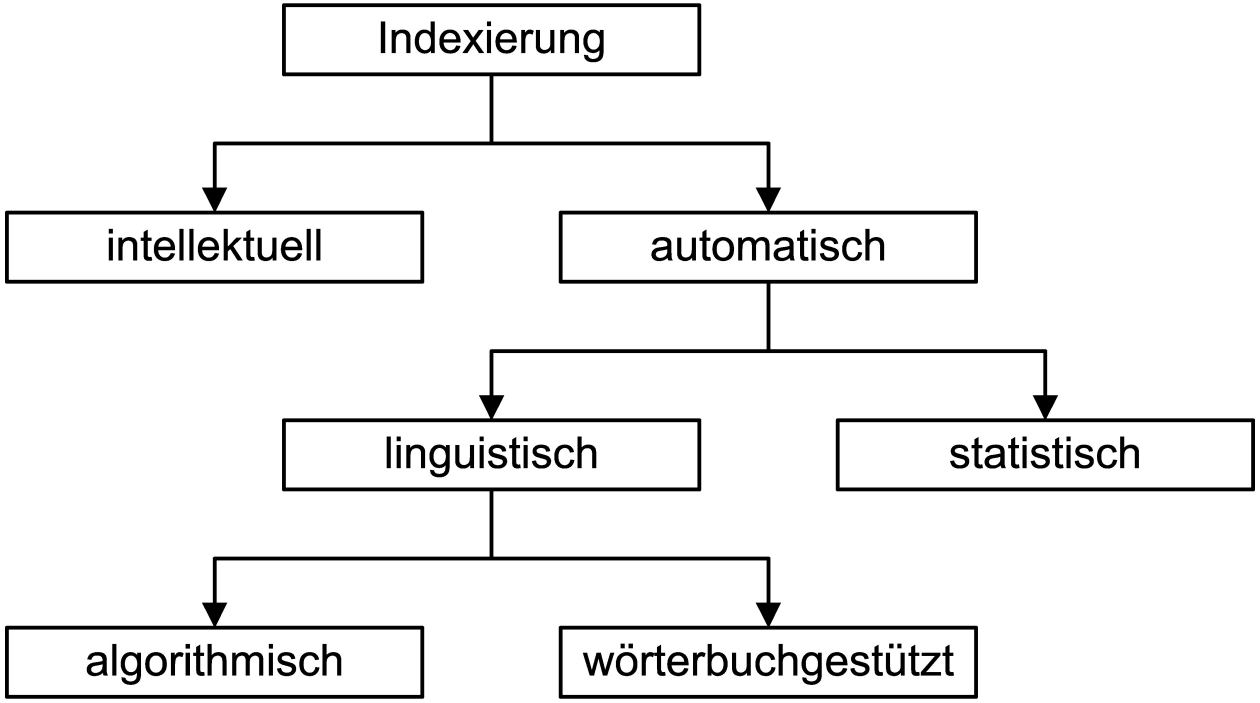
Automatisches Erschließen

Verfahren der Indexierung



Kapitel 5 - Abb. 5.1

Das Prinzip Index

Ein Satz, für uns bestehend aus Wörtern:

Für das IR-System besteht ein abgespeichertes Dokument nicht aus „Wörtern“ sondern lediglich aus einer Aneinanderreihung von Zahlencodes:

Speicherinhalt:

```
46C3BC72206461732049522D53797374656D20626573746568742065696E20616267657370656963686572746
57320446F6B756D656E74206E6174C3BC726C696368206E6963687420617573206E2809E57C3B6727465726E
E2809C20736F6E6465726E206C656469676C6963682061757320656965722041 6E65696E616E6465727265
6968756E6720766F6E205A61686C656E636F6465732E
```

Textmatching – Sequentielle Suche

Für das IR-System besteht ein abgespeichertes Dokument nicht aus „Wörtern“ sondern lediglich aus einer Aneinanderreihung von Zahlencodes:


```
46C3BC7220646173204952 2D53797374656D2062657374656874206569 6E20616267657370656963686572 74
657320446F6B756D656E74206E6174C3BC726C696368206E6963687420617573 20E2809E57C3B6727465726E
E2809C20736F6E6465726E206C656469676C6963682061757320656 96E657220416E65696E616E6465727265
6968756E6720766F6E205A61686C656E636F6465732E
```

Textmatching – Indexsuche – invertierte Liste

Zeichenketten im Dokument

```
Für das IR-System besteht ein abgespeichertes Dokument nicht aus „Wörtern“ sondern
lediglich aus einer Aneinanderreihung von Zahlencodes:
```

Index

TERM	DOK-NR
abgespeichertes	1,145,56398
Aneinanderreihung	1,3,189
aus	1,2,3,4,5,6...
besteht	1,8,15,18,89,...
das	1,2,3,4,5,6,7...
ein	1,2,3,4,5,6,7...
...	
Zahlencodes	1

allgemeine Form

Term	Dok-Nr	(Treffer)	(Position)
aus	1,2,3,...	54378	27,456,3...
...	...	...	...
term	1,...n	n	n

Automatische Indexerstellung

Zerlegung eines Textes in Zeichenketten

Vorgehensweise:

Definition von Zeichenkette, bspw. „Alles zwischen zwei Leerzeichen“

Extraktion aller Zeichenketten durch Algorithmisierung der Definition

Alphabetisches Sortieren des Ergebnisses

Tabelle 5.1 Manuelle erzeugte Indexterme

-	(open	Abstract:
algorithmische	an	Anwendungsgrenzen
auf.	Automatische	automatischen
automatischen	basierten	Bei
Beispielen	Beitrag	benannt.
beschrieben,	bzw.	das
den	der	der
Der	Der	des
deutschen	deutschsprachiger	die
Die	Dokumente	ein
ein	eine	einer
Einsatzmöglichkeiten	Einsatzszenarien	Entwicklung
erläutert:	Flexibilität	frei
Funktionalität	für	für
für	für	Grundformerkennung,
hohe	im	Indexierung
Indexierung	Indexierung	Information
ist	Kompositumerkennung	Kompositumzerlegung,
Konfigurierbarkeit	lexikalische	Lingo
Lingo	lingo	lingo
lingo	linguistisch	linguistische
Mehrwortgruppenerkennung,	mögliche	Nutzen
OCR-Fehlerkorrektur.	offene	open
Retrieval	Retrievalverbesserung	source
source)	Sprache.	standen
stehende	System	System
Systemaufbau	Systems	Titel:
und	und	und
und	unterschiedliche	verfügbares
Verfügung	von	von
von	Vordergrund.	vorgestellt
werden	wird	wird
Wortrelationierung,	zeigt	zur
zur		

Intellektuelle Registererstellung

Festlegung von Sucheinstiegen auf Textstellen zum Thema

Titel:
Lingo - ein **open source** System für die **Automatische Indexierung** deutschsprachiger Dokumente

Abstract:
Lingo ist ein frei verfügbares System (open source) zur automatischen Indexierung der deutschen Sprache. Bei der Entwicklung von lingo standen hohe Konfigurierbarkeit und Flexibilität des Systems für unterschiedliche Einsatzmöglichkeiten im Vordergrund. Der Beitrag zeigt den Nutzen einer **linguistisch basierten automatischen Indexierung** für das **Information Retrieval** auf. Die für eine Retrievalverbesserung zur Verfügung stehende linguistische Funktionalität von lingo wird vorgestellt und an Beispielen erläutert: **Grundformerkennung**, Kompositumerkennung bzw. **Kompositumzerlegung**, Wortrelationierung, lexikalische und **algorithmische Mehrwortgruppenerkennung**, OCR-Fehlerkorrektur. Der offene Systemaufbau von lingo wird beschrieben, mögliche Einsatzszenarien und Anwendungsgrenzen werden benannt.

Abb. 5.3 Beispiele für ausgewählte Registerbegriffe

Registereingang

Automatische Indexierung
- Grundformerkennung
- Kompositumerkennung
- Kompositumzerlegung
- Mehrwortgruppenerkennung
- Wortrelationierung

Information Retrieval
Lingo
Mehrwortgruppenerkennung
- algorithmisch
- lexikalisch

Open source

Registerausgang

nn
nn
nn
nn
nn
nn
nn
nn
nn
nn
nn
nn

Vorgehensweise

Auswahl von wichtigen Wörtern im Text

Verwenden einer Funktion zur Markierung von Index- bzw. Registerbegriffen

Erzeugen des Index bzw Registers - alphabetisch sortiert, ggf. mit Hierarchie

Abb. 5.4 Beispiel für ein generiertes Register

ausgewähltes Vokabular

kontrolliertes Vokabular

Stichwortregister
Automatische Indexierung
- Grundformerkennung
- Kompositumerkennung
- Kompositumzerlegung
- Mehrwortgruppenerkennung
- Wortrelationierung
Information Retrieval
Lingo
Mehrwortgruppenerkennung
- algorithmisch
- lexikalisch
Open source
...

alle Zeichenketten

unkontrollierte Zeichenketten

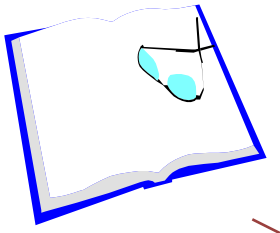
Volltextindexierung
-
(open
Abstract:
algorithmische
an
Anwendungsgrenzen
auf.
Automatische
automatischen
automatischen
basierten
Bei
Beispielen
Beitrag
benannt.
...

Abb. 5.5 Gegenüberstellung manueller und automatischer Indexerstellung

Automatisches Indexieren

Indexieren und Automatisches Indexieren

Dokumente



Volltexte

bibliografische
Referenzdaten

Zuteilung!

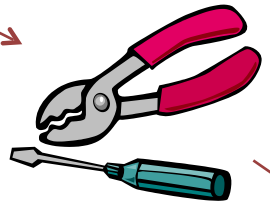
Abstract

Thesaurus

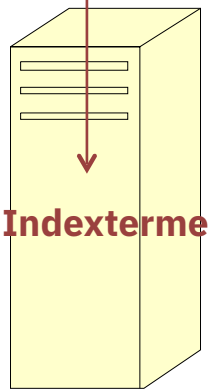
Klassifikation

Schlagwort

Inhaltserschließung
Indexierung



Repräsentationen des
Dokumentinhalts



Indexterme

Automatische
Indexierung

Verfahren zur Erzeugung von „guten“
Indextermen aus Dokumenten

Linguistik

Statistik

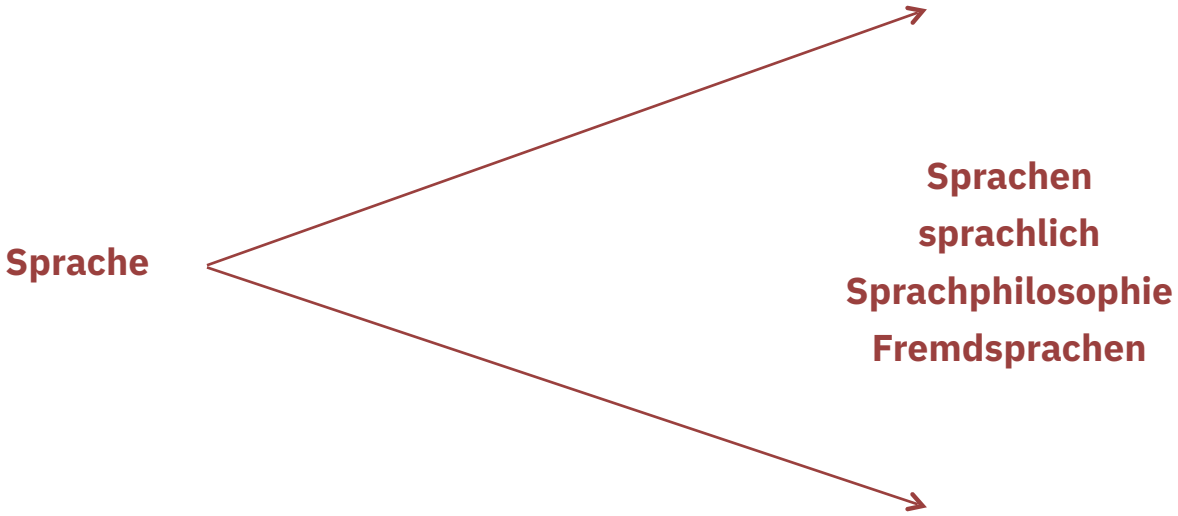
Extraktion!

Dokumentenspeicher
Suchsystem

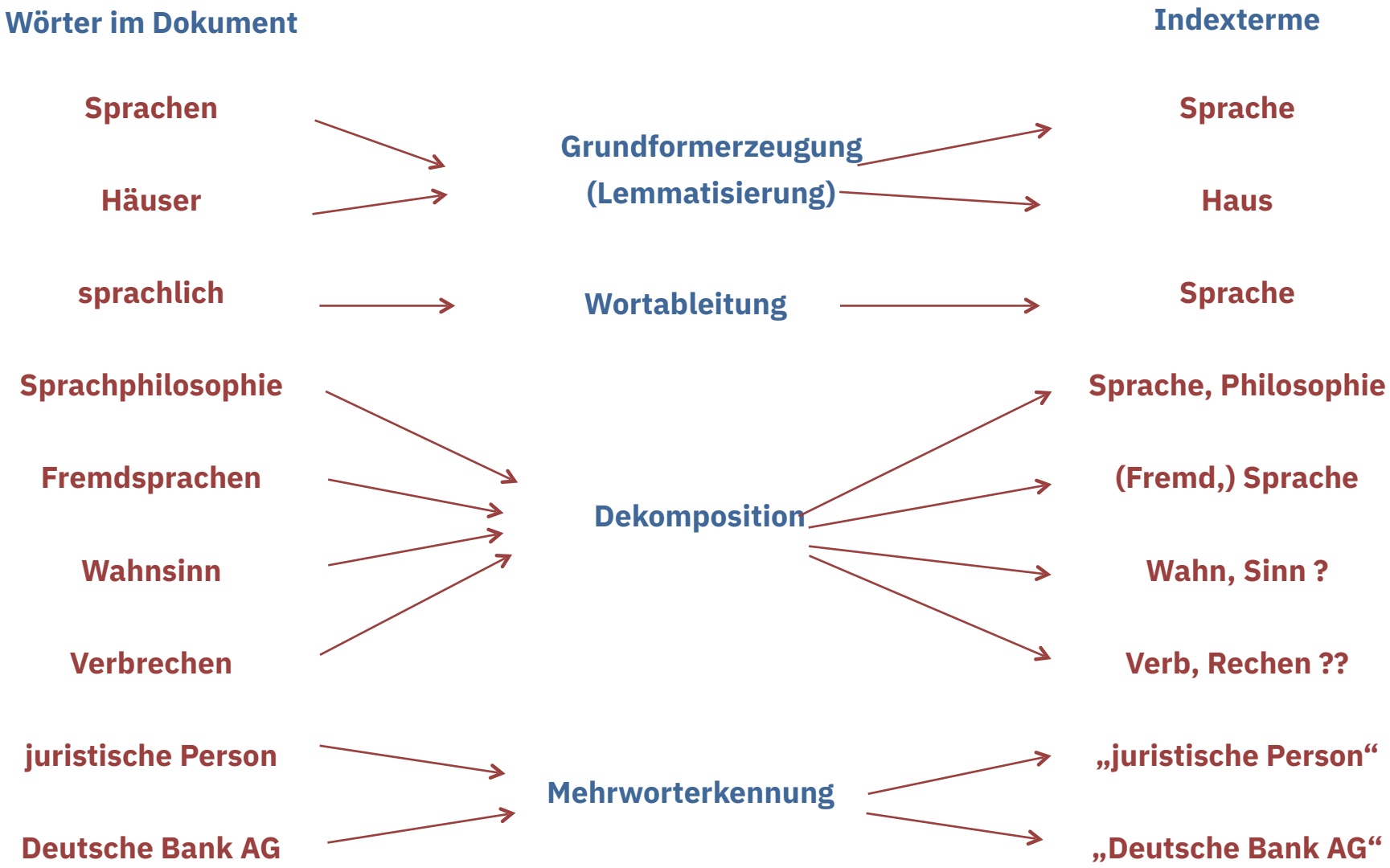
Hypothese

„Wörter in Dokumenten sind selten gute Indexterme, weil sie sprachlich zu verschieden sind.“

