kollektion durch Ermittlung von Gruppen ähnlicher Dokumente

Verfahren

Berechnung der Ähnlichkeit von Dokumenten

- durch Analyse der Beziehungen zwischen Dokumenten und den in ihnen enthaltenen Termen
- und Festlegung eines Clustering-Algorithmus‘ für die Zuweisung von Dokumenten zu Clustern

Dokument-Term-Matrix,

d.h. welche Dokumente enthalten welche Terme mit welchem Gewicht

	Term1	**Term2**	**Term3**	**Term4**	**Term5**	**Term6**	**Term7**	**Term8**
Dok1	0	4	0	0	0	2	1	3
Dok2	3	1	4	3	1	2	0	1
Dok3	3	0	0	0	3	0	3	0
Dok4	0	1	0	3	0	0	2	0
Dok5	2	2	2	3	1	4	0	2

Erzeugung einer **Dokument-Dokument-Matrix** durch Berechnung der Skalarprodukte von jeweils zwei Dokumentvektoren

	Dok1	**Dok2**	**Dok3**	**Dok4**	**Dok5**
Dok1		11	3	6	22
Dok2	11		12	10	36
Dok3	3	12		6	9
Dok4	6	10	6		11
Dok5	22	36	9	11	

Erzeugung einer **Dokument-Beziehungs-Matrix** durch Festlegung eines Schwellenwertes (hier: 10)

	Dok1	**Dok2**	**Dok3**	**Dok4**	**Dok5**
Dok1		1	0	0	1
Dok2	1		1	1	1
Dok3	0	1		0	0
Dok4	0	1	0		1
Dok5	1	1	0	1	

Anwendung eines **Clusteralgorithmus‘** zur Verteilung der Dokumente auf Cluster

Clusteralgorithmen

- **Cliquen-Algorithmus**
  alle Dokumente eines Clusters sind allen anderen Dokumenten des Clusters ähnlich; Dokumente in einem Cluster haben die engstmögliche Beziehung zueinander – Dokumente eines Clusters repräsentieren ein Thema (Topic)
- **Single-Link-Algorithmus**
  jedes Dokument eines Clusters ist mindestens einem Dokument des Clusters ähnlich; Dokumente eines Clusters haben schwache Beziehung zueinander – Dokumente eines Clusters repräsentieren keine Themen
- **Varianten** zwischen beiden Extremen

Spielarten

(1) Verwendung von Startclustern und Berechnung von **Zentroiden**

- Festlegung von Clustern und beliebige Zuweisung von Dokumenten zu Clustern
- Berechnung eines Zentroids (d.h. eines Mittelwerts aller Dokumente eines Clusters)
- Berechnung der Ähnlichkeit zwischen den Dokumenten in den Clustern und den Zentroiden der Cluster und Neuverteilung der Dokumente in die Cluster
- Durchführung des Verfahrens bis zu stabilen Clustern

(2) **Hierarchisches Clustering**, z.B. durch

- iteratives Clustern von erzeugten Clustern bis hin zum einzelnen Dokument (Top-down)
- Berechnung von Zentroiden für die Cluster und Clustering der Zentroide (erzeugt erste hierarchisch höhere Ebene; Bottom-up)
- Fortführung des Prozesses bis zur gewünschten Hierarchie

Nutzen von Clustering im Information Retrieval

- **Termclustering**
  Clustering von **Termen** einer Kollektion erzeugt Mengen ähnlicher Begriffe, die für die automatische Erstellung thesaurus-ähnlicher Werkzeuge für die Suche verwendet werden können:
    - Ausweitung der Suche durch Einbeziehung ähnlicher Begriffe;
    - Verlassen der strengen Matching-Bedingungen im Zeichenketten-Retrieval;
    - Angleichung von Such- und Autorensprache;
    - Visualisierung von Begriffsbeziehungen.

- **Dokumentclustering**
  Clustering von Dokumenten einer Kollektion erzeugt Mengen ähnlicher Dokumente, die für die Suche verwendet werden können:
    - Ausweitung der Suche auf ähnliche Dokumente;
    - Strukturierung von Treffermengen (NorthernLight-Prinzip);
    - Visualisierung von Dokumentbeziehungen in Suchergebnissen;
    - Verlassen der strengen Matching-Bedingungen im Zeichenketten-Retrieval;
    - Relevance Feedback

Literatur:

Kowalski, Gerald J.; Maybury, Mark T.: *Information Storage and Retrieval Systems: Theory and Implementation*. Second Edition. Boston 2000.

Hier: Kapitel 6: *Document and Term Clustering*, S. 139-163.

9. Literatur

Ellis, D.: New Horizons in Information Retrieval.
London 1990.

Konzentrierte und verständlich geschriebene Einführung in IR jenseits von Boole. V.a. Kapitel 2: Statistical and probabilistic retrieval.

Ferber, Reginald: Information Retrieval. Suchmodelle und Data-Mining-Verfahren für Textsammlungen und das Web. Heidelberg 1993.

Überarbeitete Fassung des Skripts:

Ferber, Reginald: Data Mining und Information Retrieval. Skript zur Vorlesung an der Technischen Universität Darmstadt im Wintersemester 1999/2000.
http://www.darmstadt.gmd.de/~ferber/dm-ir

Sehr umfassendes Skript mit dem Schwerpunkt auf Data-Mining-Verfahren. Mit erheblichen mathematischen Hürden.

Fuhr,N.: Information Retrieval: Skriptum zur Vorlesung.
Universität Dortmund 1998. (Kapitel 1-5)
http://ls6-www.informatik.uni-dortmund.de/teaching/courses/ir

Teils deutsches, teils englisches Skript. Knapper als Ferber. Da Informatik-orientiert ebenfalls mathematisch anspruchsvoll.

Stock, Wolfgang: Information Retrieval: Informationen suchen und finden.
München 2007.

Umfassende Darstellung zum Information Retrieval.

Kowalski, G.; Maybury, Mark T.: Information storage and retrieval systems: theory and implementation. Second Edition
Boston, MA: Kluwer Academic Publ., 2000. XIII, 318 S.

Typisch amerikanisches Lehrbuch zum IR: gut und verständlich geschrieben, aktuell und recht erschöpfend.

Rijsbergen, C.J. van: Information retrieval.
London: Butterworths, 1979., 2nd ed.
http://www.dcs.glasgow.ac.uk/Keith/Preface.hmtl

Der Klassiker. Enthält die theoretische Fundierung zahlreicher, heute üblicher Verfahren des automatischen Information Retrieval. Anspruchsvoll.

Salton, G. und M. J. McGill*:* Information Retrieval: Grundlegendes für Informationswissenschaftler.
Hamburg: McGraw-Hill, 1987, 465 S.

Noch ein Klassiker. Einer der wenigen deutschen Texte.