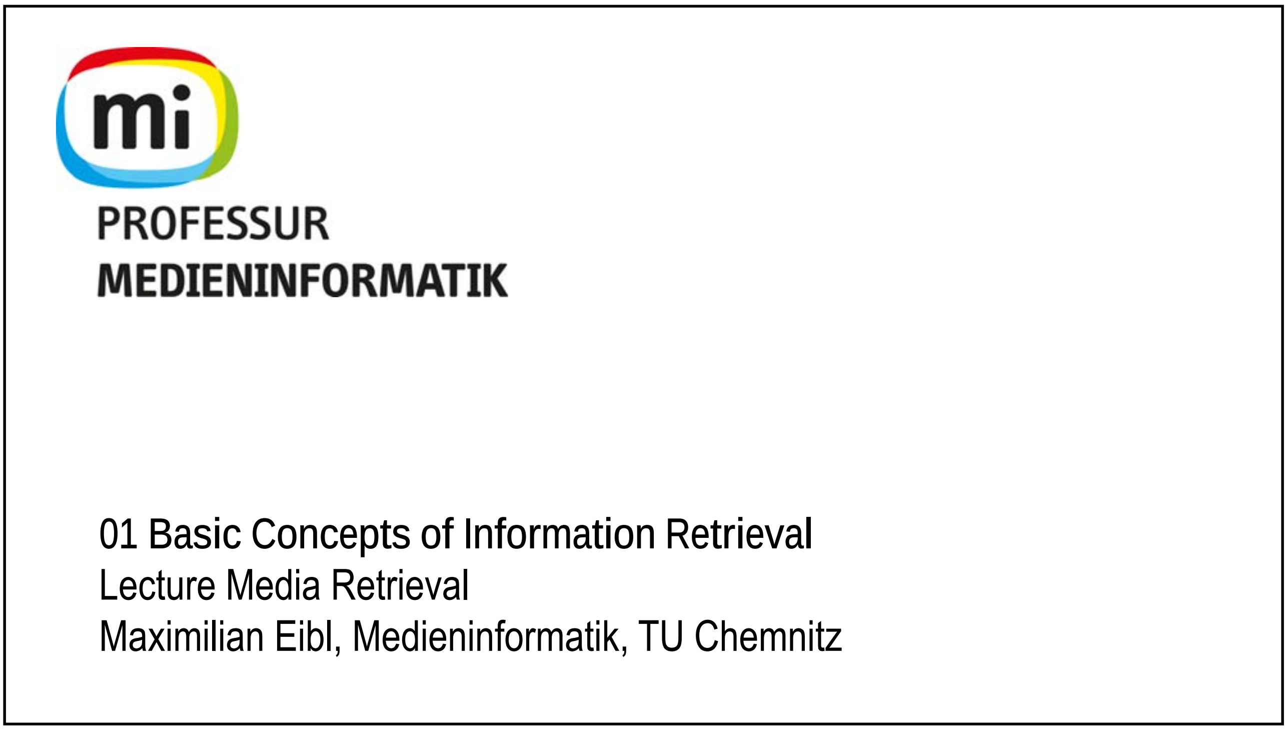
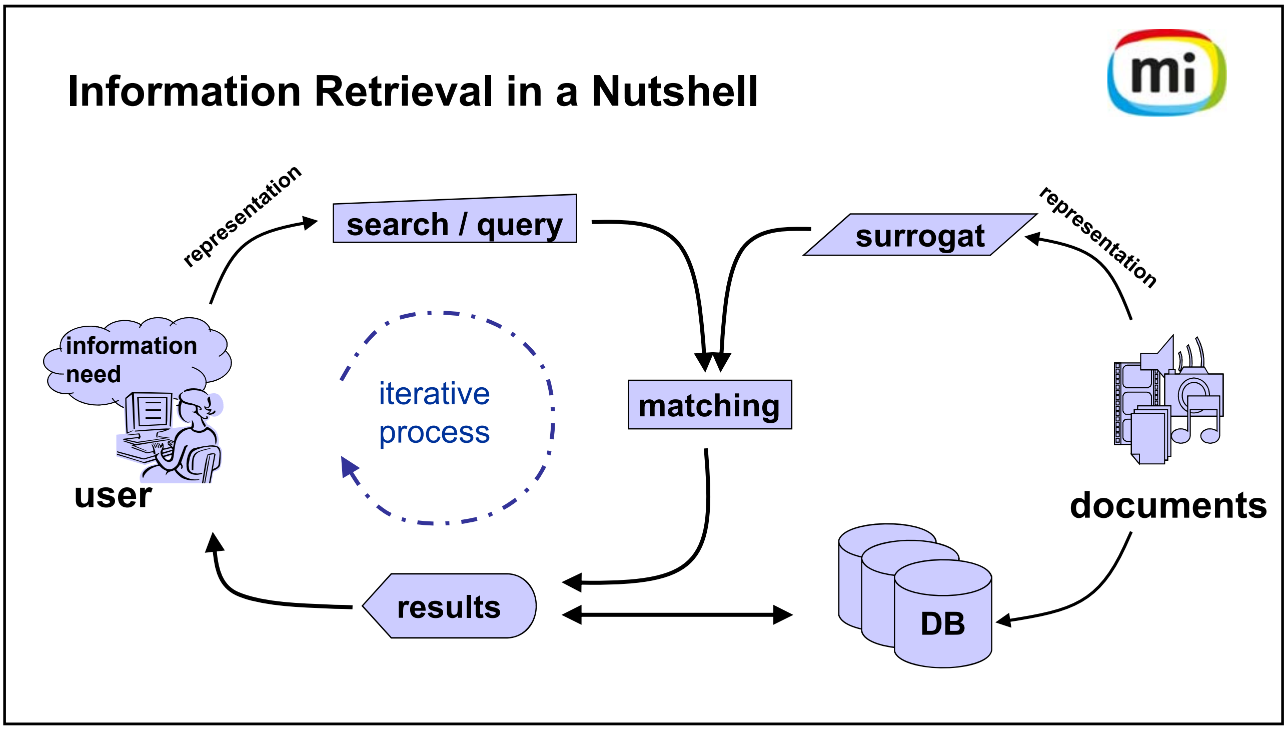
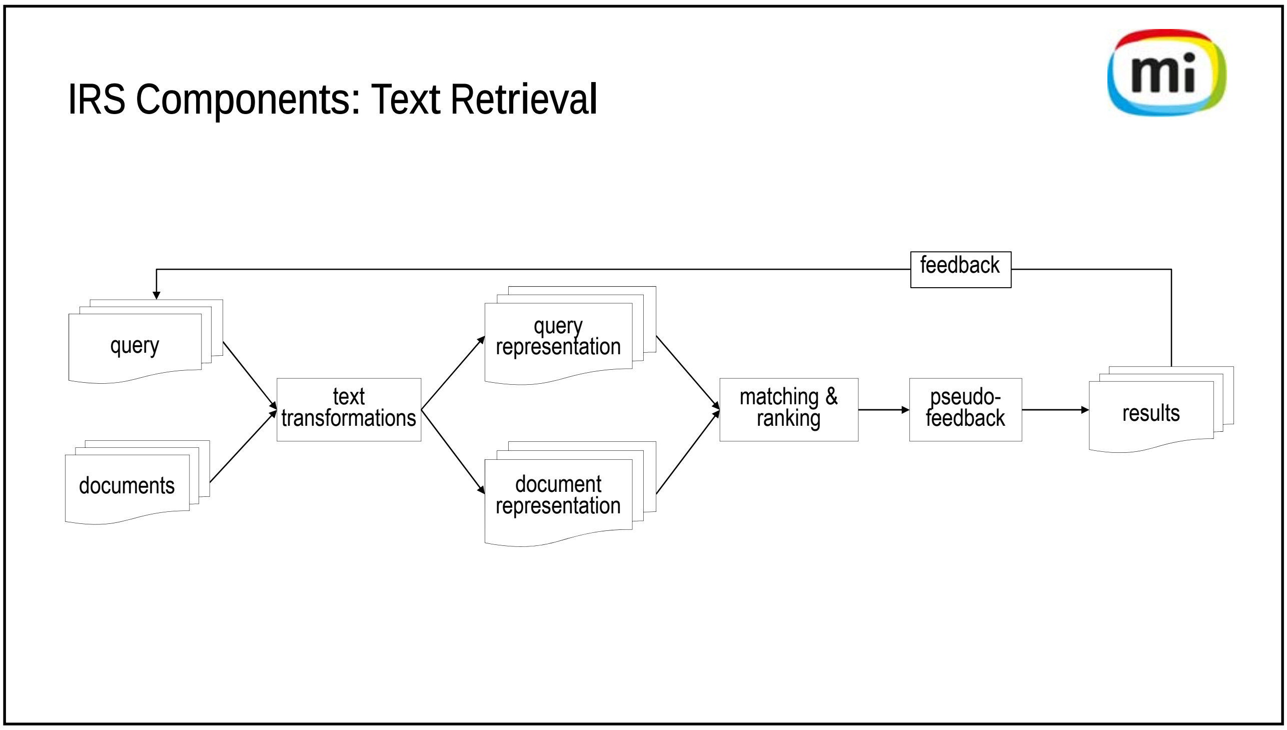