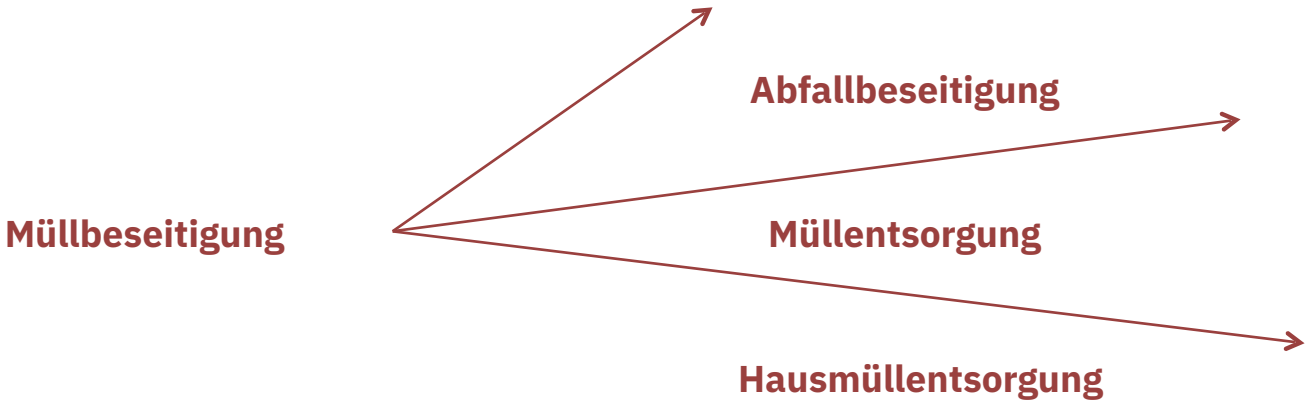
Verschiedenartigkeit von Dokument- und Suchsprache



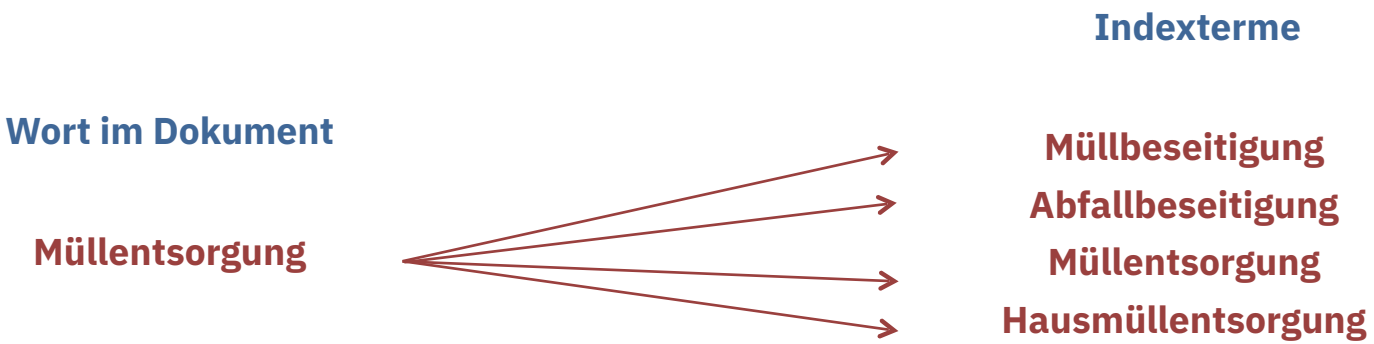
Lösung: Einsatz einer morphologischen Komponente



Problem: Suche im semantischen Umfeld



Lösung: Einbindung von Synonym- ggf. hierarchischen Relationen



Hypothese

„Nur Wörter mit bestimmten statistischen Merkmalen (Häufigkeitsmerkmalen) sind „gute“ Indexterme.“

Verfahren: Ermittlung von Worthäufigkeiten (z.B. TF, IDF)

Einsatzmöglichkeiten und Nutzen

Termgewichtung	**Automatische Klassifizierung**	**Dokumentclustering**
gewichtete Indexterme	**Notationen eines Klassifikationssystems**	**Mengen ähnlicher Dokumente**
Relevance Ranking	**Systematische Suche**	**Navigation in großen Dokumentkollektionen**

Automatische Indexierung vs. Intellektuelle Erschließung

Ziele	• Retrievalverbesserung durch geeignete Indexterme • Recallerhöhung auf Stichwortbasis	• einheitliche thematische Suche durch thematische, inhaltliche Zusammenführung • inhaltliche Repräsentation
Aufwand	• gering	• hoch
Kosten	• gering	• hoch
Ergebnisse	• Indexterme • vorhersagbar • beliebig reproduzierbar • durch wiederholte Indexierungen leicht zu verbessern	• Schlagwörter, Deskriptoren, Notationen • menschliche Fehlerquote • endgültig • inhaltlich verlässlich
Qualität	• abhängig von Qualität der Stichwörter im Quelltext	• hoch, bei konsistenter Erschließung
Nutzen	• für alle textbasierten Quelldokumente (auch erschlossene!) hoch • insb. für nicht erschlossene und nicht zu erschließende Dokumente hoch	• bezogen auf die erschlossenen Dokumente hoch • in Kollektionen mit niedriger Erschließungsquote gering

Was tun?

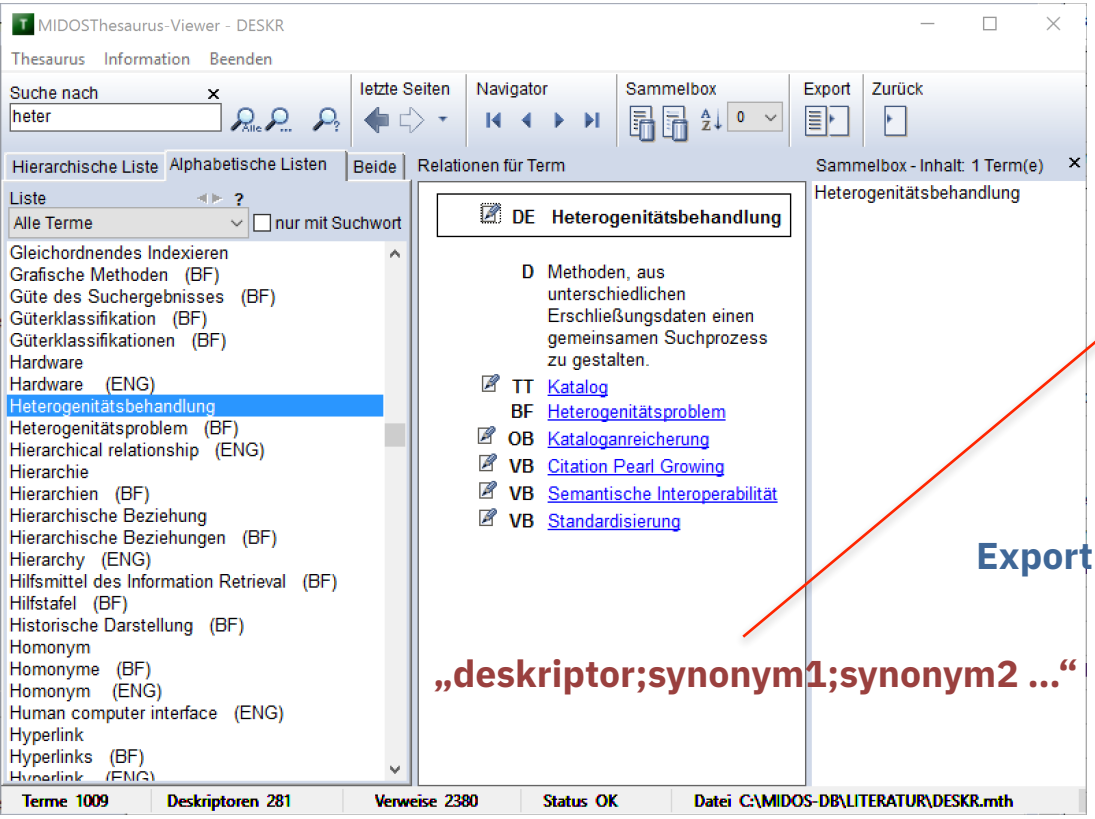
- Linguistische Verfahren verbessern die Suchbarkeit allein auf der Basis von Stichwörtern. Aus nicht geeigneten bzw. nicht vorhandenen Stichwörtern lässt sich kein brauchbares Indexat machen!
- Statistische Verfahren bewerten die vorhandene Stichwortbasis – s.o.
- Klassische Erschließungsziele wie Zusammenführung von Gleichem, vollständiger Nachweis, zuverlässige und einheitliche inhaltliche Suche sind nur durch Erschließung, nicht durch automatische Indexierung zu erreichen.
- Eine Entscheidung für automatische Indexierung muss das Wissen um deren Leistungsfähigkeit insb. deren Grenzen der Leistungsfähigkeit berücksichtigen.
- Einer Entscheidung für automatische Indexierung sollte daher eine Zielbestimmung hinsichtlich der gewünschten Retrievalmöglichkeiten jetzt und in 10 Jahren vorausgehen. Entscheidungen gegen eine Erschließung sind nur mit erheblichen Konsistenzverlusten umkehrbar.

Allgemein sind die Grenzen automatischer Erschließungsverfahren dort erreicht, wo die Intelligenz beginnt.

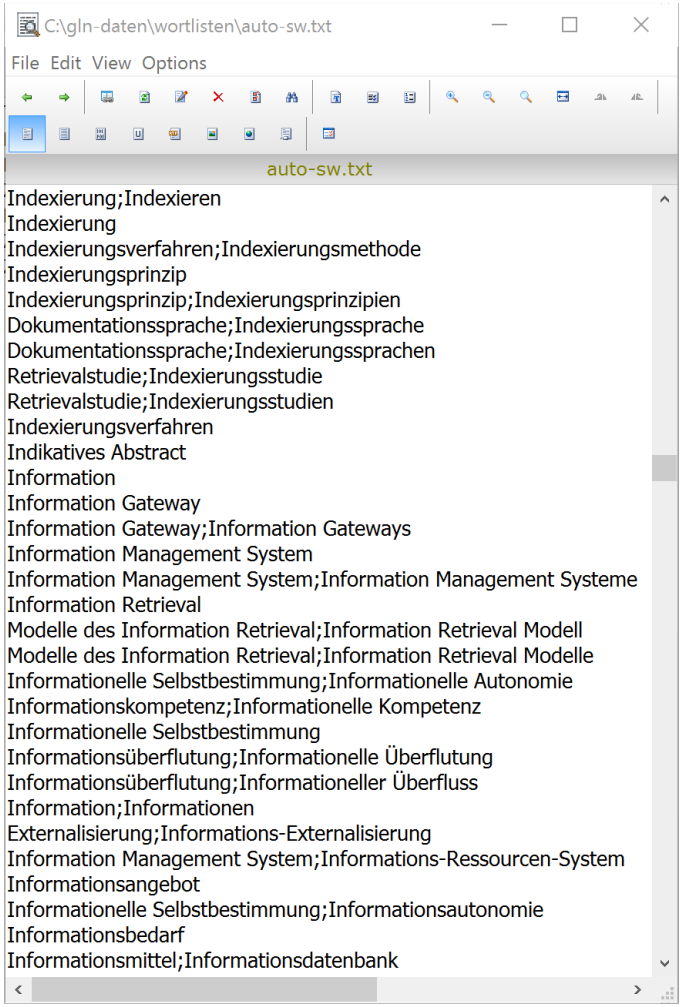
Automatische Schlagwortvergabe

Die „Automatische Schlagwortvergabe“ von Midos erzeugt auf der Basis von sog. Synonymlisten Deskriptoren zu Datensätzen. Dabei werden die Zeichenketten in den für das Verfahren ausgewählten Kategorien mit den Einträgen der Synonymliste verglichen und die erkannten bzw. erzeugten Deskriptoren in eine eigene Kategorie des Datensatzes geschrieben.

Thesaurus mit Allgemeindeskriptoren (deskr.mth)



Wortliste (auto-sw.txt



Export

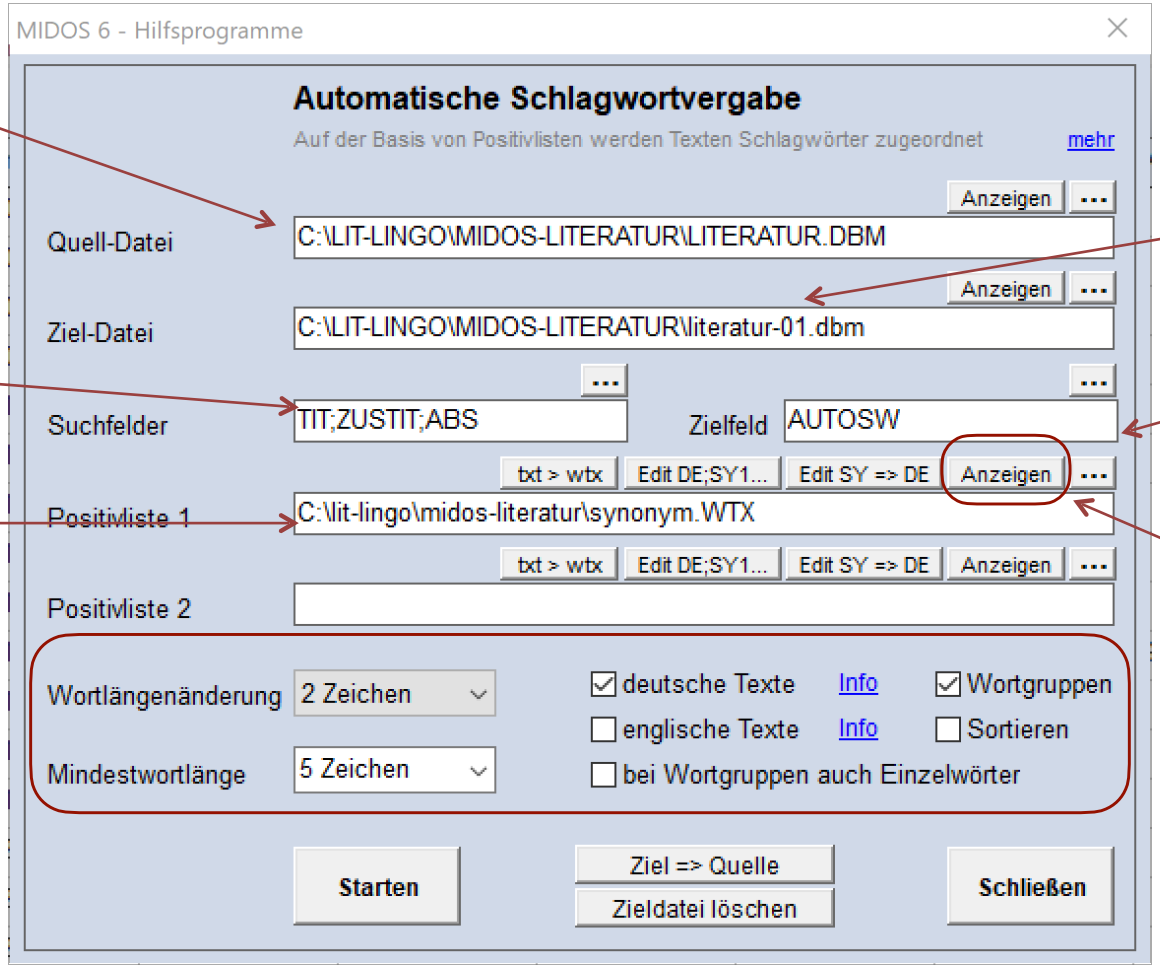
„deskriptor;synonym1;synonym2 ...“

Dialog „Automatische Schlagwortvergabe“

zu bearbeitende Datenbank

zu durchsuchende Felder

zu verwendende Positivliste



verschlagwortete Datenbank

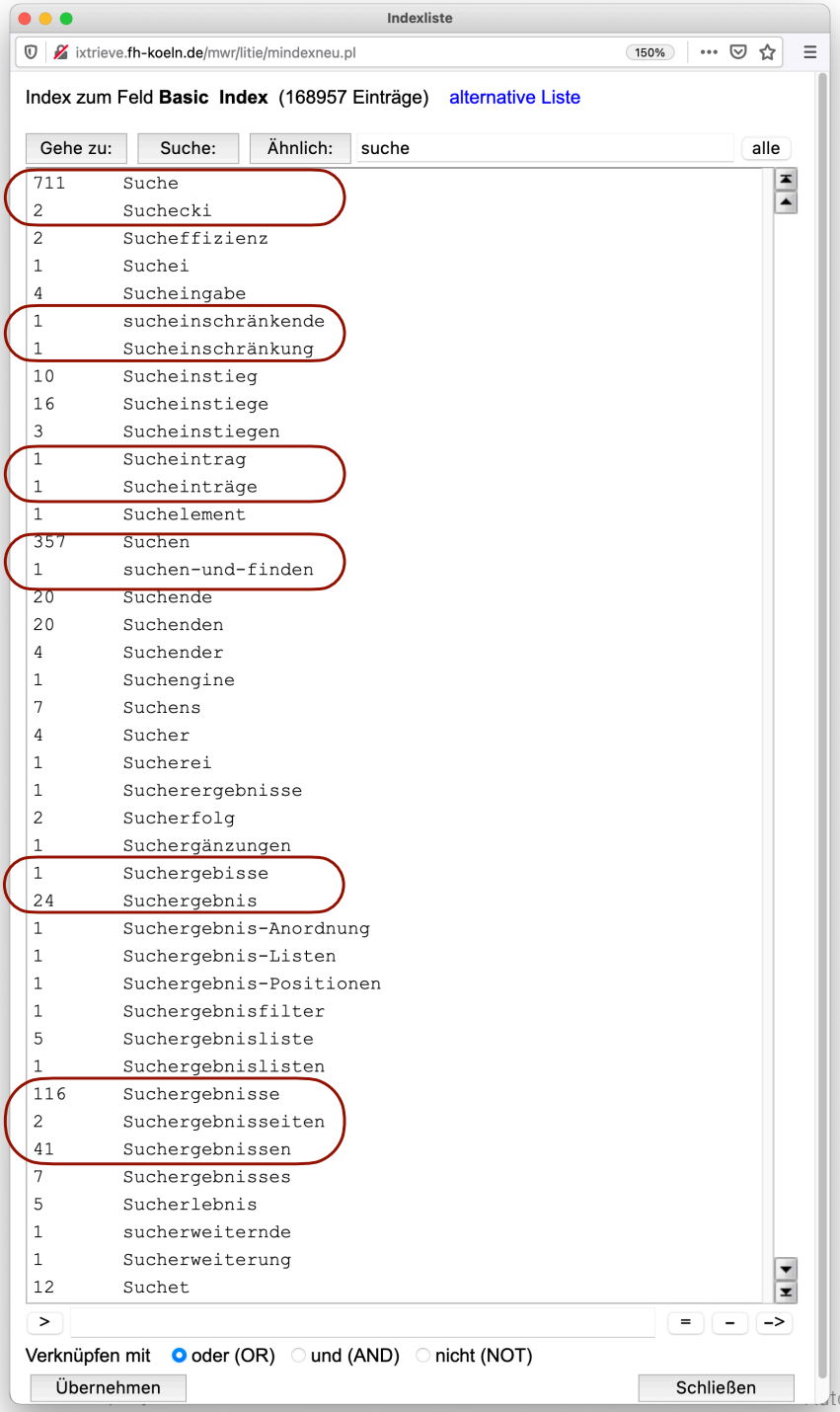
Feld für die Ergebnisse

Blick in die Positivliste

Einstellungen für zusätzliche sprachliche Verarbeitung

Testen!

Automatische Indexierung
Linguistisch basierte Verfahren



Stichwort - Wortform - Grundform

Zielvorstellung

sprachlich begründete Reduzierung von Wortformen in Dokumenten (Stichwörter) auf Grundformen

Wortform	**Suchen**
Grundform	**Suche**

Wortform	**Suchergebnisse**
Grundform	**Suchergebnis**
	Suche
	Ergebnis

Wortform	**sucheinschränkende**
Grundform	**sucheinschränkend**
	Sucheinschränkung
	Suche
	Einschränkung

Bestandteile der Sprache I

Phonem: kleinstes bedeutungsunterscheidendes Lautmerkmal

Maus - Haus **Mantel - Hantel**

Morphem: kleinste bedeutungstragende Einheiten einer Sprache

Be-haus-ung, haus-en, Haus-ierer

Wortform: Erscheinungsformen von Wörtern in der Sprache; Zuordnung zur lexikalischen Einheit, z.B.:

Haus, Häuser, Hauses, Häusern etc.
hausen, hausend

Wort: bedeutungstragende Einheiten der Sprache, bestehend aus einzelnen Morphemen oder einer Kombination mehrerer Morpheme;

Haus [Substantiv; Gebäude]
Maus [Substantiv; 1. Tier, 2. PC-Bediengerät]
hausen [Verb, umgspr. für wohnen]

Wörter können im Satz ausgetauscht werden und Satzglieder bilden

Der Mond ist aus grünem Käse.
Der Mond ist aus gelbem Käse.

Syntagma: mehrere Wörter können zu Wortgruppen, Wortfolgen oder Wortverbindungen verknüpft werden

der gelbe Mond
juristische Person
Regeln für den Schlagwortkatalog

Satz: Wortfolge mit mindestens einem Objekt (Subjekt) und einem Prädikat

Hans schläft.
Hans schläft gerne in meiner Vorlesung.

Morphologie - Wörter und ihre Bestandteile

einfache Wörter (Simplizia)

Uhr (Kernmorphem)

Uhr-en (Kernmorphem und Flexionsmorphem)

Ableitungen (Derivationen)

Ver-bind-ung-en
(Kernmorphem, Flexionsmorphem, Wortbildungsmorphem

Komposita (mind. 2 Kernmorpheme)

Uhr-en-ver-gleich-s-test

Wortbildung erfolgt zum Beispiel durch

hinzufügen von Präfixen zum Wortstamm

ver-walt-en
P K F

hinzufügen von Suffixen zum Wortstamm

Ver-walt-ung
P K F

Ver-un-rein-ig-ung
P P K S S

Grundformerzeugung - Lemmatisierung

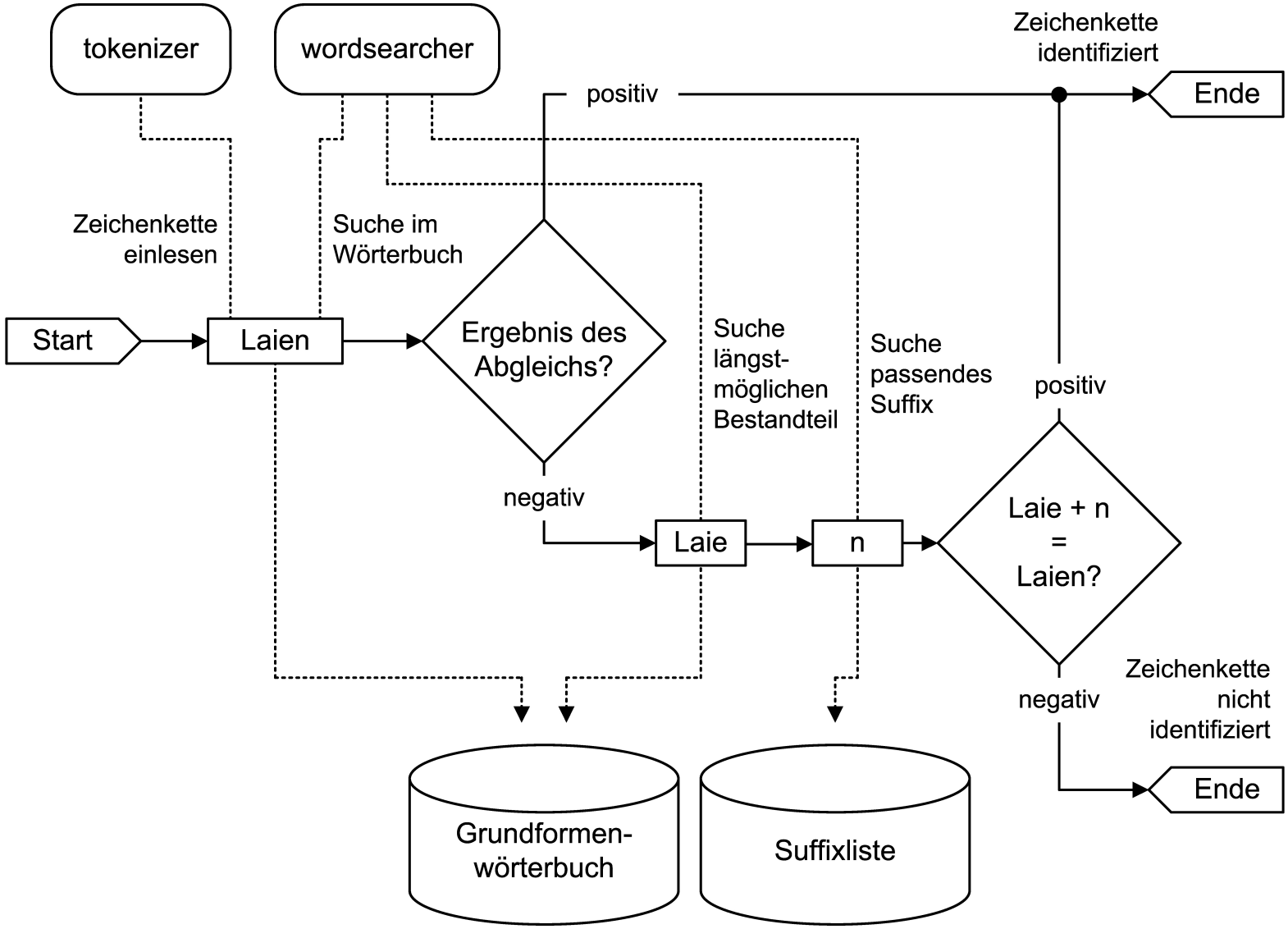


Abb. 5.14 Identifizierungsstrategie von *Lingo*

Tokenizer

Der tokenizer identifiziert Zeichenketten für den nachfolgenden Identifizierungsprozess (wordsearcher); Zeichen, die nicht Bestandteile von Wörtern sind, wie z.B. "&, $, =, @", wirken als Begrenzer von Zeichenketten.

```
:Der/WORD/0/0:
:tokenizer/WORD/1/4:
:identifiziert/WORD/2/14:
:Zeichenketten/WORD/3/28:
:für/WORD/4/42:
:den/WORD/5/47:
:nachfolgenden/WORD/6/51:
:Identifizierungsprozess/WORD/
:wordsearcher/WORD/9/90:
:(/OTHR/8/89:
:Zeichen/WORD/12/105:
:)/OTHR/10/102:
:;/PUNC/11/103:
:die/WORD/14/114:
:nicht/WORD/15/118:
:,/PUNC/13/112:
:Bestandteile/WORD/16/124:
:von/WORD/17/137:
:Wörtern/WORD/18/141:
…
```

```
:sind/WORD/19/150:
:wie/WORD/21/156:
:,/PUNC/20/154:
:z.B/ABRV/22/160:
:./PUNC/23/163:
:"/OTHR/24/165:
:&/OTHR/25/166:
:,/PUNC/26/167:
:$/OTHR/27/169:
:,/PUNC/28/170:
:=/OTHR/29/172:
:,/PUNC/30/173:
:wirken/WORD/34/179:
:@/OTHR/31/175:
:"/OTHR/32/176:
:als/WORD/35/186:
:,/PUNC/33/177:
:Begrenzer/WORD/36/190:
:von/WORD/37/200:
:./PUNC/39/217:
```

```
# tokenizer rules:
#   SPAC = \s+
#   NUMS = [+-]?(?:\d{4,}|\d{1,3}(?:\.\d{3,3})*)(?:\.|(?:,\d+)?%?)
#   URLS = (?:www\.|mailto:|(?:news|https?|ftps?)://|\S+?[._]\S+?@\S+?\.)\S+
#   ABRV = (?:(?:(?:CHAR)+\.)+)(?:CHAR)+
#   WORD = ALNUM(?:-*ALNUM)*
#   PUNC = [!,.:;?¡¿]
#   OTHR = [-"$#%&'()*+\-/<=>@\[\\\]^_{|}~¢£¤¥¦§¨©«¬®¯°±²³´¶·¸¹»¼½¾×÷]
#   HELP = \S*
```

Wordsearcher

Wörterbücher von *Lingo* sind reine Textdateien

```
lahmend=lahmend #a lahmen #v
lähmend=lähmend #a lähmen #v
lahmen=lahm #a lahmen #v
lähmen=lähmen #v
lahmgelegt=lahmgelegt #a lahmlegen #v
lahm=lahm #a lahmen #v
lahmlegend=lahmlegend #a lahmlegen #v
lähmung=lähmung #s.f
lahmzulegen=lahmzulegen #v
lahnung=lahnung #s.f
lahr=lahr #e.f
laib=laib #s.m
laichen=laichen #v
laie=laie #s.m
laienhaft=laienhaft #a
laientum=laientum #s.n
laizistisch=laizistisch #a
lakai=lakai #s.m
lake=lake #s.f
laken=laken #s.n
lakonisch=lakonisch #a
lakritze=lakritze #s.f
laktase=laktase #s.f lact
laktation=laktation #s.f
```

Wortform

Grundform(en)

Wortklasse

Genus

```
# Systemwörterbuch lingo-dic.txt
# enthält deutsche Grundformen mit Wortklassen
#
# #s = Substantiv
# #a = Adjektiv
# #v = Verbe
# #e = Eigenname
# #t = Takeitasis
# #w = Wortform
```

Identifizierungsstrategie

Token

Laien

Abgleich mit dem Wörterbuch

```
lahr=lahr #e.f
laib=laib #s.m
laichen=laichen #v
laie=laie #s.m
laienhaft=laienhaft #a
laientum=laientum #s.n
laizistisch=laizistisch #a
lakai=lakai #s.m
lake=lake #s.f
laken=laken #s.n
lakonisch=lakonisch #a
lakritze=lakritze #s.f
laktase=laktase #s.f lactase #s.f
laktation=laktation #s.f
```

längster übereinstimmender Eintrag

laie

Überprüfung:
falls „laien“ eine zulässige Wortform der Grundform „laie“ ist, muss „n“ erlaubtes Suffix für die Wortklasse „s“ sein

Bestätigung

Ergebnis

<Laien = [(laie/s/m)]>

```
suffix:
  # Suffixliste, Stand: 30-06-2005
  # Suffixklasse: s = Substantiv, a = Adjektiv, v = Verb, e = Eigenwort, f = Fugung
  # Suffixe je Klasse: "<suffix>['/'<ersetzung>][ <suffix>['/'<ersetzung>]]"
  - [s, 'e en er ern es n s se sen ses']
  - [a, 'este ste ster sten stes ester estes esten e em en er ere eren erer eres es erem']
  - [v, 'e/en en/en est/en et/en st/en t/en te/en ten/en eten/en ete/en etest/en s']
  - [e, 's']
  - [f, 's n e en es er ch/che ch/chen']
```

Kompositumerkennung

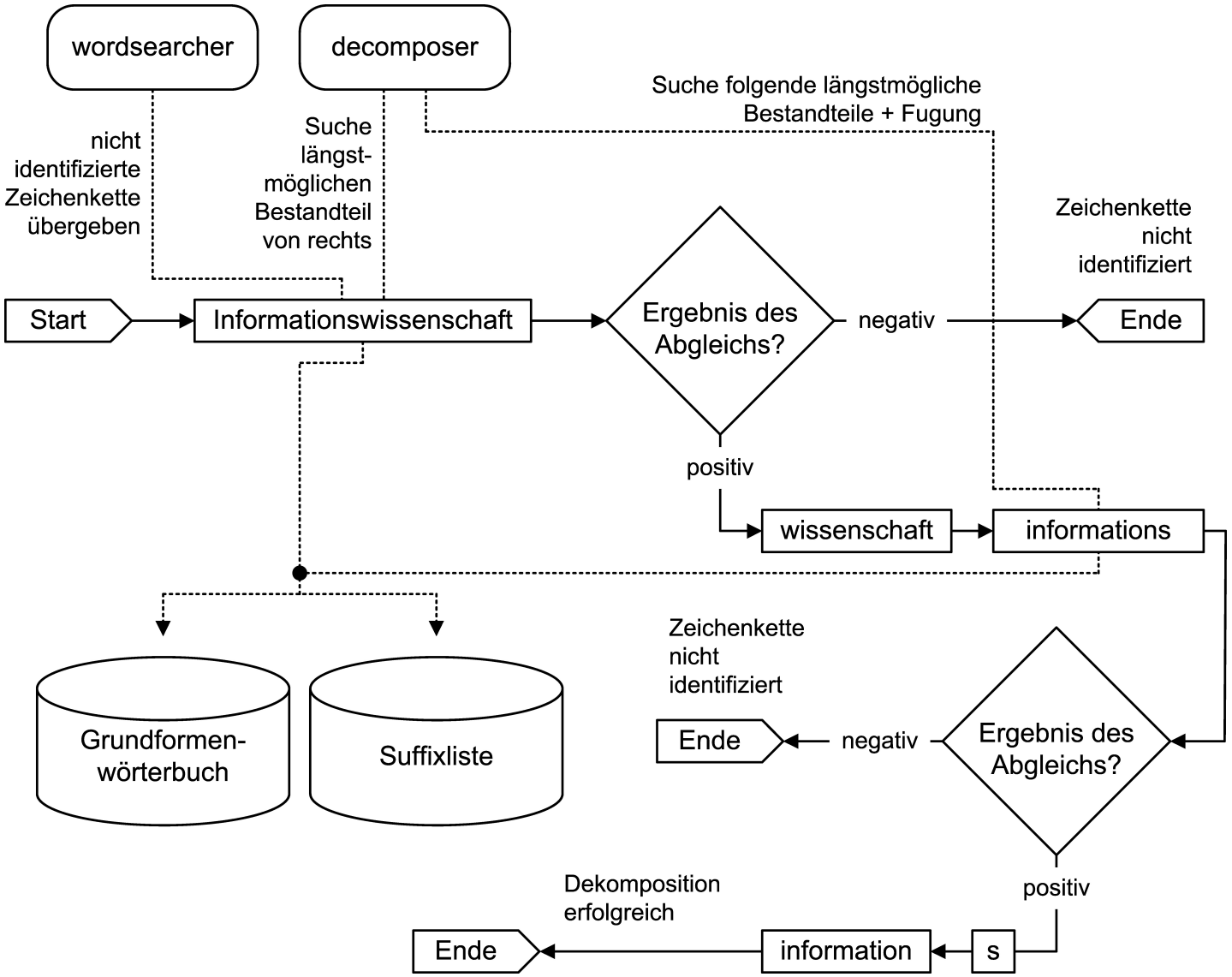


Abb. 5.16 Identifizierung von Komposita mit *Lingo*

Token

Informationswirtschaft

Abgleich mit dem Wörterbuch

```
informant=informant #s.f
informatiker=informatiker #s.m
informatik=informatik #s.f
information=information #s.f
informationsverarbeitend=informationsverarbeitend #a
informativ=informativ #a
informatorisch=informatorisch #a
```

Informationswirtschaft

nicht im Wörterbuch

?