PROFESSUR
MEDIENINFORMATIK

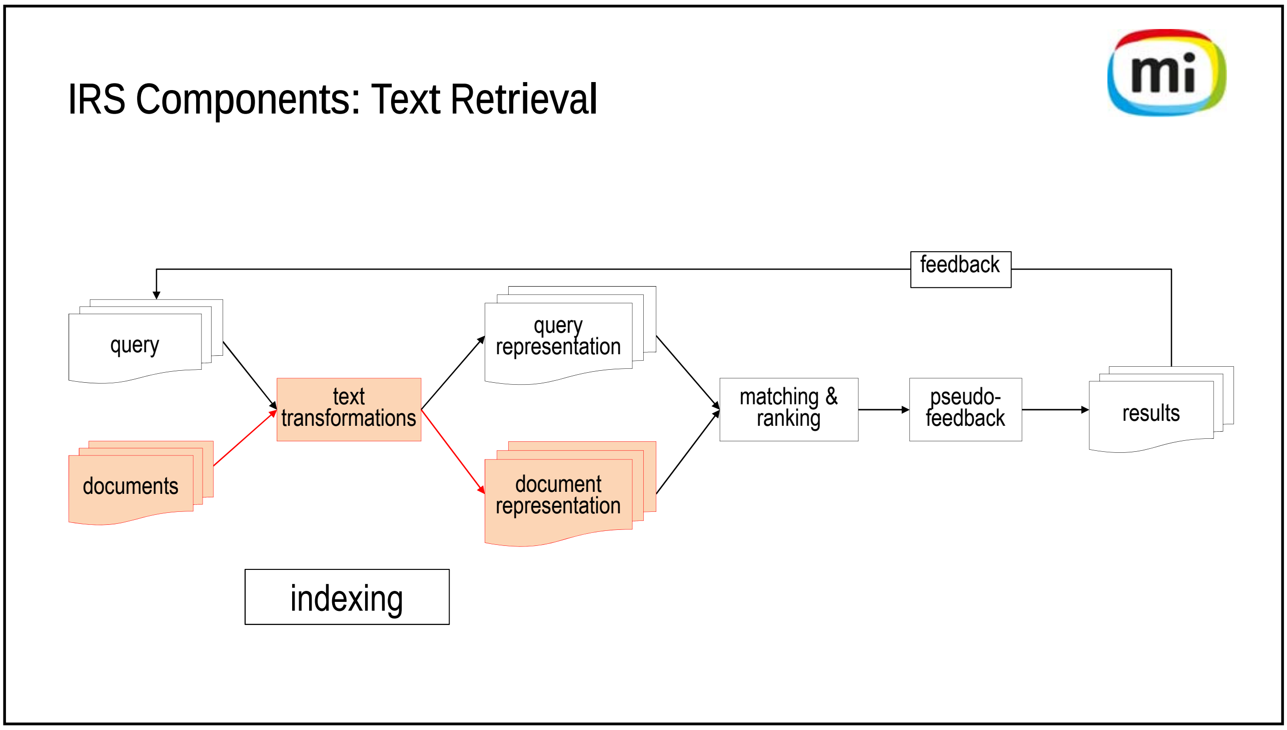
01 Basic Concepts of Information Retrieval
Lecture Media Retrieval
Maximilian Eibl, Medieninformatik, TU Chemnitz



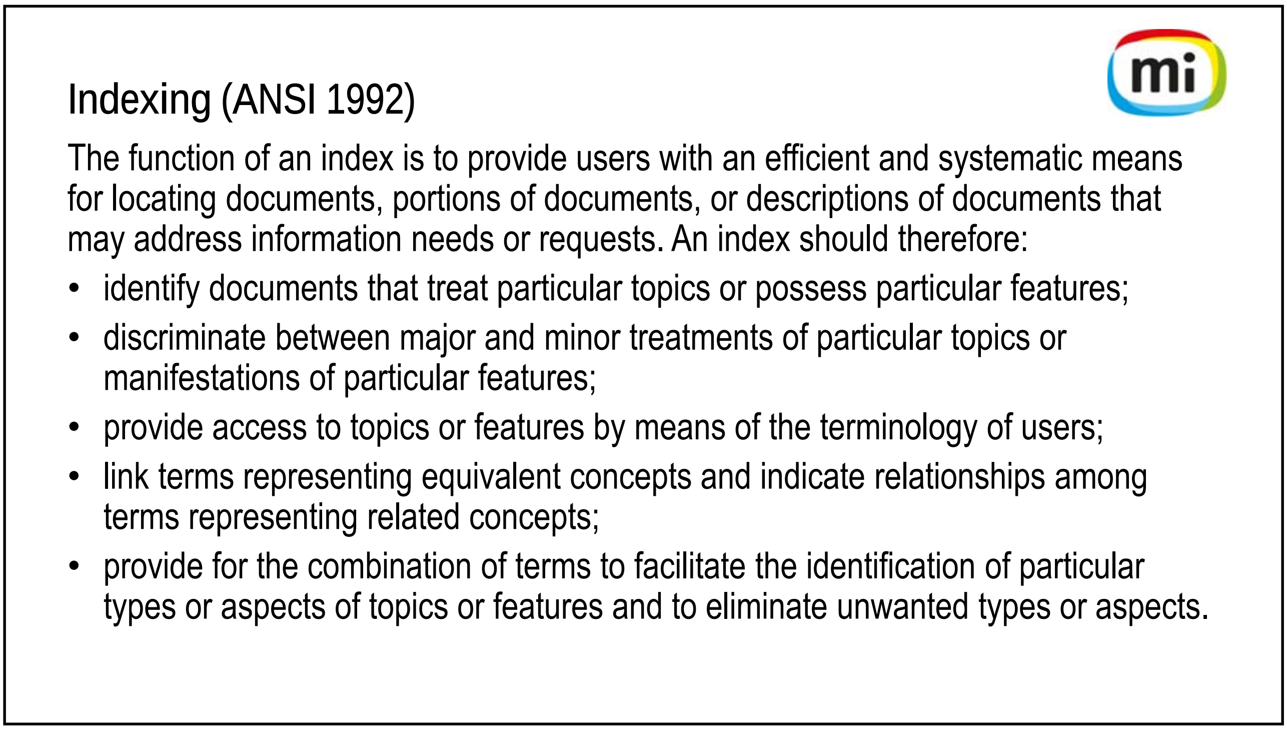
Information Retrieval in a Nutshell



IRS Components: Text Retrieval



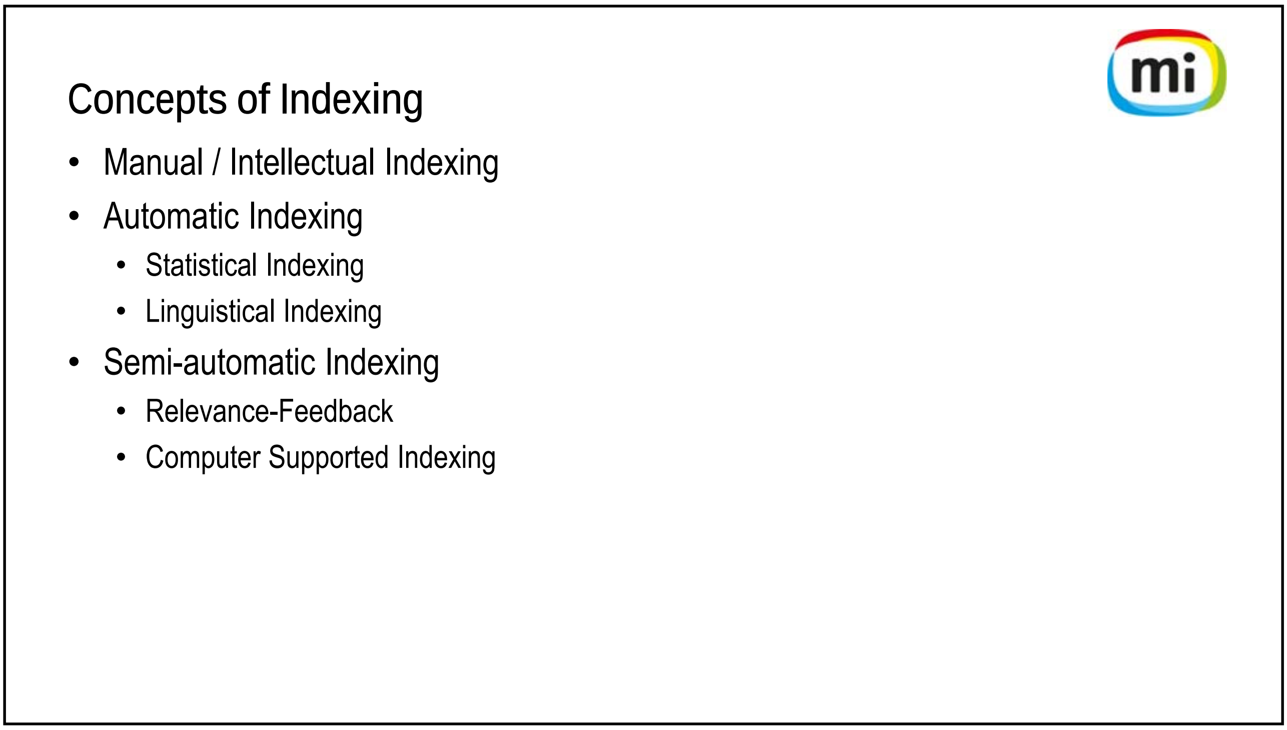
IRS Components: Text Retrieval



Indexing (ANSI 1992)

The function of an index is to provide users with an efficient and systematic means for locating documents, portions of documents, or descriptions of documents that may address information needs or requests. An index should therefore:

- identify documents that treat particular topics or possess particular features;
- discriminate between major and minor treatments of particular topics or manifestations of particular features;
- provide access to topics or features by means of the terminology of users;
- link terms representing equivalent concepts and indicate relationships among terms representing related concepts;
- provide for the combination of terms to facilitate the identification of particular types or aspects of topics or features and to eliminate unwanted types or aspects.



Concepts of Indexing

- Manual / Intellectual Indexing
- Automatic Indexing
  - Statistical Indexing
  - Linguistical Indexing
- Semi-automatic Indexing
  - Relevance-Feedback
  - Computer Supported Indexing

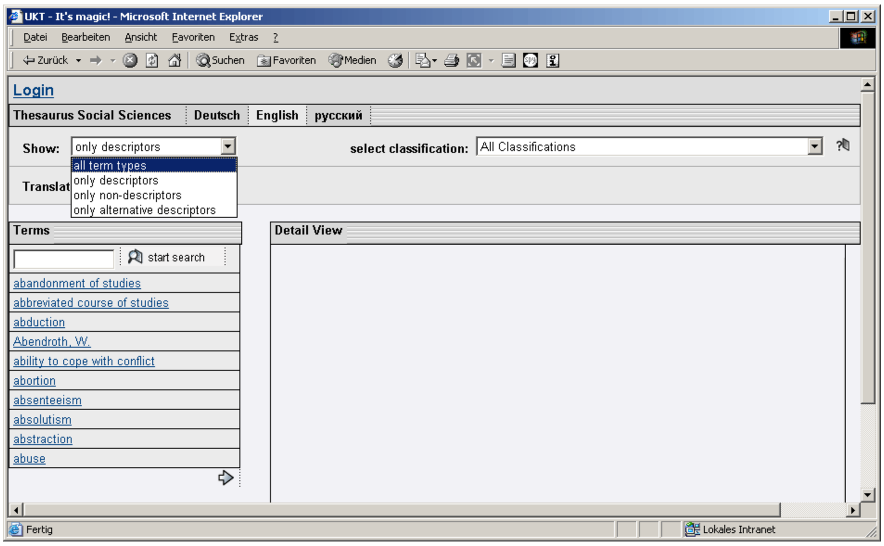
Manual / Intellectual Indexing

Thesaurus: Controlled Terms / Controlled Vocabulary

Thesaurus: Relations			
DIN 1463-1		ISO 2788	
BF	Benutzt für	UF	Used for
BS	Benutze Synonym	USE/SYN	Use synonym
OB	Oberbegriff	BT	Broader term
UB	Unterbegriff	NT	Narrower term
VB	Verwandter Begriff	RT	Related term
SB	Spitzenbegriff	TT	Top term