Suche nach längst-möglichem Bestandteil von rechts

```
wirtschaftlichkeit=wirtschaftlichkeit #s.f
wirtschaftlich=wirtschaftlich #a
wirtschaft=wirtschaft #s.f
wirt=wirt #s.m
wir=wir #w
```

Informations

positiv - Suche nach vorderem Teil

Information-s

Bestätigung

Ergebnis

<informationswirtschaft|COM = [(informationswirtschaft/k), (information/s+/f), (wirtschaft/s+/f)]>

Überprüfung auf zulässige Endung im Kompositum (Fugung)

```
suffix:
  # Suffixliste, Stand: 30-06-2005
  # Suffixklasse: s = Substantiv, a = Adjektiv, v = Verb, e = Eigenwort, f = Fugung
  # Suffixe je Klasse: "<suffix>['/'<ersetzung>][ <suffix>['/'<ersetzung>]]"
  - [s, 'e en er ern es n s se sen ses']
  - [a, 'este ste ster sten stes ester estes esten e em en er ere eren erer eres es erem']
  - [v, 'e/en en/en est/en et/en st/en t/en te/en ten/en eten/en ete/en etest/en s']
  - [e, 's']
  - [f, 's n e en es er ch/che ch/chen']
```

Konfiguration (lit-lingo-lem.cfg)

Analyseteil

```
#
# Lingo-Konfiguration
# Version "lem" - nur tokenizer, word_searcher, decomposer, vectorfilter
# 2021-02-16 / le
---
meeting:

  attendees:

    ##################################################
    # Text bereitstellen
    #

    # Angegebene Datei zeilenweise einlesen und verarbeiten
    - text_reader:     { files: $(files), progress: true }


    ##################################################
    # Inhalte verarbeiten
    #

    # Zeile in einzelnen Sinnbestandteile (Token) zerlegen
    - tokenizer:       { }

    # Verbleibende Token im Wörterbuch suchen
    - word_searcher:   {source: 'usr-dic sys-dic'  mode: first}

    # Nicht erkannte Wörter auf Kompositum testen
    - decomposer:      {source: 'usr-dic sys-dic'  out: res}
```

verwendete Wörterbücher

Verarbeitung nach erster Identifizierung beenden

Ausgabe des Ergebnisses als „res“

Konfiguration (lit-lingo-lem.cfg)

Ausgabeteil

attendee für die Ausgabe der „log“-Datei

Filtern der Ergebnisse in „res“ und Sortierung

Ausgabe der Ergebnisse in Datei

```
##################################################
# Ergebnisse ausgeben
#
```

Einlesen des Ergebnisses aus dem Analyseteil

```
# Erstelle Datei mit Endung .log für Datenstrom
- debug_filter:    { in: res, prompt: 'lex:) ' }
- text_writer:     { ext: log, sep: "\n" }

# Erstelle Datei mit Endung .non für nicht erkannte Wörter
- vector_filter:   { in: res, lexicals: '\?', sort: term_abs }
- text_writer:     { ext: non, sep: "\n" }
```

Filtereinstellungen: Wortklassen

```
# Erstelle Datei mit Endung .vec für erkannte Indexterme
- vector_filter:   { in: res, lexicals: '^[ksave]$' }
- text_writer:     { ext: vec, sep: "\n" }

# Erstelle Datei mit Endung .ven für erkannte Indexterme mit absoluter Häufigkeit
- vector_filter:   { in: res, lexicals: '^[ksavem]$', sort: term_abs }
- text_writer:     { ext: ven, sep: "\n" }
```

Dateiendung

jeder Eintrag auf neuer Zeile

Sortierung der ausgegebenen Ergebnisse: absolute Termhäufigkeit

Mehrworterkennung

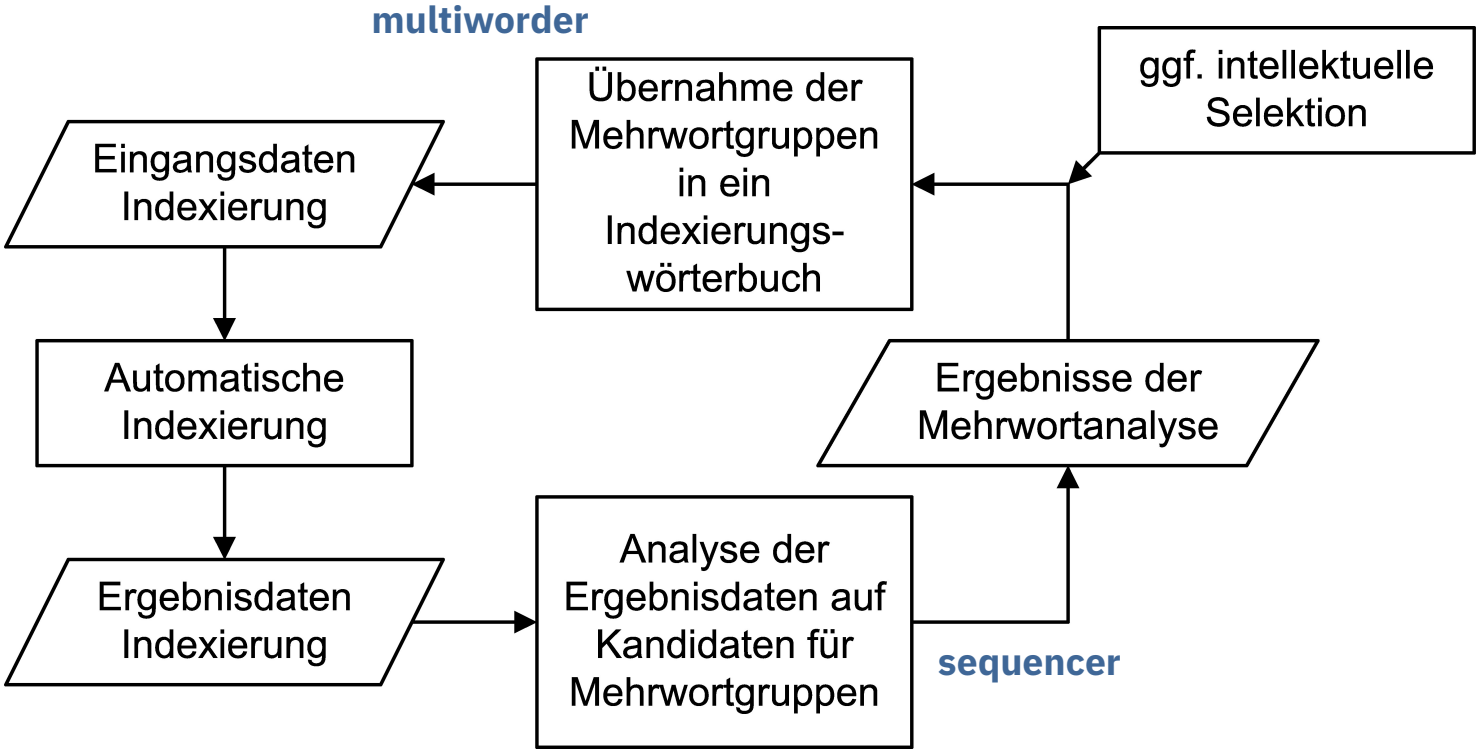


Abb. 5.17 Ablaufschema der Mehrworterkennung

lit-lingo-sem.cfg

```
---
meeting:

  attendees:

    ##############################################
    # Text bereitstellen
    #

    # Angegebene Datei zeilenweise einlesen und verarbeiten
    - text_reader:     { files: $(files), progress: true }


    ##############################################
    # Inhalte verarbeiten
    #

    # Zeile in einzelnen Sinnbestandteile (Token) zerlegen
    - tokenizer:       { }

    # Verbleibende Token im Wörterbuch suchen
    - word_searcher:   { source: 'usr-dic sys-dic', mode: first }

    # Nicht erkannte Wörter auf Kompositum testen
    - decomposer:      { source: 'usr-dic sys-dic' }

    # Mehrwortgruppen im Strom erkennen
    - multi_worder:    { source: sys-mul }

    # Wortsequenzen anhand von Regeln identifizieren
    - sequencer:       { stopper: 'PUNC,OTHR', out: res }
```

wörterbuchbasierte Mehrworterkennung

regelbasierte Mehrworterkennung

Sequencer - algorithmische Identifizierung von Mehrwortgruppen

Konfigurationsdatei „de.lang“

```
sequencer:
      sequences: [ [AS, '2, 1'], [AK, '2, 1'], [AAK, '3, 1 2'], [AAS, '3, 1 2'] ]
```

Wortklassenmuster **Ausgabe**

```
automatisch akquisition -- akquisition, automatisch
automatisch bestimmung -- bestimmung, automatisch
automatisch dokumenterschließung -- dokumenterschließung, automatisch          AK
automatisch dokumentklassifikation -- dokumentklassifikation, automatisch
automatisch erschließung -- erschließung, automatisch                          AS
automatisch gruppierung -- gruppierung, automatisch
automatisch indexieren -- indexieren, automatisch
automatisch indexierung -- indexierung, automatisch
automatisch indexierungssystem -- indexierungssystem, automatisch
automatisch inhaltlich erschließung -- erschließung, automatisch inhaltlich    AAS
automatisch inhaltserschließung -- inhaltserschließung, automatisch
automatisch klassifikation -- klassifikation, automatisch
automatisch maschine -- maschine, automatisch
automatisch methode -- methode, automatisch
automatisch recherche -- recherche, automatisch
automatisch selektion -- selektion, automatisch
automatisch semantisch klassifikation -- klassifikation, automatisch semantisch
automatisch thematisch textklassifikation -- textklassifikation, automatisch thematisch   AAK
automatisch verfahren -- verfahren, automatisch
automatisch vollindexierung -- vollindexierung, automatisch
automatisch wortformenreduktion -- wortformenreduktion, automatisch
automatisch übersetzungssystem -- übersetzungssystem, automatisch
```

Multiworder - wörterbuchbasierte Identifizierung von Mehrwortgruppen

lingo-mul.txt

```
automatische bildverarbeitung
automatische datenverarbeitung
automatische differentiation
automatische dokumentation
automatische fahrtzielführung
automatische fahrzeugführung
automatische flugsteuerung
automatische folge
automatische handlungsplanung
automatische inhaltsanalyse
automatische justierung
automatische lastabhängige bremskraftregelung
```

Lingo-Wörterbücher sind zwar reine Textdateien, werden aber vor der Indexierung zu **Indexdateien** (*sdbm*) konvertiert, um die Verarbeitung zu beschleunigen. Dabei kann eine Vorverarbeitung stattfinden, z.B. eine Lemmatisierung der Einträge, um auch Varianten erkennen zu können.

```
automatisch lastabhängig bremskraftregelung #m
```

erkennt:

```
automatische lastabhängige bremskraftregelung
automatische lastabhängige bremskraftregelungen
automatischen lastabhängigen bremskraftregelung
automatischen lastabhängigen bremskraftregelungen
```

de.lang

```
---
language:
  name: 'Deutsch'

  dictionary:
    databases:
      # System dictionaries
      sys-dic: { name: de/lingo-dic.txt, txt-format: WordClass, separator: '=' }
      sys-abk: { name: de/lingo-abk.txt, txt-format: WordClass, separator: '=' }
      sys-syn: { name: de/lingo-syn.txt, txt-format: KeyValue,  separator: '=', def-wc: y }
      sys-mul: { name: de/lingo-mul.txt, txt-format: SingleWord, use-lex: sys-dic, def-wc: m }
```

interner Name **Quelldatei** **Format der Einträge** **Vorverarbeitung: Lemmatisierung** **Wortklasse**

Synonymer - Zuteilung von Synonymen als Indexterme

gnd-syn-lingo.txt

```
VIR*Visual Information Retrieval
Viröse Vergilbung*Vergilbungskrankheit
Viröse Zweignekrose*Viröse Zweignekrose
Viraginität*Virilismus
Viral Marketing*Viral Marketing
Virale hämorrhagische Septikämie*Hämorrhagische Virusseptikämie
Virale Infektion*Virusinfektion
Virales Insektizid*Virales Insektizid
Virales Marketing*Viral Marketing
virales onkogen*virales onkogen
Viramid*Ribavirin
Virashaivas*Wiraschaiwa
Virasoro-Algebra*Virasoro-Algebra
Virazol*Ribavirin
Virelai*Virelai
Viren / Hülle*Virushülle
Viren / Inaktivierung*Virusinaktivierung
Viren*Viren
Virenhemmendes Mittel*Virostatikum
Virenneutralisation*Virusneutralisation
virenscanner*virenscanner
Virenschutzpaket DOS*VSPdos
Virenschutzpaket Windows 3.0*VSPwin
Virenschutzsoftware*Virenschutzsoftware
```

„Viraginität“ im Text erzeugt zusätzlich „Virilismus“

Synonyme können Mehrwortgruppen sein, daher zunächst Mehrworterkennung, danach Synonymgenerierung

„Hilfseintrag“ zur Identifizierung von Deskriptoren

Eigennamen

GND-Besonderheiten: Kombinationen

mehrwortige Eigennamen

Synonym*Vorzugsbenennung **Wortklasse**

```
# User dictionaries de.lang
usr-dic: { name: de/user-dic.txt,  txt-format: WordClass, separator: '=' }
usr-syn: { name: de/gnd-syn-lingo.txt, txt-format: KeyValue, separator: '*', def-wc: y }
```