Example: http://sowiport.gesis.org/thesaurus

Manual / Intellectual Indexing: Thesaurus

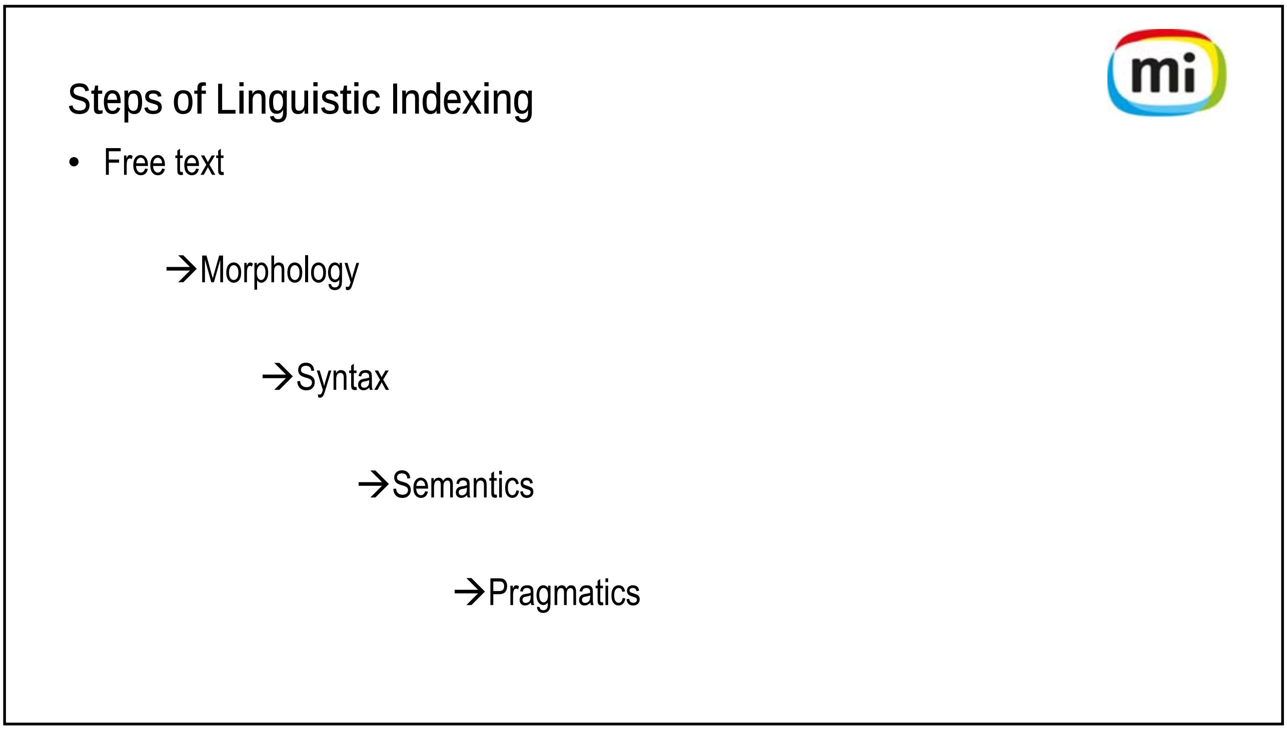

UKT - It's magic! - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück
Suchen
Favoriten
Medien
Login
Thesaurus Social Sciences
Deutsch
English
русский
Translat
Terms
start search
Detail View
Fertig
Lokales Intranet

Show: only descriptors

all term types
only descriptors
only non-descriptors
only alternative descriptors

select classification: All Classifications

abandonment of studies
abbreviated course of studies
abduction
Abendroth, W.
ability to cope with conflict
abortion
absenteeism
absolutism
abstraction
abuse



Steps of Linguistic Indexing

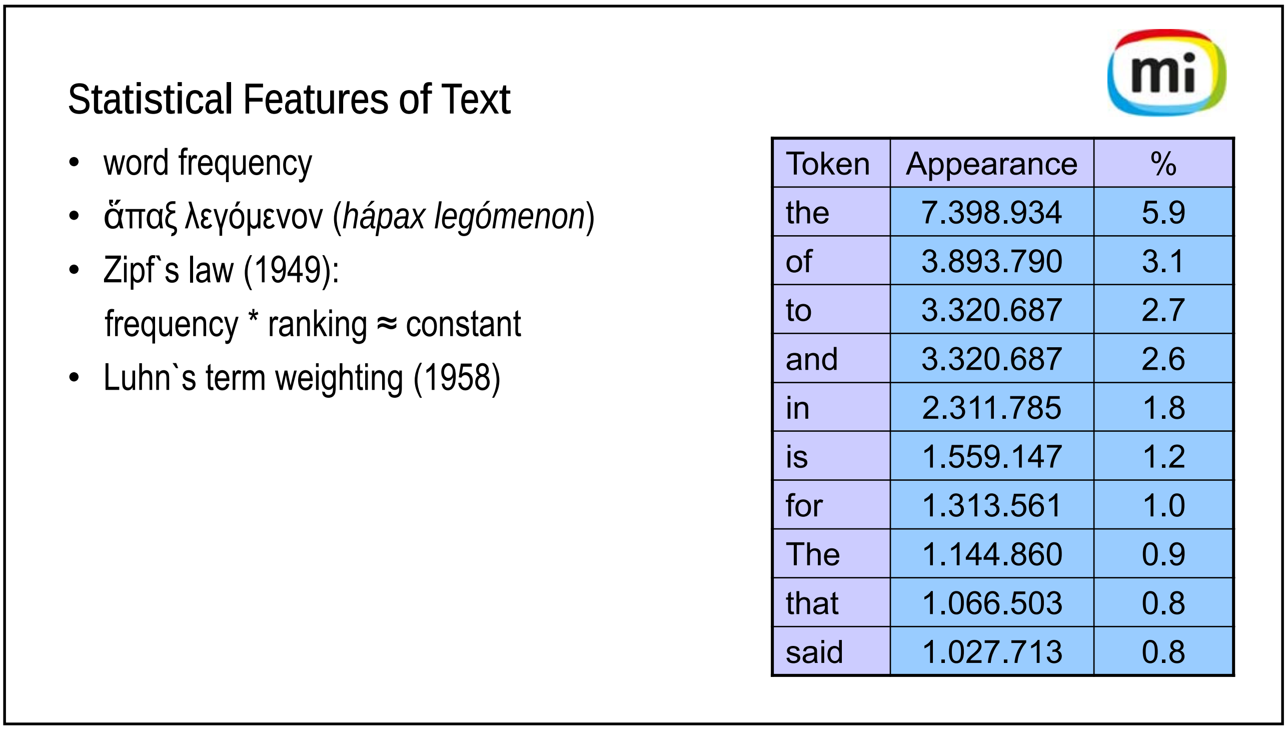
- Free text

→Morphology

→Syntax

→Semantics

→Pragmatics

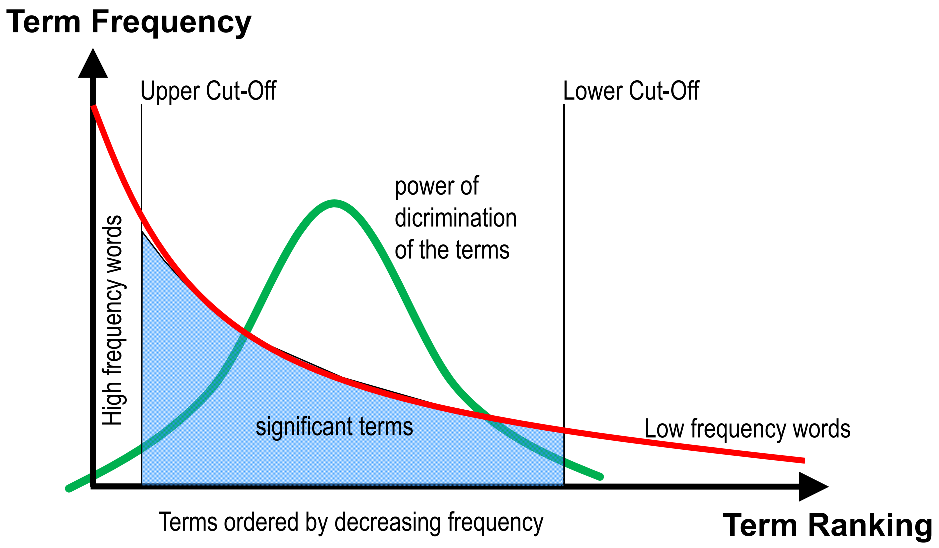


Statistical Features of Text

- word frequency
- ἅπαξ λεγόμενον (*hápax legómenon*)
- Zipf`s law (1949):
  frequency * ranking ≈ constant
- Luhn`s term weighting (1958)

Token	Appearance	%
the	7.398.934	5.9
of	3.893.790	3.1
to	3.320.687	2.7
and	3.320.687	2.6
in	2.311.785	1.8
is	1.559.147	1.2
for	1.313.561	1.0
The	1.144.860	0.9
that	1.066.503	0.8
said	1.027.713	0.8

Discriminating power of words



see:
Salton&McGill: Abb.3-2
Van Jijsbergen: Fig.2.1

Statistical Features of Text: How can we use it?

- Stop word list

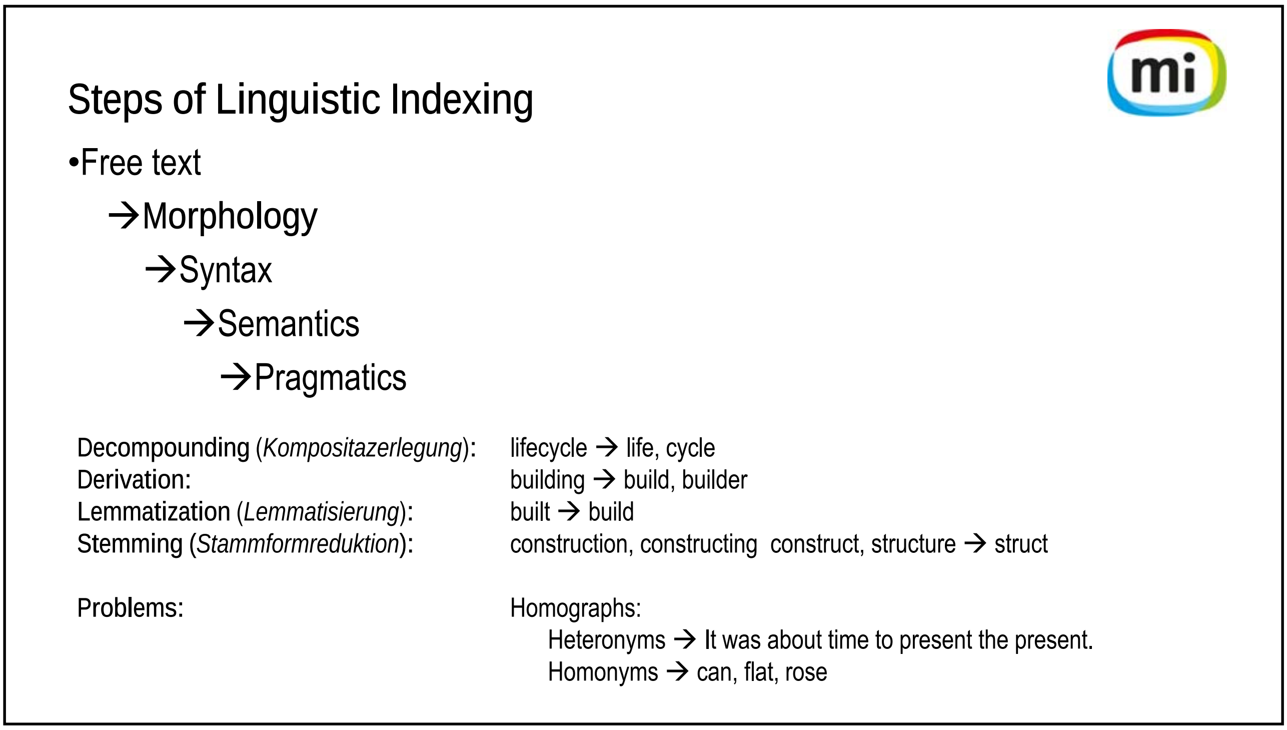
  → High frequency words (is, this, he, she, it, be, and, or, not …)

  Reduction of data approx. 40% (Baeza-Yates&Ribeiro-Neto: 167) to 70% (Salton&McGill:66)