Alternatives Wörterbuchformat: MultiValue

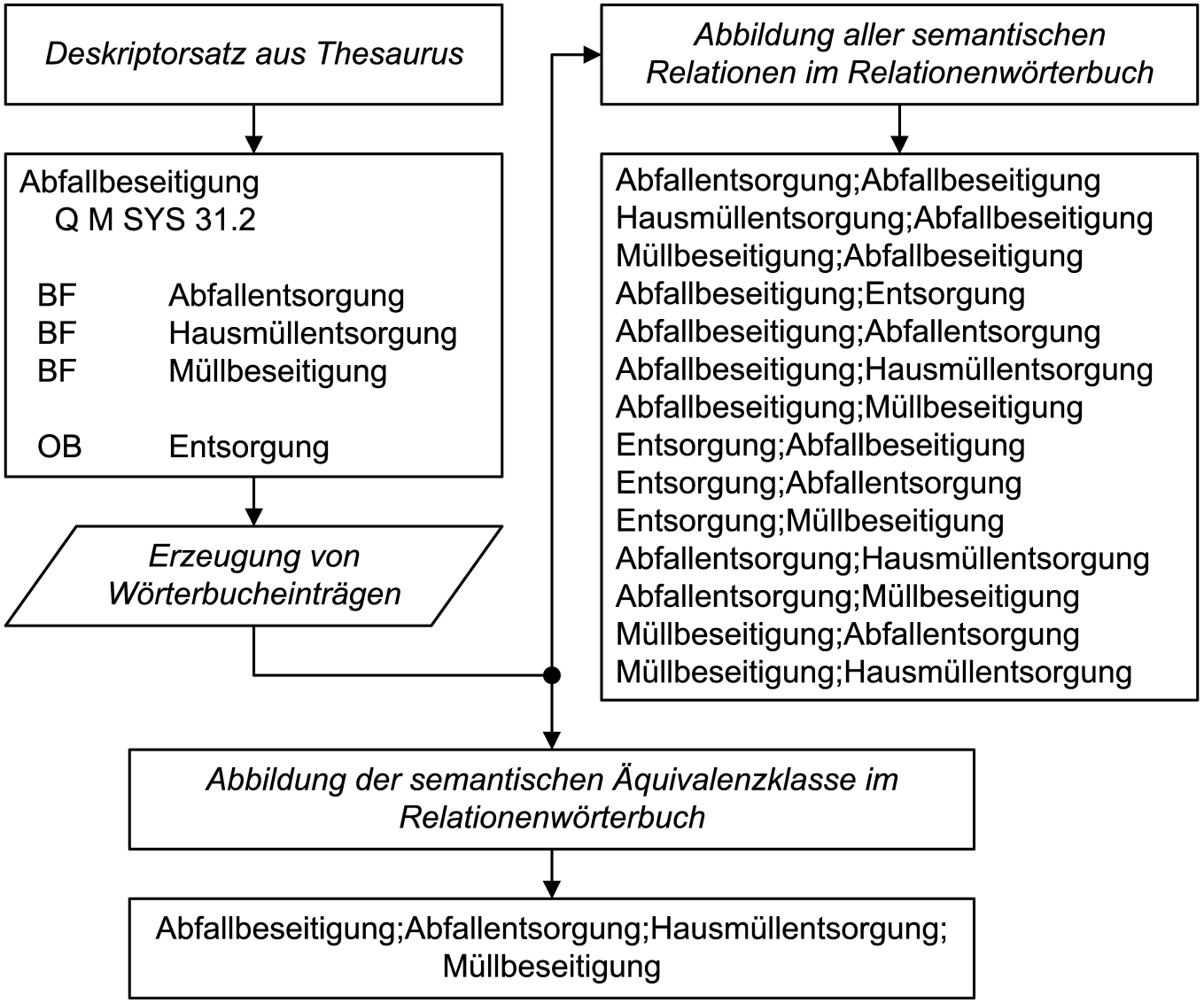


Abb. 5.19 Aufbau der Relationenwörterbücher für Synonyme

Indexieren einer Dokumentkollektion

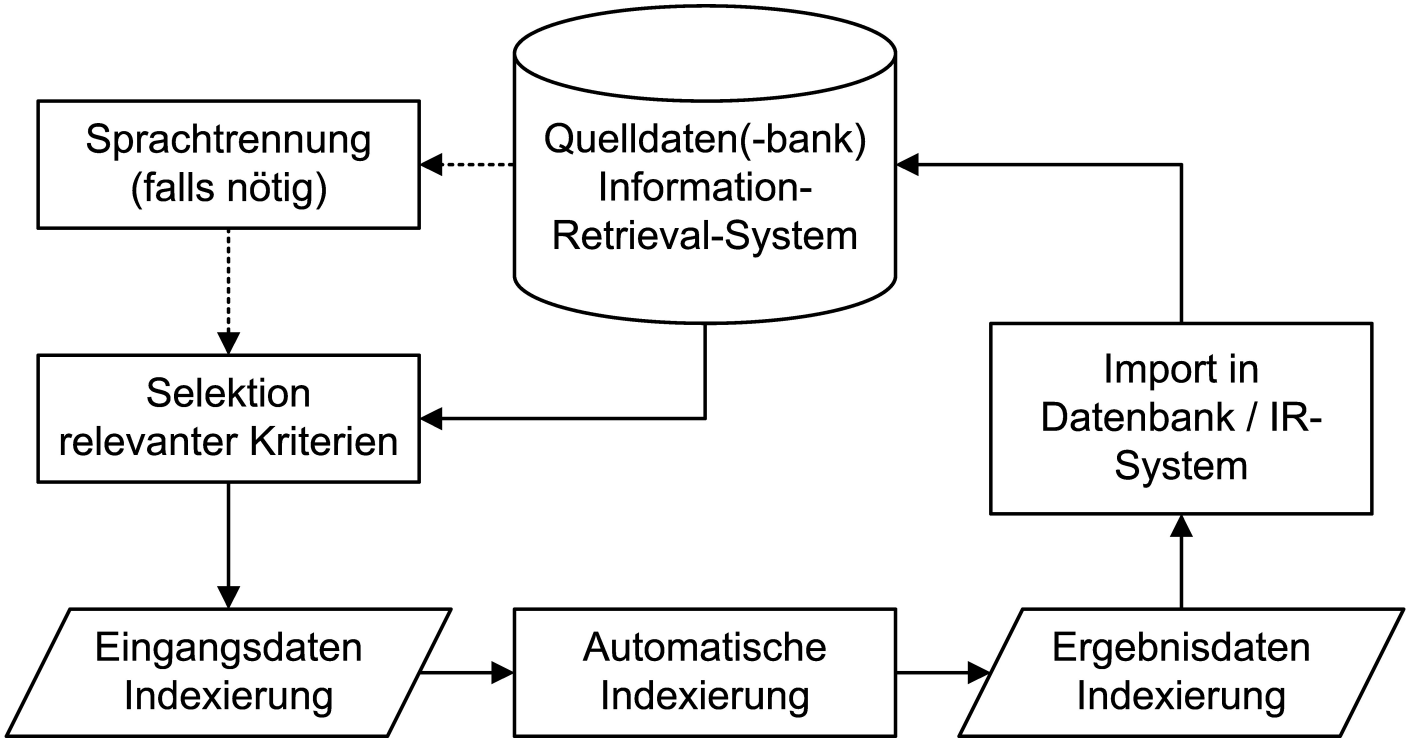
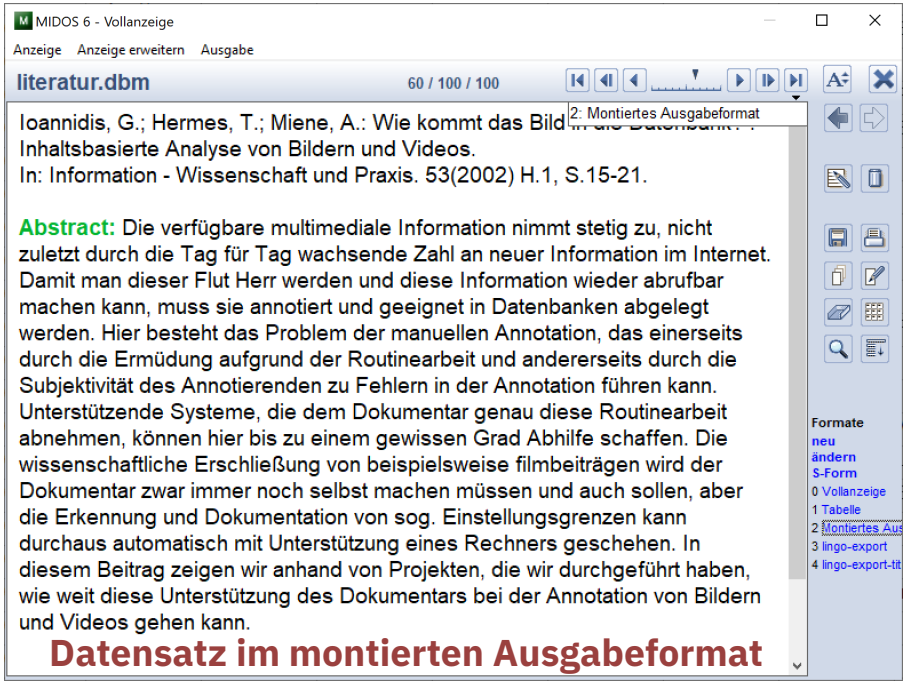
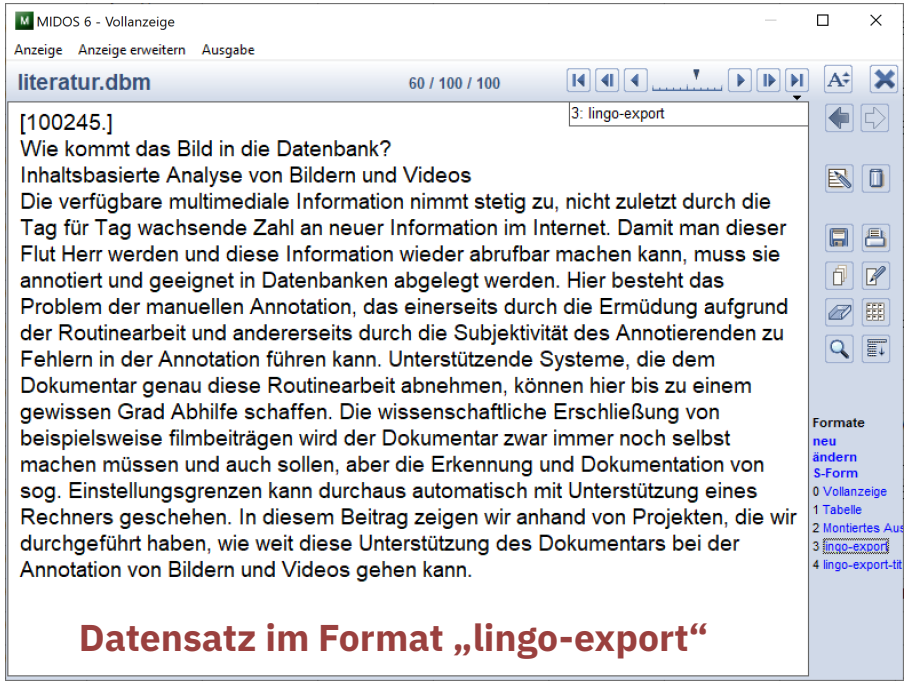


Abb. 5.20 Ablaufschema eines Indexierungslaufs

Export der Daten



Datensatz im montierten Ausgabeformat



Datensatz im Format „lingo-export“

Export-Dialog in Midos

lit-lir.cfg

```
#
# Lingo-Konfiguration für Dateien im LIR-Format
#
# Gebräuchliche Patterns sind
#
#   "^\021(\d+\-\d+)\022"
#   "^\[(\d+)\.\]"
#
---
meeting:

  attendees:

    ##########################################
    # Text bereitstellen
    #

    # Angegebene Datei zeilenweise einlesen und verarbeiten
    - text_reader:     { files: $(files), records: true, progress: true }
```

„**LIR-Modus**“: Indexieren einer Datei, die aus einzelnen Datensätzen besteht

Erkennungsmuster für die Identnummer eines Datensatzes

„records: true“ - datensatzweise Indexierung

exportierte Datensätze

```
[100019.]
Halbautomatische Volltextanalyse, Datenbankaufbau und Document Retrieval
[100022.]
Nutzung von Klassifikationssystemen zur verbesserten Beschreibung, Organisation und Suche von Internetressourcen
```

Indexierungsergebnis

```
100019*datenbankaufbau|halbautomatisch|retrieval|volltextanalyse
100022*beschreibung|beschreibung, verbessert|internetressource|klassifikationssystem|nutzung|organisation|suche|
suchen|verbessert
```

Identnummer

```
100019*datenbankaufbau|halbautomatisch|retrieval|volltextanalyse
100022*beschreibung|beschreibung, verbessert internetressource|klassifikationssystem|nutzung|organisation|suche|
suchen|verbessert
```

Indexierungsergebnis „export.vec“

Indexterm

Trenner: |

Trenner für Felder ID, Lingo

Job-Datei für Import in Midos

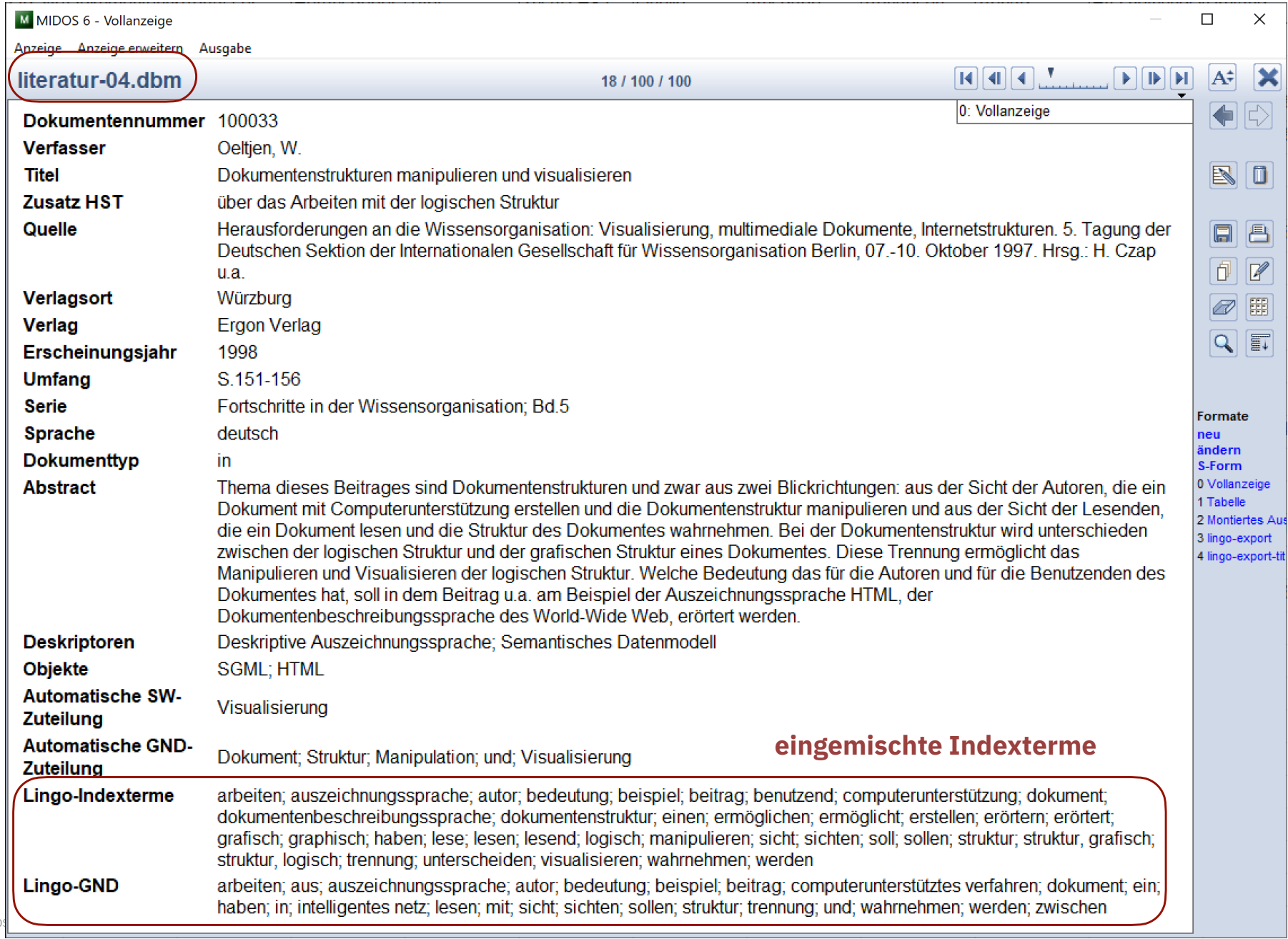
```
Job-Kommandofolge für MWUPDATE (Achtung !!! Datei nur mit MWUPDATE bearbeiten)
UTF8TOANSI
C:\lit-lingo\lingo-work\txt\export.vec
C:\lit-lingo\lingo-work\txt\lingo.csv

UTF8TOANSI
C:\lit-lingo\lingo-work\txt\export.syn
C:\lit-lingo\lingo-work\txt\lingo-gnd.csv

READAUSTAUSCH
C:\lit-lingo\lingo-work\txt\lingo.csv
C:\lit-lingo\lingo-work\txt\lingo.dbm
FELD:DOKNR;LINGO TZ:* SZ:^ nohead noEZ
READAUSTAUSCH
C:\lit-lingo\lingo-work\txt\lingo-gnd.csv
C:\lit-lingo\lingo-work\txt\lingo-gnd.dbm
FELD:DOKNR;LINGOGND TZ:* SZ:^ nohead noEZ
MISCHDATNEU
C:\lit-lingo\midos-literatur\literatur-02.dbm
C:\lit-lingo\lingo-work\txt\lingo.dbm
C:\lit-lingo\midos-literatur\literatur-03.dbm KEYFELD:DOKNR
MISCHDATNEU
C:\lit-lingo\midos-literatur\literatur-03.dbm
C:\lit-lingo\lingo-work\txt\lingo-gnd.dbm
C:\lit-lingo\midos-literatur\literatur-04.dbm KEYFELD:DOKNR
```

1. Umbenennen in „csv“ und konvertieren in UTF-8

2. Konvertieren von „csv“ in Midos-Datenbankformat „dbm“

2. Einmischen der Indexierungsergebnisse im „dbm“-Format in die Datensätze der Datenbank und umbenennen der Datenbank

Ergebnis des Midos-Jobs



Stemming-Verfahren - algorithmische Lösungen für die sprachliche Vereinheitlichung

Voraussetzung

Die Angleichung der Wörter an die Grammatik erfolgt regelhaft mit einem überschaubaren Vorrat an Regeln und einer nicht zu großen Zahl an Ausnahmen. Typischer Einsatzbereich ist die Verarbeitung der englischen Sprache.

famil-ies ⟶ **famil-y** **satisf-ied** ⟶ **satisfy**

index-ing ⟶ **index** **retriev-ed** ⟶ **retrieve**

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

Abb. 5.26 Kuhlen-Algorithmus

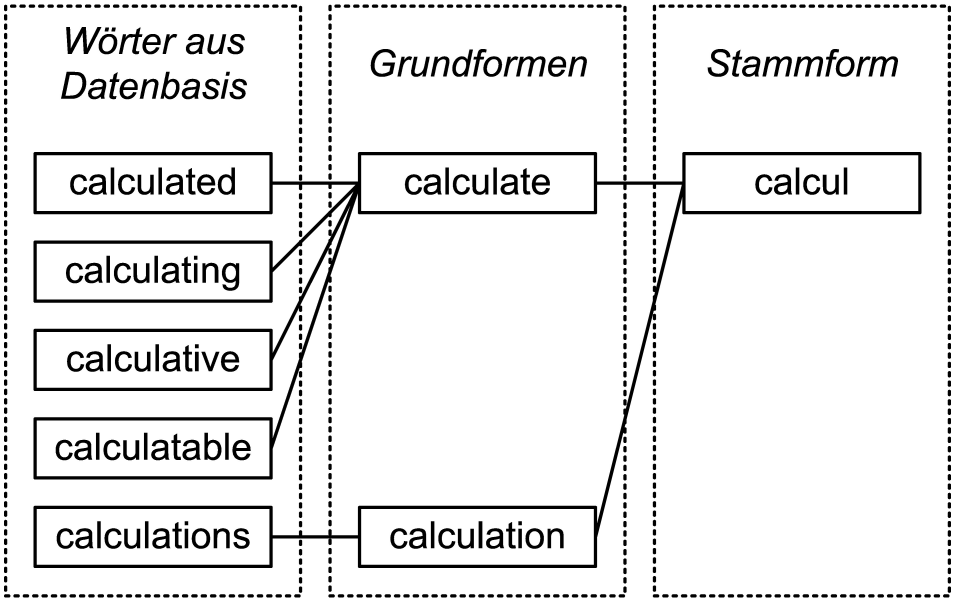


Abb. 5.27 Stammformreduktion für das Englische

Grundformen sind vollständige Wörter, **Stammformen** nicht

Grundformen sind taugliche Indexterme, **Stammformen** nicht – falls Stämme Indexterm werden, muss auch der Suchterm „gestemmt“ werden.

Overstemming

winner	→	win-ner
winning	→	win-ning
wine	→	win-e

Understemming

divide	→	divid-e
dividing	→	divid-ing
division	→	divis-ion

Porter: „An algorithm for suffix stripping“

These may all be represented by the single form

[C]VCVC ... [V] **(V=Vokal und C=Konsonant)**

where the square brackets denote arbitrary presence of their contents.
Using (VC){m} to denote VC repeated m times, this may again be written as

[C](VC){m}[V]. **m = Zahl der Wiederholungen**

m will be called the \measure\ of any word or word part when represented in this form. The case m = 0 covers the null word. Here are some examples:

Beispiele

m=0 TR, EE, TREE, Y, BY.
m=1 TROUBLE, OATS, TREES, IVY.
m=2 TROUBLES, PRIVATE, OATEN, ORRERY.

The \rules\ for removing a suffix will be given in the form

(condition) S1 -> S2 **Regeln für Suffixentfernung**

This means that if a word ends with the suffix S1, and the stem before S1 satisfies the given condition, S1 is replaced by S2. The condition is usually given in terms of m, e.g.

(m > 1) EMENT ->

zusätzliche Bedingungen

The `condition' part may also contain the following:

*S - the stem ends with S (and similarly for the other letters).
v - the stem contains a vowel.
*d - the stem ends with a double consonant (e.g. -TT, -SS).
*o - the stem ends cvc, where the second c is not W, X or Y (e.g. -WIL, -HOP).

Abfolge im Algorithmus

```
Step 1a
    SSES -> SS          caresses  ->  caress
    S    ->             cats      ->  cat

Step 1b
    (m>0) EED -> EE     agreed    ->  agree
    (*v*) ED  ->        plastered ->  plaster
    (*v*) ING ->        motoring  ->  motor

If the second or third of the rules in Step 1b is
successful, the following is done:

    AT -> ATE           conflat(ed)  ->  conflate
    BL -> BLE           troubl(ed)   ->  trouble
    IZ -> IZE           siz(ed)      ->  size

Step 2
    (m>0) IZATION ->  IZE      vietnamization ->  vietnamize

Step 3
    (m>0) ALIZE ->  AL         formalize      ->  formal

Step 4
    (m>1) AL    ->             revival        ->  reviv
```

generalizations

```
generalizations → generalization # S ->
generalization → generalize # (m>0) IZATION → IZE
generalize → general # (m>0) ALIZE → AL
general → gener # (m>1) AL
```

gener

Porter, Martin F.: „An algorithm for suffix stripping“, in: Program 14/3 (1980), S. 130–137.

https://tartarus.org/martin/PorterStemmer/def.txt

Automatische Indexierung
Statistisch basierte Verfahren

Textstatistik

Allgemeine Überlegungen

- Es besteht ein Zusammenhang zwischen der Auftretenshäufigkeit von Wörtern und deren Bedeutung für das Retrieval.
- Wichtig sind solche Wörter, die
  - Dokumente hinreichend signifikant vertreten und gleichzeitig
  - von nicht-relevanten Dokumenten trennen.

Zipf's law

Worthäufigkeit * Häufigkeitsrang = Konstante

Worthäufigkeit: Auftretenshäufigkeit eines Wortes in einer Kollektion
Häufigkeitsrang: Position in der nach Häufigkeit sortierten Liste

Beispiel:

1.	Häufigstes Wort	10.000
2.	Zweithäufigstes Wort	5.000
3.	Dritthäufigstes Wort	3.300
…		
10.000.	Zehntausend...	1

Vermutungen

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

Wortverteilung in den Kollektionen von TREC-1 (1993)

Quelle	WSJ	AP	ZIFF	FR	DOE
Größe des Korpus in MB	295	266	251	258	190
Mittelwert Wörter/Datensatz	182	353	181	313	82
Verschiedene Wörter im Korpus	156.000	198.000	174.000	126.000	186.000
Dokumenthäufigkeit = 1	65.000	90.000	86.000	59.000	96.000
Mittelwert bei Dokumenthäuf. > 1	199	174	165	106	159

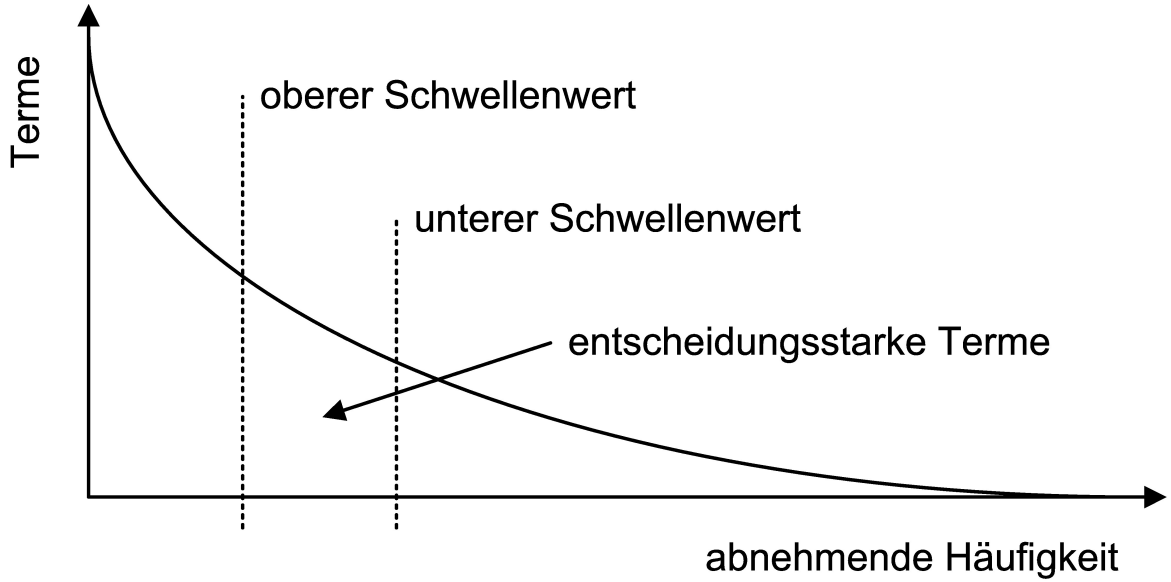


Abb. 5.6 Verteilung von Termhäufigkeiten

Termgewichtung

(1) Einfache Termhäufigkeit (TF)

„Je häufiger ein Term in einem Dokument vorkommt, umso wichtiger ist er für dieses Dokument."

Termhäufigkeit = Häufigkeit Term je Dokument

(2) Relative Termhäufigkeit (WDF)

„Die einfache Termhäufigkeit bevorzugt lange Dokumente, interessant ist also die Termhäufigkeit in Relation zur Länge des Dokuments."

WDF = Häufigkeit Term je Dokument / Gesamtzahl Terme im Dokument

(3) Inverse Dokumenthäufigkeit (IDF)

„Terme, die in wenigen Dokumenten vorkommen, sind wichtiger als Terme, die in sehr vielen oder fast allen Dokumenten vorkommen (Bsp. Funktionswörter)"

IDF = 1 / Häufigkeit der Dokumente je Term

(4) Termhäufigkeit * Inverse Dokumenthäufigkeit (TF*IDF)

„Terme, die in relativ wenigen Dokumenten relativ häufig vorkommen, sind wichtig."

TF*IDF = Termhäufigkeit bzw. WDF / Dokumenthäufigkeit

http://ixtrieve.fh-koeln.de/lir/cgi-bin/lir.pl?db=litie

LIR – "Lehr- und Lernsystem Information Retrieval"

[New query] | Query [Help] | [LIR home]

Searching database: Literatur zur Informationserschließung [Change]

Query[?]:

Options[?]: Regular expression | End-truncated | Case-sensitive

Ranking[?]: Kascade komplex Threshold: @100

Custom[?]: tf * log(N / df)

Search | Reset

Literatur zur Informationserschließung (42.740 records)

Search tips:

- Query will be performed as "ranked query"[?]
- Try regular expressions[?]
- +/-TERM = TERM *must* / *must not* occur
- TERM^NUMBER = TERM's weight gets multiplied by NUMBER (=> *boost factor*)
- Custom ranking[?]:
  tf = Term frequency
  df = Document frequency
  N = Number of documents in the collection
- Threshold:
  NUMBER(%) = Display only documents with at least a weight of NUMBER (percent of the maximum weight)
  @NUMBER(%) = Display top NUMBER (percent) documents

Sample query: +indexierung ranking.*^1.5 automatisch^-1.5 -thesaurus

[New query] | Query [Help] | [LIR home]

lir.pl v0.7

0.0056 seconds

LIR – "Lehr- und Lernsystem Information Retrieval"

[New query] | Query | Results ! [Help] | [LIR home]

Searching database **"Literatur zur Informationserschließung"**

Query: indexierung

Options: Regular expression End-truncated Case-sensitive

Ranking: IDF Threshold: @100

Search Reset

Art der Termgewichtung

Displaying results **1 - 100** (**100** of 241 results)

Select ranking: IDF and threshold: @100 Re-rank

View Search similar **Relevance Feedback**

1. *(231.)* 00059 **@10.2192** *(4.6068)* > Ranking-Experimente mit gewichteter Indexierung **[indexierung]**
2. *(217.)* 00282 **@10.2192** *(4.898)* > Syntax und Gewichtung in Informationssprachen : Ein Fortschrittsbericht über präzisere Indexierung und Computer-Suche **[indexierung]**
3. *(136.)* 00670 **@10.2192** *(5.1434)* > Transparente Indexierungsstrukturen im Fach Literaturwissenschaft **[indexierung]**
4. *(215.)* 00833 **@10.2192** *(4.9658)* > DIN 31623: Indexierung zur inhaltlichen Erschließung von Dokumenten : T.1: Begriffe, Grundlagen; T.2: Gleichordnende Indexierung mit Deskriptoren; T.3: Syntaktische Indexierung mit Deskriptoren **[indexierung]**
5. *(232.)* 00930 **@10.2192** *(4.6068)* > Entwicklung und Fortschritt bei Klassifikation und Indexierung **[indexierung]**
6. *(126.)* 00954 **@10.2192** *(5.1458)* > Automatische Indexierung zwischen Forschung und Anwendung **[indexierung]**
7. *(160.)* 01461 **@10.2192** *(5.1324)* > Probabilistische Modelle in Information-Retrieval-Systemen **[indexierung]**
8. *(44.)* 01490 **@10.2192** *(5.1718)* > Wäre es nicht langsam Zeit, die Informationstechnologie in der bibliothekarischen Sacherschließung etwas erster zu nehmen? : ein Wort zur RSWK **[indexierung]**
9. *(127.)* 01528 **@10.2192** *(5.1452)* > ¬Die Pilotstudie "DB-Thesaurus" : Allgemeiner Thesaurus für Bibliotheken **[indexierung]**
10. *(192.)* 01533 **@10.2192** *(5.1092)* > Verbindliche versus freie Indexierung **[indexierung]**
11. *(218.)* 01535 **@10.2192** *(4.898)* > LyberWorld : eine 3D-basierte Benutzerschnittstelle für die computerunterstützte Informationssuche in Dokumentmengen **[indexierung]**
12. *(158.)* 01553 **@10.2192** *(5.135)* > Zukunftsperspektiven der Klassifikation und Indexierung **[indexierung]**
13. *(67.)* 01555 **@10.2192** *(5.1646)* > ¬Die Funktion von semantischen Kategorien in Indexierungssprachen und bei der Indexierung **[indexierung]**
14. *(233.)* 02006 **@10.2192** *(4.6068)* > Grundriß eines Thesaurus als funktionsfähiges Hilfsmittel für Indexierung und Recherche **[indexierung]**
15. *(151.)* 02319 **@10.2192** *(5.1374)* > Automatische Indexierung: Entwicklung und Perspektiven **[indexierung]**

LIR – "Lehr- und Lernsystem Information Retrieval"

[New query] | Query | Results [Help] | [LIR home]

Searching database **"Literatur zur Informationserschließung"**

Query: indexierung

Options: Regular expression End-truncated Case-sensitive

Ranking: Kascade komplex Threshold: @100

Search Reset

Neues Ranking mit verändertem Gewichtungsalgorithmus

Displaying results **1 - 100** (**100** of 241 results)

Select ranking: Kascade komplex and threshold: @100 Re-rank

View Search similar

1. *(139.)* 31099 **@5.193** *(10.2192)* > Weltkongress Bibliothek und Information, 72. IFLA-Generalkonferenz in Seoul, Korea : Aus den Veranstaltungen der Division IV Bibliographic Control, der Core Activities ICABS und UNIMARC sowie der Information Technology Section **[indexierung]**
2. *(155.)* 32695 **@5.1928** *(10.2192)* > Weltkongress Bibliothek und Information : 73. IFLA-Generalkonferenz in Durban, Südafrika. Aus den Veranstaltungen der Division IV Bibliographic Control, der Core Activities (CABS und UNIMARC sowie der Information Technology Section **[indexierung]**
3. *(118.)* 29305 **@5.1918** *(10.2192)* > Weltkongress Bibliothek und Information, 71. IFLA-Generalkonferenz in Oslo, Norwegen : Aus den Veranstaltungen der Division IV Bibliographic Control, der Core Activities ICABS und UNIMARC sowie der Information Technology Section **[indexierung]**
4. *(124.)* 30144 **@5.1912** *(10.2192)* > ¬Die Volltextabfrage und das Alleinstellungsmerkmal des physischen Buches **[indexierung]**
5. *(143.)* 31276 **@5.1908** *(10.2192)* > Durchführung von Digitalisierungsprojekten in Bibliotheken **[indexierung]**
6. *(108.)* 27877 **@5.1898** *(10.2192)* > Indexierung von Online-Katalogen : Ein gemeinsames Konzept der ALEPH-Anwender in Berlin **[indexierung]**
7. *(162.)* 33828 **@5.1896** *(10.2192)* > Neue Plattform für Indexierung und Retrieval : Studenten erstellen die Webseite »iXtrieve« **[indexierung]**
8. *(176.)* 35150 **@5.1896** *(10.2192)* > Zwischenbilanz Collaborative Catalog Enrichment **[indexierung]**
9. *(138.)* 31082 **@5.188** *(10.2192)* > Automatische Indexierung des Reallexikons zur Deutschen Kunstgeschichte **[indexierung]**
10. *(142.)* 31218 **@5.186** *(10.2192)* > Information und Sprache : Beiträge zu Informationswissenschaft, Computerlinguistik, Bibliothekswesen und verwandten Fächern. Festschrift für Harald H. Zimmermann. Herausgegeben von Ilse Harms, Heinz-Dirk Luckhardt und Hans W. Giessen **[indexierung]**
11. *(99.)* 27319 **@5.1858** *(10.2192)* > Suchmaschine mit Mehrwert : Mirago **[indexierung]**
12. *(136.)* 31079 **@5.1844** *(10.2192)* > Indexieren mit AUTINDEX **[indexierung]**
13. *(95.)* 27185 **@5.1842** *(10.2192)* > Sacherschließung - wir müssen sie (uns) leisten! : Vorträge im Rahmen der 28. Jahrestagung der Gesellschaft für Klassifikation, Universität Dortmund 9. bis 11. Mai 2004 **[indexierung]**
14. *(107.)* 27876 **@5.1832** *(10.2192)* > Weltkongress Bibliothek und Information, 70. IFLA-Generalkonferenz in Buenos Aires : Aus den Veranstaltungen der Division IV Bibliographic Control, der Core Activities ICABS und UNIMARC sowie der Information Technology Section **[indexierung]**
15. *(119.)* 29306 **@5.1826** *(10.2192)* > Weltkongress Bibliothek und Information, 71. IFLA-Generalkonferenz in Oslo, Norwegen : Aus den Veranstaltungen der Division IV Bibliographic Control, der Core Activities ICABS und UNIMARC sowie der Information Technology Section **[indexierung]**