- Low frequency words

- Later: term weighting

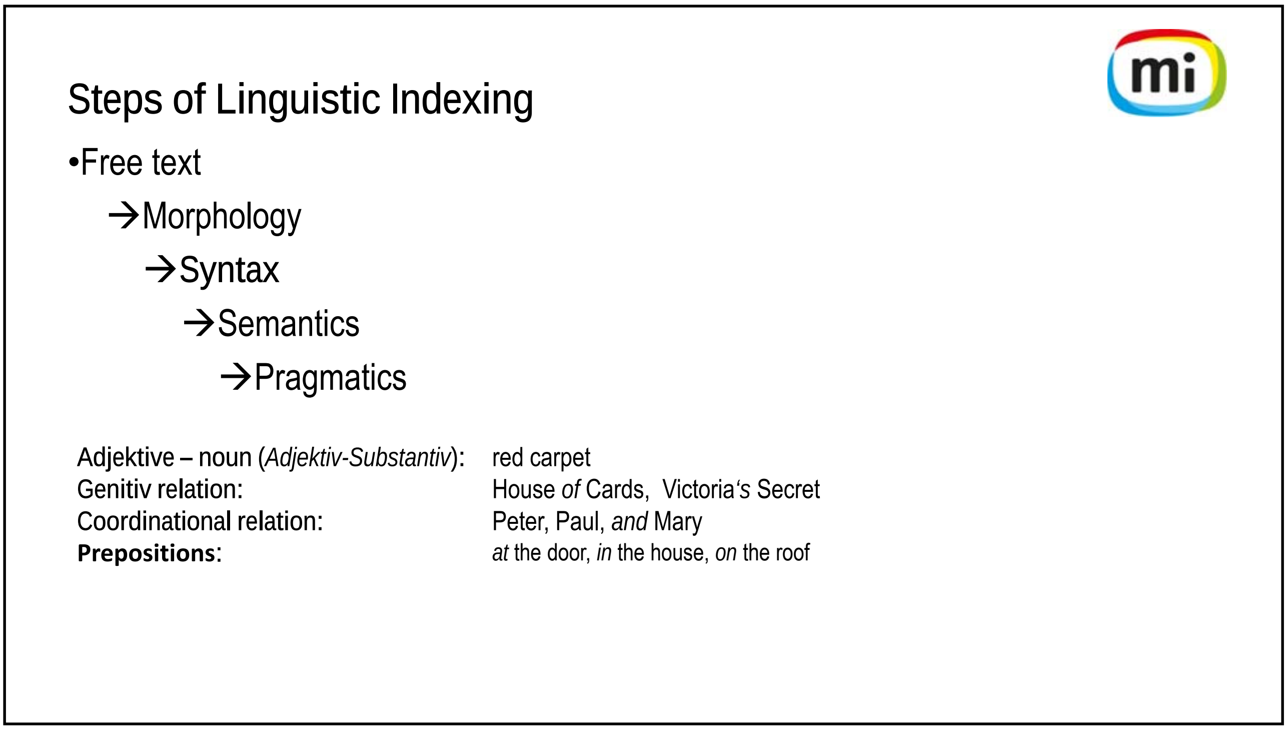
https://commons.wikimedia.org/wiki/File:Sarah-Bernhardt_(Hamlet).jpg



Steps of Linguistic Indexing

- Free text
  - → **Morphology**
    - → Syntax
      - → Semantics
        - → Pragmatics

Decompounding (*Kompositazerlegung*):	lifecycle → life, cycle
Derivation:	building → build, builder
Lemmatization (*Lemmatisierung*):	built → build
Stemming (*Stammformreduktion*):	construction, constructing construct, structure → struct
Problems:	Homographs: Heteronyms → It was about time to present the present. Homonyms → can, flat, rose



Steps of Linguistic Indexing

- Free text
  - → Morphology
    - → **Syntax**
      - → Semantics
        - → Pragmatics

Adjektive – noun (*Adjektiv-Substantiv*):	red carpet
Genitiv relation:	House *of* Cards, Victoria*'s* Secret
Coordinational relation:	Peter, Paul, *and* Mary
Prepositions:	*at* the door, *in* the house, *on* the roof

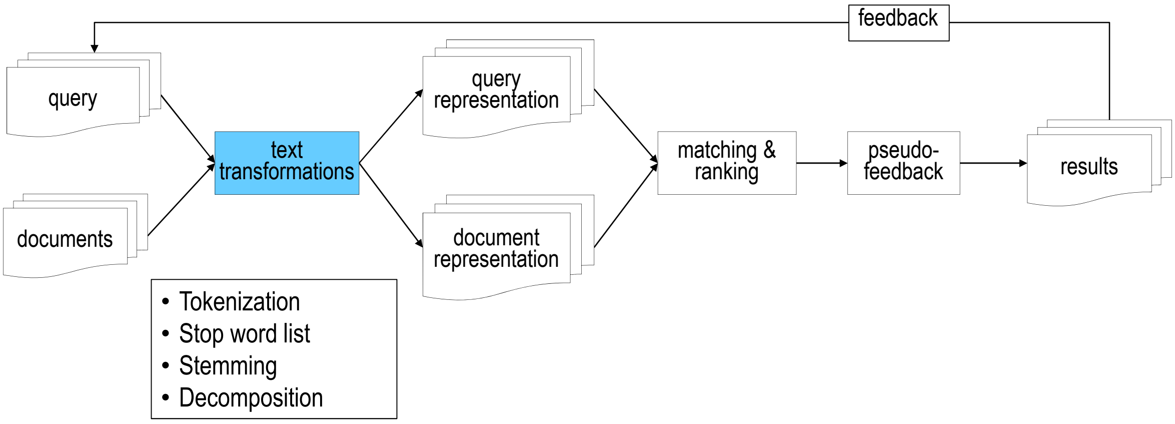
Steps of Linguistic Indexing

- Free text
  - →Morphology
    - →Syntax
      - →Semantics
        - →Pragmatics

Knowledge representation: Thesaurus, Classifikation

Problem: Polysemie

Transformations in Text Retrieval



- Tokenization
- Stop word list
- Stemming
- Decomposition

Tokenization: Concepts

- Word: delimited string of characters as can be found in texts
- Term: normalized word (upper / lower case, ending, …)
- Token: instance of a word / term in a text
- Type: class of a token

- Example: „To be or not to be“
  - Words: be, be, not, or, To, to
  - Terms: be, be, not, or, to, to
  - Token: words / terms
  - Type: be, not, or, to

Stemming (*Stammformreduktion*)

1. *dictionary-based stemmers*
2. *n-gram stemmers*
3. *affix stemmers*

} algorithmic stemmers

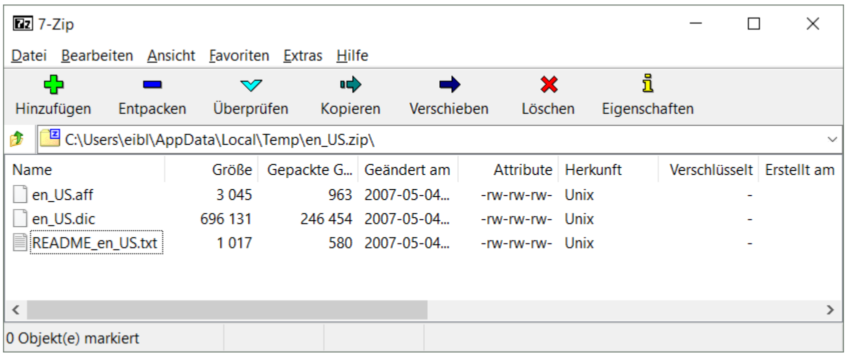
Stemming: affix

Definition by Merriam Webster: a letter or group of letters added to the beginning or end of a word to change its meaning.

a. prefix (*un*happy)
b. infix (German: ein*ge*schoben, ein*zu*schieben)
c. suffix (*suffix stripping*):
   - *a*-suffix, *attached* suffix (mandar*gli* = mandare + ***gli*** = to send + to him)
   - *i*-suffix, or *inflectional* suffix (fit + ***ed*** -> fitted (double ***t***))
   - *d*-suffix, or *derivational* suffix (medico + ***astro*** = medicastro = quack doctor, in English: -ness in some adjectives to get a noun)