LIR – "Lehr- und Lernsystem Information Retrieval"

[New query] | Query | Results [Help] | [LIR home]

Searching database **"Literatur zur Informationserschließung"**

Query: indexierung

Options: Regular expression End-truncated Case-sensitive

Ranking: Kascade komplex Threshold: @100

Search Reset

Relevance Feedback über Markierung geeigneter Treffer und Ähnlichkeitssuche

Displaying results **1 - 100** (**100** of **241** results)

Select ranking: Kascade komplex and threshold: @100 Re-rank

View Search similar

1. *(139.)* 31099 **@5.193** *(10.2192)* > Weltkongress Bibliothek und Information, 72. IFLA-Generalkonferenz in Seoul, Korea : Aus den Veranstaltungen der Division IV Bibliographic Control, der Core Activities ICABS und UNIMARC sowie der Information Technology Section **[indexierung]**
2. *(155.)* 32695 **@5.1928** *(10.2192)* > Weltkongress Bibliothek und Information : 73. IFLA-Generalkonferenz in Durban, Südafrika. Aus den Veranstaltungen der Division IV Bibliographic Control, der Core Activities (CABS und UNIMARC sowie der Information Technology Section **[indexierung]**
3. *(118.)* 29305 **@5.1918** *(10.2192)* > Weltkongress Bibliothek und Information, 71. IFLA-Generalkonferenz in Oslo, Norwegen : Aus den Veranstaltungen der Division IV Bibliographic Control, der Core Activities ICABS und UNIMARC sowie der Information Technology Section **[indexierung]**
4. *(124.)* 30144 **@5.1912** *(10.2192)* > ¬Die Volltextabfrage und das Alleinstellungsmerkmal des physischen Buches **[indexierung]**
5. *(143.)* 31276 **@5.1908** *(10.2192)* > Durchführung von Digitalisierungsprojekten in Bibliotheken **[indexierung]**
6. *(108.)* 27877 **@5.1898** *(10.2192)* > Indexierung von Online-Katalogen : Ein gemeinsames Konzept der ALEPH-Anwender in Berlin **[indexierung]**
7. *(162.)* 33828 **@5.1896** *(10.2192)* > Neue Plattform für Indexierung und Retrieval : Studenten erstellen die Webseite »iXtrieve« **[indexierung]**
8. *(176.)* 35150 **@5.1896** *(10.2192)* > Zwischenbilanz Collaborative Catalog Enrichment **[indexierung]**

Relevance Feedback

Sukzessive Annäherung an die ideale Treffermenge durch

- Auswahl geeigneter Dokumente in der Treffermenge
- erneute Suche des Systems nach „ähnlichen“ Dokumenten (bspw. durch eine Suche mit gemeinsamen hochgewichteten Indextermen)

LIR – "Lehr- und Lernsystem Information Retrieval"

[New query] | Query | Results | Records [Help] | [LIR home]

Searching database **"Literatur zur Informationserschließung"** neue Suche

Query: basis-technologie begriffsrecherche bibliothekskatalog bibliotheksverbünden datenbank-management-system derivationsmorp

Options: Regular expression End-truncated Case-sensitive

Ranking: Kascade komplex Threshold: @100

Search Reset

Relevance Feedback

Displaying results **1 - 100** (**100** of 295 results)

Select ranking: Kascade komplex and threshold: @100 Re-rank neue Treffermenge durch Ähnlichkeitssuche

View Search similar

1. 35150 **@55.1416** > Zwischenbilanz Collaborative Catalog Enrichment **[basis-technologie bibliotheksverbünden datenbank-management-system indexierungsergebnis inhaltsverzeichnis kataloganreicherung perspektivenwechsel retrieval-technologie sammlungsorganisation sprachverarbeitungskonzept]**
2. 31276 **@55.0066** > Durchführung von Digitalisierungsprojekten in Bibliotheken **[bibliothekskatalog derivationsmorphologie digitalisierungsprojekt erschließungsmöglichkeit flexionsmorphologie indexierungssprache indexierungsverfahren indexterminologie rechtschreibprüfung unregelmäßigkeit]**
3. 30144 **@54.2986** > ¬Die Volltextabfrage und das Alleinstellungsmerkmal des physischen Buches **[begriffsrecherche online-suchabfrage qualitätssteigerung registererstellung texterschließung trefferqualität verweisungsbegriff volltextabfragen volltextretrieval wissenstheoretisch]**
4. 35151 **@16.5346** > Vom Katalog zur Bibliothek : Zwischenschritt und Zwischenstand "Kataloganreicherung" **[basis-technologie datenbank-management-system kataloganreicherung]**
5. 36716 **@16.4342** > Kataloganreicherung auch für Reihe B der Deutschen Nationalbibliografie **[bibliotheksverbünden inhaltsverzeichnis kataloganreicherung]**
6. 31041 **@16.4222** > Kosten und Nutzung der Optimierung von Erschließung **[indexierungsverfahren inhaltsverzeichnis kataloganreicherung]**
7. 34220 **@16.3558** > Kataloganreicherungsdaten der Deutschen Nationalbibliothek für Dritte zugänglich **[bibliothekskatalog inhaltsverzeichnis kataloganreicherung]**
8. 34488 **@16.3556** > Kataloganreicherungsdaten der Deutschen Nationalbibliothek für Dritte zugänglich **[bibliothekskatalog inhaltsverzeichnis kataloganreicherung]**
9. 28558 **@16.3538** > Aus der 48. Sitzung der Arbeitsgemeinschaft der Verbundsysteme am 12. und 13. November 2004 in Göttingen **[bibliothekskatalog inhaltsverzeichnis kataloganreicherung]**
10. 27434 **@16.3496** > In Bibliothekskatalogen "googlen" : Integration von Inhaltsverzeichnissen, Volltexten und WEB-Ressourcen in Bibliothekskataloge **[bibliothekskatalog inhaltsverzeichnis kataloganreicherung]**

LIR – Help

Help contents

Quick tips

- The query will be performed as "ranked query" (or "best match query", as opposed to "exact match query").
- There's no support for (hierarchical) grouping by parentheses; phrases ("...") are supported though.
- Regular expressions are supported!
- Terms beginning with a + sign **must** occur.
- Terms beginning with a – sign **must not** occur.
- You can modify ("boost") a term's weight by appending "^" plus value to the term which will be multiplied with the term's initial weight (negative values are possible as well).
- Sample query: `+indexierung ranking.*^1.5 automatisch^-1.5 -thesaurus`
  Will find documents containing "indexierung" and preferably "ranking" (the ".*" indicates end-truncation) but not "thesaurus", any occurence of "automatisch" will decrease the document's rank.

Ranked query

A *ranked query* or *best match query*, as opposed to *exact match query* or *boolean query*, is a means of improving query results by assigning each result a weight according to its relevance for this particular query and displaying the results in descending order of relevance.
Thus, the most relevant documents will be presented at the top of the result list, without having less relevant documents completely omitted as is the case with boolean queries, where documents not containing *all* of the query terms simply fall short. [1]
In so doing, *recall* gets immensely increased while some kind of *precision* is still achieved through the relevance ranking.

Yet still problematic is the concept of *relevance* [2] of a certain document for a particular query. The usual approach is to assign weights to the index terms and generate a document's weight (which acts as a measure for its relevance) from the matching term's weights.
lir.pl offers using various term weights calculated by different algorithms[?] [3] to allow for comparison of their effects and appropriateness as relevance measures.

Algorithms

Salton
Kascade einfach
Kascade komplex
Robertson
IDF
Custom
None

Termgewichte und Erschließung - Automatisches Klassifizieren

Verfahren

- Erstellen einer Kollektion mit intellektuell erschlossenen (klassifizierten) Dokumenten (bestehend aus Dokumenten in Klassen) = „Lernkollektion“
- 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 (und hoch gewichteten) Terme einer Klasse
- Ermittlung der Häufigkeit dieser Terme in anderen Klassen
- Verwendung der geeigneten Terme als Repräsentationen der Klasse (idealerweise Terme, die für eine Klasse typisch sind, für andere nicht)

Ergebnis

Klassen und Terme, die die Klassen repräsentieren

Erstellen und Optimieren des Klassifizierers

- Festlegung der Bedingungen (Regeln), die zur Zuweisung eines Dokuments zu einer Klasse führen sollen, beispielsweise:
  - wie viele Terme einer Klasse müssen mindestens im Dokument enthalten sein
  - welche Gewichte müssen diese haben
- Testen des Klassifizierers anhand einer intellektuell erschlossenen Testkollektion
- ggf. Modifizierung der Regeln

Ergebnis

Klassifizierer, der algorithmisch die intellektuelle Erschließungspraxis der Testkollektion „simuliert“

Einsatz des Klassifizierers

- Anwendung des Klassifizierers für nicht erschlossene Dokumente: Zuteilung von Notationen eines Klassifikationssystems

Evaluierung des Klassifizierers

- idealerweise Evaluierung des Ergebnis durch Retrievaltests

Golub, Koraljka, Johan Hagelbäck und Anders Ardö: „Automatic classification using DDC on the swedish union catalogue“, in: Mayr, Philipp u. a. (Hrsg.): Preface of the 18th European Networked Knowledge Organization Systems Workshop (NKOS 2018), Porto: CEUR 2018 (CEUR workshop proceedings, vol 2020), S. 4–16.

Reiner, Ulrike: „Automatische DDC-Klassifizierung: bibliografische Titeldatensätze der Deutschen Nationalbibliografie“, in: Dialog mit Bibliotheken 22/1 (2010), S. 23–29.

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

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	

Erzeugen 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	

Anwenden eines ‚**Clusteralgorithmus**' zur Verteilung der Dokumente auf Cluster

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

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 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

Maybury, M und G Kowalski: Information storage and retrieval systems : theory and implementation, Boston, Mass.: Kluwer 2000.

Latent Semantic Indexing

Idee

- im Vektorraummodell sind Beziehungen zwischen Dokumenten und gewichteten Termen berechenbar:
  - ähnliche Dokumente werden durch gemeinsame hochgewichtete Terme charakterisiert
  - ähnliche Terme werden durch ihr gemeinsames Auftreten in verschiedenen Dokumente charakterisiert
- Cluster ähnlicher Terme entsprechen „Konzepten“
- Konzepte enthalten Synonyme, wodurch Indexierungen und Suchen mit Synonymen unterstützt werden

Deerwester, S, S Dumais und T Landauer: „Indexing by latent semantic analysis“, in: Journal of the American Society for Information Science 41/6 (1990), S. 391–407.

https://www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2006-07/IR/folien/4Folie_02_IRmodels_d.pdf

Technical Memo Example

Titles:

c1: *Human* machine *interface* for Lab 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, unordered *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*

Terms	Documents								
	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

Das Vektorraum-Modell

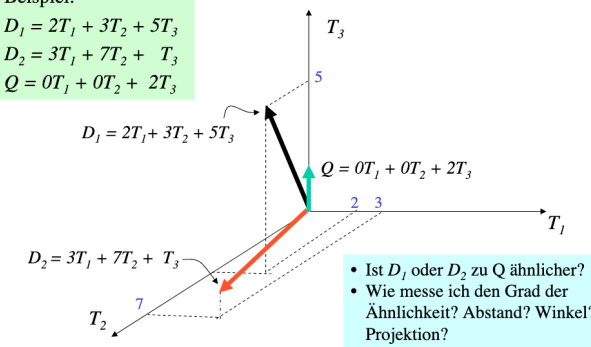
- Gehe davon aus, das t eindeutige Terme nach der Vorverarbeitung bleiben; nenne sie Indexterme oder das Vokabular.
- Diese “orthogonalen” Terme spannen einen Vektorraum mit Dimension t auf.
- Jedem Term i in einem Dokument oder einer Anfrage j wird ein reellwertiges Gewicht w_{ij} zugeordnet (im einfachsten Fall die Anzahl des Auftretens von i in j).
- Sowohl Dokumente als auch Anfragen werden als t-dimensionale Vektoren ausgedrückt:

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

13

Grafische Darstellung

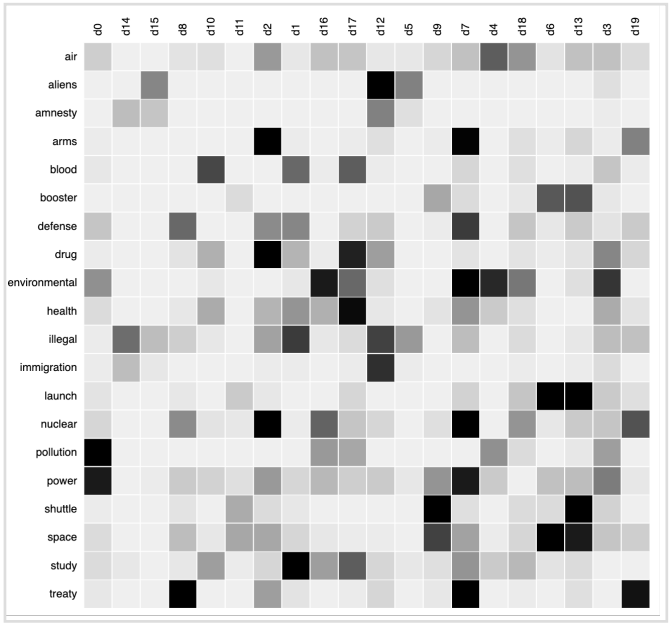
Beispiel:
$D_1 = 2T_1 + 3T_2 + 5T_3$
$D_2 = 3T_1 + 7T_2 + T_3$
$Q = 0T_1 + 0T_2 + 2T_3$



- Ist D_1 oder D_2 zu Q ähnlicher?
- Wie messe ich den Grad der Ähnlichkeit? Abstand? Winkel? Projektion?

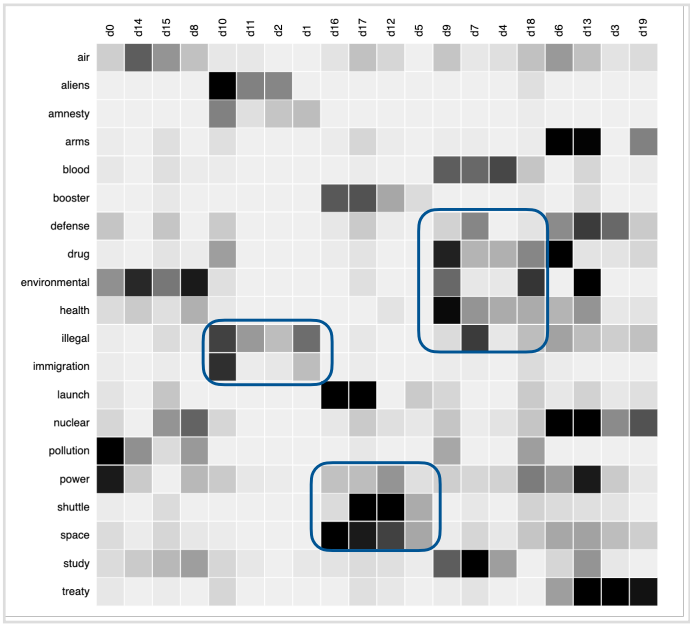
14

Term-Dokument-Matrix



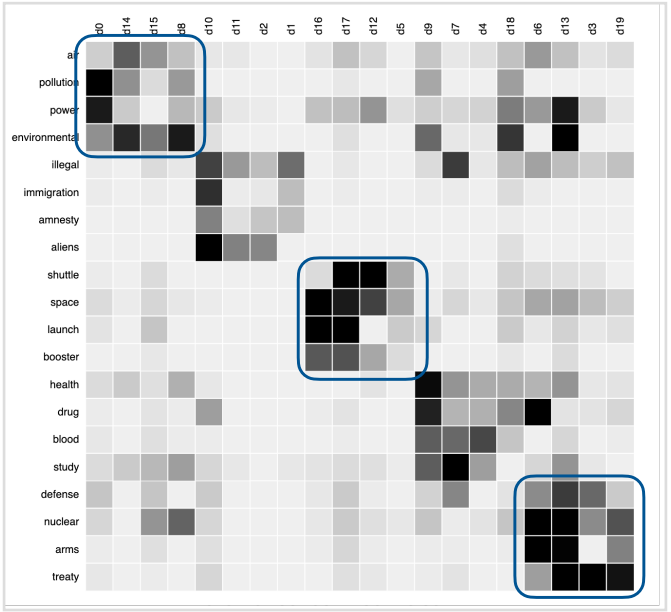
← Terme

Ordnung nach Dokumentähnlichkeit



↑ Dokumente

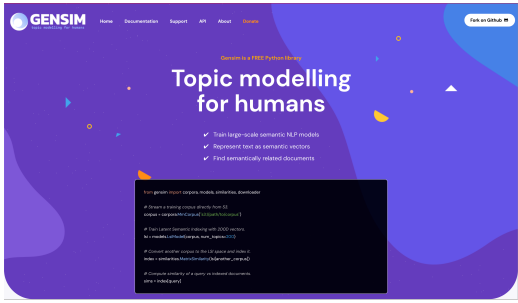
Ordnung nach Termähnlichkeit



„ähnliche“ Dokumente

„ähnliche“ Terme

Konzepte



https://topicmodels.west.uni-koblenz.de/ckling/tmt/svd_ap.html

https://radimrehurek.com/gensim/index.html

Automatisches Erschließen

Automatische Zuteilung von normiertem Vokabular

Link zu diesem Datensatz	http://d-nb.info/1000028488
Art des Inhalts	Hochschulschrift
Titel	Unternehmerische Ordnungsverantwortung : Potenzial und Grenzen einer veränderten Rolle von Unternehmen in der modernen Gesellschaft / Johanna Braun
Person(en)	Braun, Johanna (Verfasser)
Verlag	Berlin : wvb. Wiss. Verl.
Zeitliche Einordnung	Erscheinungsdatum: 2009
Umfang/Format	XIX, 225 S. : graph. Darst. ; 23 cm, 410 gr.
Hochschulschrift	Zugl.: Halle (Saale), Univ., Diss., 2009
ISBN/Einband/Preis	978-3-86573-504-1 kart. : EUR 35.00
EAN	9783865735041
Sprache(n)	Deutsch (ger)
Beziehungen	Ökonomik und Ethik ; Bd. 11
Schlagwörter	Wirtschaftsordnung ; Wirtschaftsethik ; Unternehmensethik ; Corporate Social Responsibility
DDC-Notation	174.4 [DDC22ger]; 658.408 [DDC22ger]
Sachgruppe(n)	100 Philosophie ; 650 Management
Weiterführende Informationen	Inhaltsverzeichnis Kurzbeschreibung