Stemming I: dictionary-based stemmer

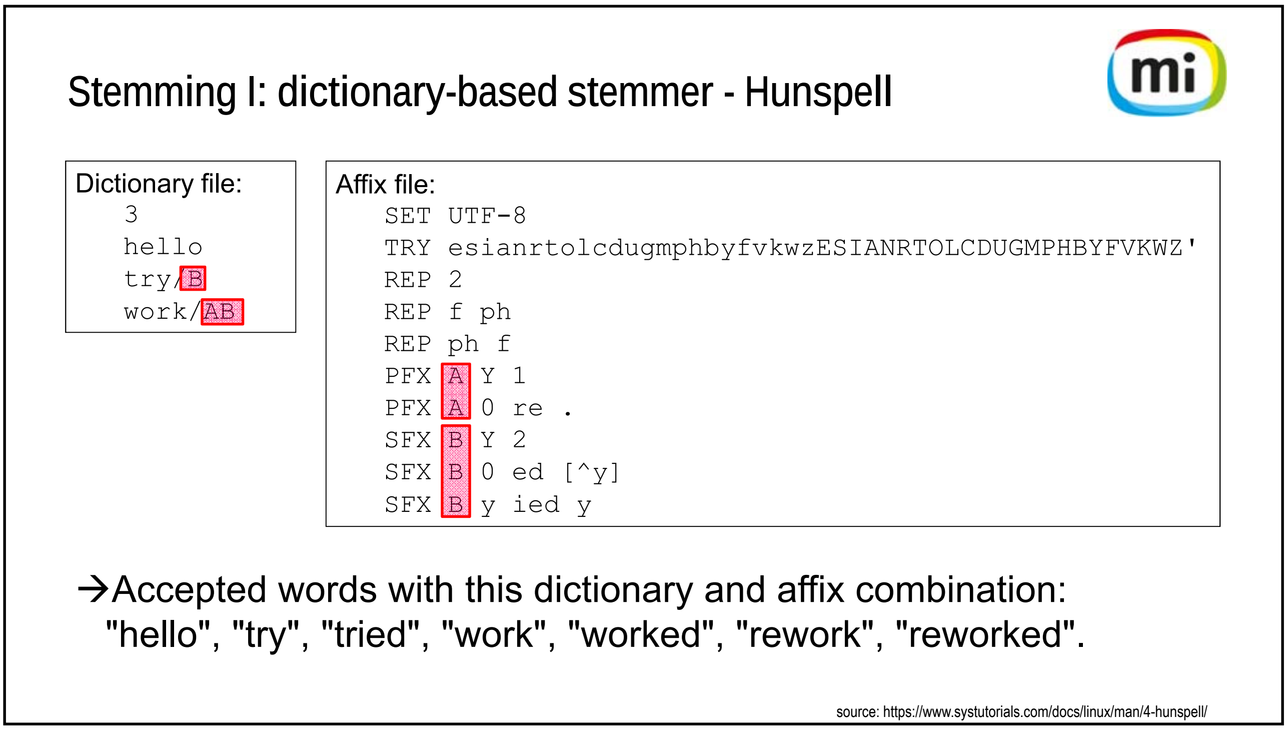
Hunspell Dictionary Stemmer


7-Zip
Datei Bearbeiten Ansicht Favoriten Extras Hilfe
Hinzufügen Entpacken Überprüfen Kopieren Verschieben Löschen Eigenschaften

C:\Users\eibl\AppData\Local\Temp\en_US.zip\

Name	Größe	Gepackte G...	Geändert am	Attribute	Herkunft	Verschlüsselt	Erstellt am
en_US.aff	3 045	963	2007-05-04...	-rw-rw-rw-	Unix	-	
en_US.dic	696 131	246 454	2007-05-04...	-rw-rw-rw-	Unix	-	
README_en_US.txt	1 017	580	2007-05-04...	-rw-rw-rw-	Unix	-	

0 Objekt(e) markiert



Stemming I: dictionary-based stemmer - Hunspell

Dictionary file:

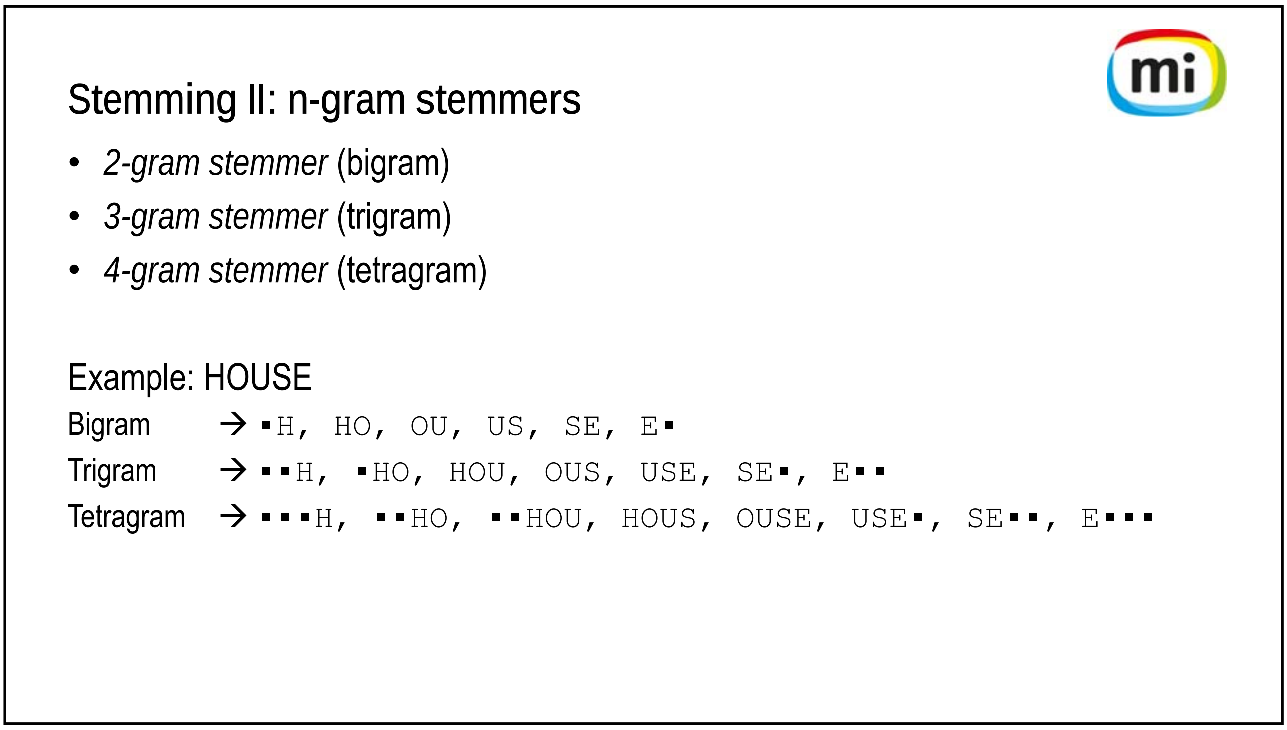
```
    3
    hello
    try/B
    work/AB
```

Affix file:

```
    SET UTF-8
    TRY esianrtolcdugmphbyfvkwzESIANRTOLCDUGMPHBYFVKWZ'
    REP 2
    REP f ph
    REP ph f
    PFX A Y 1
    PFX A 0 re .
    SFX B Y 2
    SFX B 0 ed [^y]
    SFX B y ied y
```