Frankfurt	Signatur: 2010 A 24921 Bereitstellung in Frankfurt
Leipzig	Signatur: 2010 A 46327 Bereitstellung in Leipzig

Braun, Johanna: *Unternehmerische Ordnungsverantwortung : Potenzial und Grenzen einer veränderten Rolle von Unternehmen in der modernen Gesellschaft.*
Berlin: wvb. Wiss. Verl., 2009, XIX, 225 S.; graph. Darst
Zugl.: Halle (Saale), Univ., Diss., 2009
(Ökonomik und Ethik ; 11)

GND-Schlagwörter:
s Wirtschaftsordnung -- 9(DE-588)4066475-2; s Wirtschaftsethik -- 9(DE-588)4066439-9; s Unternehmensethik -- 9(DE-588)4202404-3; s Corporate Social Responsibility -- 9(DE-588)7697760-2

Maschinelle SLW: s Verantwortung; 040625478; 0.03378; s Unternehmen; 04061963X; 0.01504
VLB-Schlagwörter:
Ökonomische Ethik; Unternehmerische Verantwortung; Unternehmerische Praxis; Immaterielle Vermögenswerte; BC: Paperback; 1784: Hardcover, Softcover / Wirtschaft/Management

DDC: *174.4 -- c 174.4 -- eDDC22ger; 658.408 -- c 658.408 -- eDDC22ger*
DNB-Sachgruppen: *100; 650; 100; 650*

Munkelt, Johanna, Philipp Schaer und Klaus Lepsky: „Towards an IR test collection for the German National Library“, in: Gemulla, Rainer u. a. (Hrsg.): LWDA 2018 : Lernen, Wissen, Daten, Analysen 2018 ; Proceedings of the Conference „Lernen, Wissen, Daten, Analysen“ Mannheim, Germany, August 22-24, 2018, Bd. 2191, Mannheim: CEUR-WS 2018 (CEUR-WS), S. 275–280.
http://ceur-ws.org/Vol-2191/paper31.pdf

Ressourcen für automatische Verfahren

Titelstichwörter **Verlagsschlagwörter** **Scans von Inhaltsverzeichnissen**

Braun, Johanna: *Unternehmerische Ordnungsverantwortung: Potenzial und Grenzen einer veränderten Rolle von Unternehmen in der modernen Gesellschaft.*
Berlin: wvb. Wiss. Verl., 2009, XIX, 225 S.; graph. Darst
Zugl.: Halle (Saale), Univ., Diss., 2009
(Ökonomik und Ethik ; 11)

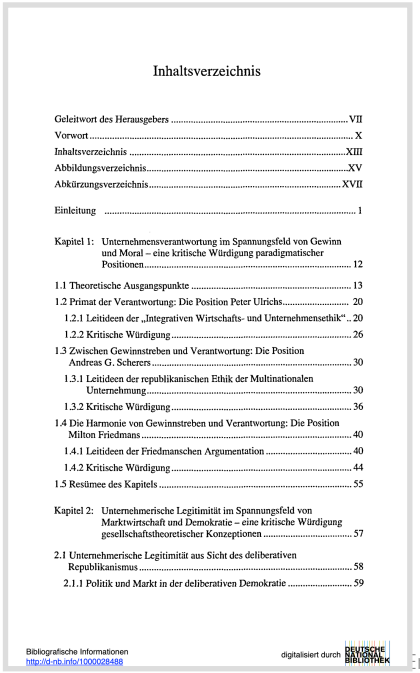
intellektuell vergebene Schlagwörter?

GND-Schlagwörter:
s Wirtschaftsordnung -- 9(DE-588)4066475-2; s Wirtschaftsethik -- 9(DE-588)4066439-9; s Unternehmensethik -- 9(DE-588)4202404-3; s Corporate Social Responsibility -- 9(DE-588)7697760-2

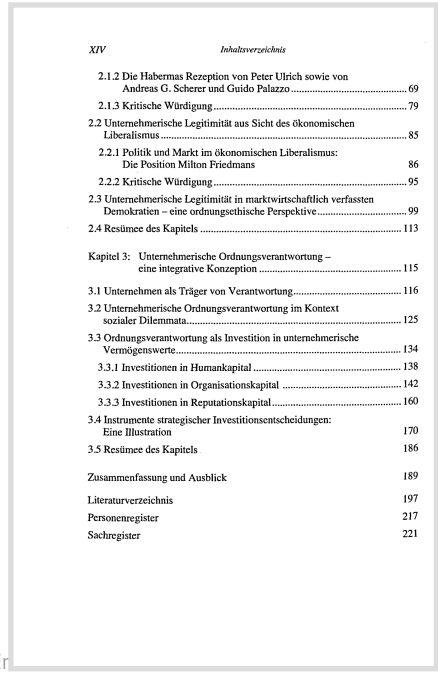
Maschinelle SLW: s Verantwortung; 040625478; 0.03378; s Unternehmen; 04061963X; 0.01504
VLB-Schlagwörter:
Ökonomische Ethik; Unternehmerische Verantwortung; Unternehmerische Praxis; Immaterielle Vermögenswerte; BC: Paperback; 1784: Hardcover, Softcover / Wirtschaft/Management

DDC: *174.4 -- c 174.4 -- eDDC22ger; 658.408 -- c 658.408 -- eDDC22ger*
DNB-Sachgruppen: *100; 650; 100; 650*

normiertes Vokabular mit
Typ: „s“ = Sachschlagwort
Schlagwort: Unternehmen
Identnummer: 04061963X
Konfidenzwert: 0.01504



Inhaltsverzeichnis



Mögliche Verfahrensweisen*

Linguistische Vorverarbeitung der Metadaten und Ressourcen

- OCR gescannter Ressourcen
- grammatikalische Normierung (Lemmatisierung), Dekomposition
- ggf. Entfernen unerwünschter Terme (Hochfrequenzterme)
- ggf. Gewichtung der Terme in Metadaten und Ressourcen

Linguistische Vorverarbeitung der normierten Terminologie

- ggf. Reduktion des Zeichensatzes und Entfernen störender Merkmale
- Auswertung der Synonymbeziehungen zur Synonymerkennung in Metadaten
- Auswertung der Sachgruppen für die Disambiguierung mehrdeutiger Terme
- Auswertung der hierarchischen Beziehungen für Aussagen zur Termspezifität

Abgleich von normierter Terminologie und Metadaten

- Erzeugung von potenziellen Indextermen
- Berechnung eines Qualitätsmaßes für zugeteilte Terme durch beispielsweise:
    - Berücksichtigung der Termgewichte der Terme in den Metadaten
    - Berücksichtigung der Sachgruppen der Terme
    - Berücksichtigung der Spezifität
    - Berücksichtigung bekannter Zuteilungswahrscheinlichkeiten aus erschlossenen Lernkollektionen
- Zuteilung aller Terme oberhalb eines festgelegten Wertes für die Zuteilungswahrscheinlichkeit

* Anm.: Dies ist keine Beschreibung des eingesetzten Algorithmus, da dessen genaue Arbeitsweise nicht veröffentlicht ist. Die Überlegungen basieren auf den vorhandenen bibliografischen Daten, den verfügbaren Erschließungsressourcen (Terminologie, Klassifikation) und den darin liegenden grundsätzlichen Möglichkeiten.

AIR/PHYS

Kollektion

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

Lernphase/Indexierungsidee

statistische Auswertung von 400.000 intellektuell erschlossenen Dokumenten, v.a. Untersuchung der Beziehung

Term ⇨ z ⇨ Deskriptor,

wobei z ein Maß für die Wahrscheinlichkeit ist, mit der ein Deskriptor einem Dokument (intellektuell) zugeteilt ist, wenn ein 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 Term t vorkommt

Automatische Indexierung

- Aufbau eines „Indexierungswörterbuchs“ mit folgenden Elementen:
  - 350.000 Term-Deskriptor-Beziehungen mit z > 0,3
  - 70.000 Synonymrelationen
  - 200.000 Deskriptor-Deskriptor-Beziehungen mit z > 0 (als gewichtetes Maß für das gemeinsame Auftreten von Deskriptoren bei einem Dokument)

- 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

- Erschließung von 10.000 Dokumenten/Monat
- Zuteilung von im Schnitt 12 Deskriptoren je Dokument
- semi-automatisches Verfahren: intellektuelle Nachbearbeitung mit ca. einem Drittel Korrekturbedarf

Retrievaltest (15.000 Dokumente, 300 Suchfragen)

- 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

Knorz, Gerhard: „Automatische Indexierung“, Wissensrepräsentation und Information Retrieval, Potsdam: Universität 1994, S. 138–196.

KASCADE

Ziele

- Anreicherung von bibliografischen Referenzdaten aus dem Fach Jura
- Entwicklung einer selektiven automatischen Indexierung zur gewichteten Extraktion von Deskriptoren (SELIX)
- Entwicklung einer zuverlässigen Erkennung für Themen-Aspekt-Beziehungen in Mehrwortgruppen (THEAS)
- Durchführung eines umfangreichen Retrievaltests

Anreicherung der Titeldaten

- Scanning von Inhaltsverzeichnissen von ca. 3.000 Titeln aus dem Bestand Jura zur Verbreiterung der Indexierungsbasis (Abstracts: zu selten; Sachregister: problematisch)
- OCR mit newsWorks und MILOS-Rechtschreibkontrolle

Automatische Indexierung mit SELIX

- Linguistische basierte Indexierung mit
  - Grundformermittlung und Dekomposition für Sachtitel, Schlagwörter und Volltexte der Inhaltsverzeichnisse
  - Einbeziehung von semantischen Relationen (GND)
- SELIX-Gewichtungsindexierung

Lohmann, Hartmut: *KASCADE: Dokumentanreicherung und automatische Inhaltserschließung; Projektbericht und Ergebnisse des Retrievaltests*, Düsseldorf: Universitäts- und Landesbibliothek 2000 (Schriften der Universitäts- und Landesbibliothek Düsseldorf 31).

KASCADE - Gewichtung

Salton

HfkImD(g,d) * log (nDok(g)/nAnzDok)

[Termhäufigkeit * log (Dokumenthäufigkeit / Dokumentenzahl), vgl. IDF]

Robertson

((K 1 + 1) * HfkImD(g,d) / (K + HfkImD(g,d))) *
log((nAnzDok – nDok(g) + 0.5) / (nDok(g) + 0.5))

KASCADE

Kollektionsgewicht nG1

(nG1 ermittelt, ob eine Grundform für eine Dokumentenkollektion als Indexterm geeignet ist (für alle Dokumente (einer Kollektion) gleich))

nG1(g) = 1 – nDok(g) / E(nDok(g)) (für: nDok(g) < E(nDok(g)); 0 sonst)

mit E(nDok(g)) = nAnzDok * (1 – exp(-λ)) wenn eine Poisson Zufallsverteilung angenommen wird:
P(i) = exp (-λ) * (λi/i!) λ = nColl(g) / nAnzDok

Dokumentgewicht nG2

(nG2 ermittelt, ob eine Grundform für ein Dokument als Indexterm wichtig ist)

nG2(g,d) = (p(1) * 1 + ... + p(HfkImD(g,d)) * HfkImD(g,d)) / λ

mit: P(i) = exp (-λ) * (λi/i!) λ = nColl * (nDokLen/nColLen)

Längengewicht nG3

(nG3 bevorteilt längere Wörter im Gewichtungsverfahren (unabhängig von Dokument und Kollektion))

nG3(g) = log (nGruLen(g)) / 4

Gewichtungsfunktion

nG = F 1 * nG1 + F 2 * nG2 + F 3 * nG3

(wobei F1-F3 frei wählbar sind (Standard: 1))

Hüther, Hubert: „Selix im DFG-Projekt Kascade“, in: Zimmermann, Harald H. und Volker Schramm (Hrsg.): Knowledge management und kommunikationssysteme, workflow management, multimedia, knowledge transfer - proceedings des 6. Internationalen symposiums für informationswissenschaft (ISI ’98), prag 3.-7. November 1998, Bd. 34, Hochschulverband für Informationswissenschaft 1998 (Schriften zur informationswissenschaft), S. 397–403.

KASCADE - Retrievaltest

Rahmenbedingungen

- 3.000 Referenzdatensätze aus dem Fach Jura
- alle angereichert um Inhaltsverzeichnisse im Volltext
- 60 von Juristen formulierte Suchthemen
- Testdurchführung durch Projektmitarbeiter
- Relevanzbewertung durch Juristen
- Recall-Berechnung nach Pooling-Methode

Besonderheiten bei den Suchthemen

- breite thematische Streuung – speziell neben allgemein
- viele Komposita und Mehrwortbegriffe
- viele komplexe Themen, d.h. Themenverknüpfungen
- nur 15% Einwort-Suchthemen (mit nur einem Nichtkompositum)

Ergebnisse

	Mittelwerte von		Null-Treffer-Suchen
	Recall	Precision	
Titel und Deskriptor (automatisch indexiert)	0.06	0.98	42
Titel, Deskriptor, Inhaltsverz. (nicht automatisch indexiert)	0.54	0.75	7
Titel, Deskriptor, Inhaltsverz. (automatisch indexiert)	0.92	0.70	4

Lepsky, Klaus und Harald Zimmermann: „Katalogerweiterung durch Scanning und automatische Dokumenterschließung: Ergebnisse des DFG-Projekts KASCADE“, in: Zeitschrift für Bibliothekswesen und Bibliographie 47 (2000), S. 305–316.

Weiterführende Literatur

Literatur zur Informationserschließung

Automatisches Indexieren

Automatisches Klassifizieren

Texttechnologie

Allen, James: Natural language understanding, Redwood City, Calif. [u.a.]: Benjamin/Cummings Publ. 1995.

Carstensen, Kai-Uwe: Computerlinguistik und Sprachtechnologie : eine Einführung, Heidelberg: Spektrum, Akad. Verl. 2010.

Gödert, Winfried, Klaus Lepsky und Matthias Nagelschmidt: Informationserschließung und Automatisches Indexieren : ein Lehr- und Arbeitsbuch, Berlin [u.a.]: Springer 2012 (X.media.press).

Hausser, Roland: Grundlagen der Computerlinguistik : Mensch-Maschine-Kommunikation in natürlicher Sprache, Berlin [u.a.]: Springer 2000.

Jurafsky, Daniel und James H. Martin: Speech and Language Processing : an introduction to natural language processing, computational linguistics, and speech recognition, Pearson Prentice Hall/Pearson education international 2009 (Pearson international edition).

Lobin, Henning: Computerlinguistik und Texttechnologie, Paderborn: W. Fink 2010.

Lepsky, Klaus: „Automatische Indexierung“, in: Kuhlen, Rainer, Wolfgang Semar und Dietmar Strauch (Hrsg.): Grundlagen der praktischen Information und Dokumentation: Handbuch zur Einführung in die Informationswissenschaft und -praxis, Berlin: De Gruyter 2013, S. 272–285.

Manning, Christopher und Hinrich Schütze: Foundations of statistical natural language processing, MIT Press 1999.