→Accepted words with this dictionary and affix combination: "hello", "try", "tried", "work", "worked", "rework", "reworked".

source: https://www.systutorials.com/docs/linux/man/4-hunspell/

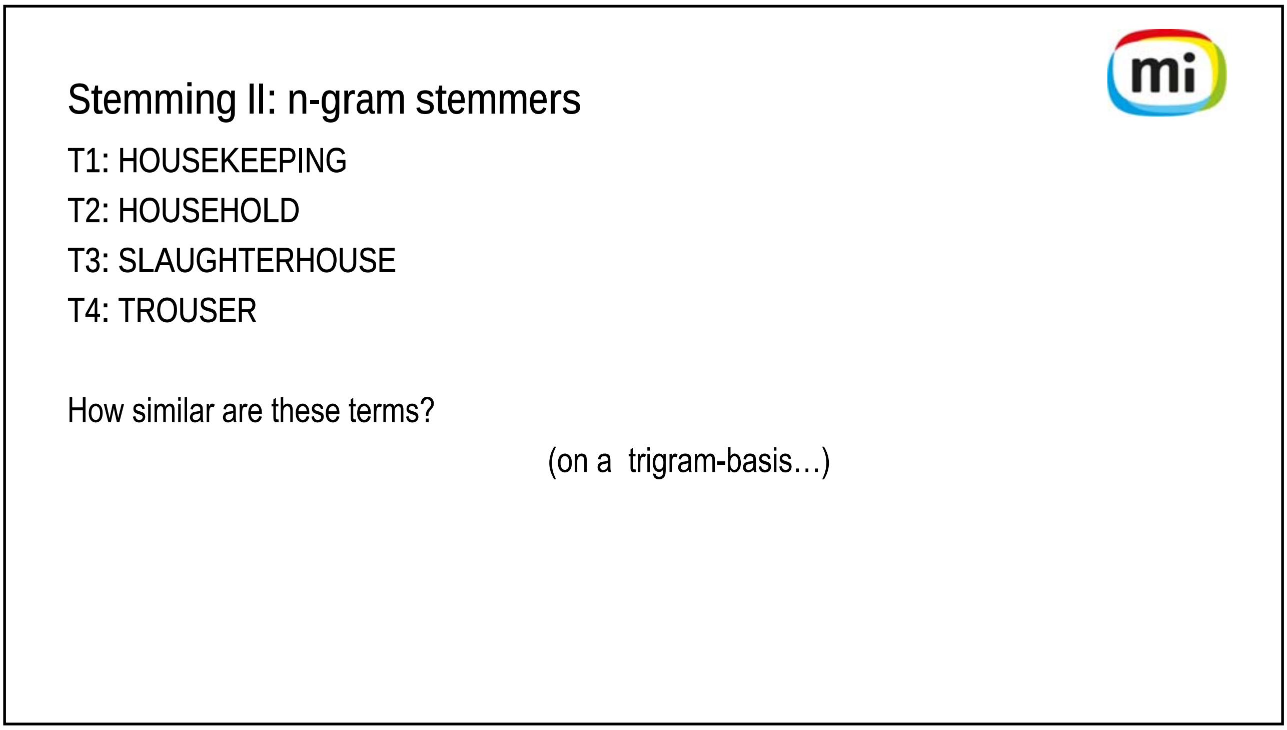


Stemming II: n-gram stemmers

- *2-gram stemmer* (bigram)
- *3-gram stemmer* (trigram)
- *4-gram stemmer* (tetragram)

Example: HOUSE

```
Bigram     → ▪H, HO, OU, US, SE, E▪
Trigram    → ▪▪H, ▪HO, HOU, OUS, USE, SE▪, E▪▪
Tetragram  → ▪▪▪H, ▪▪HO, ▪▪HOU, HOUS, OUSE, USE▪, SE▪▪, E▪▪▪
```

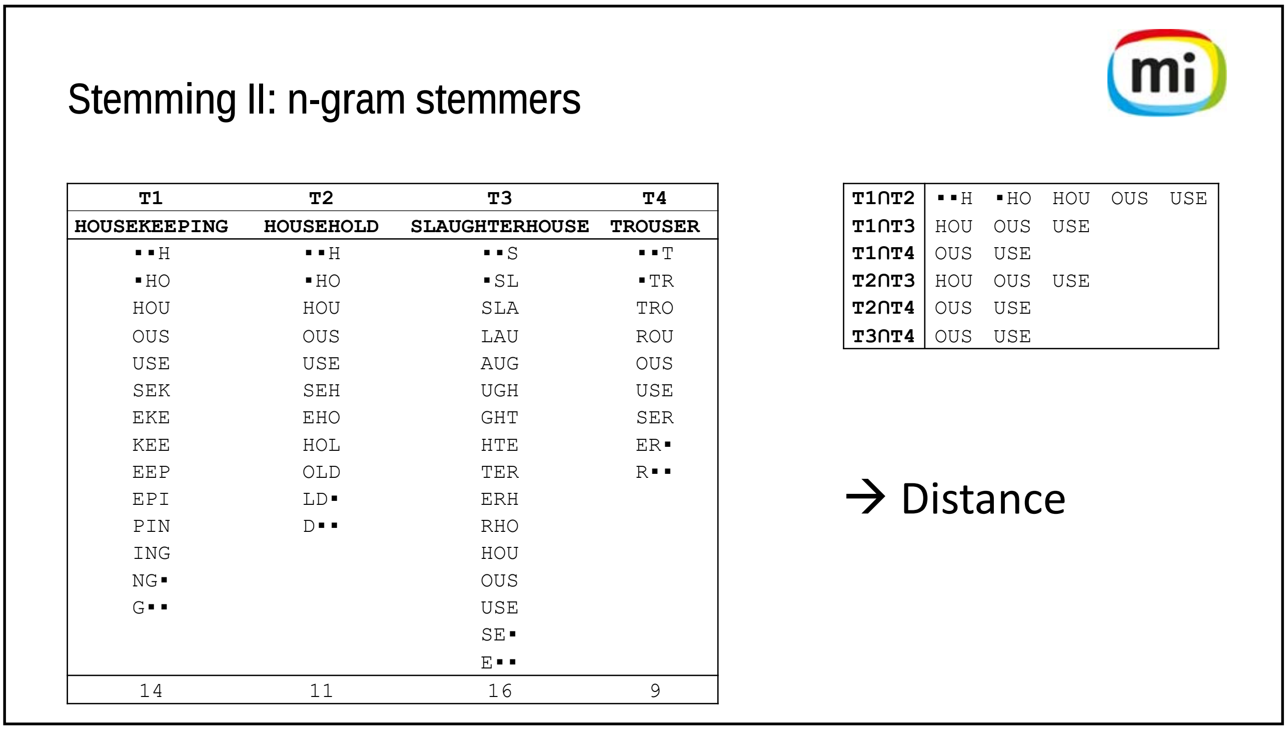


Stemming II: n-gram stemmers

T1: HOUSEKEEPING
T2: HOUSEHOLD
T3: SLAUGHTERHOUSE
T4: TROUSER

How similar are these terms?

(on a trigram-basis…)



Stemming II: n-gram stemmers

T1	T2	T3	T4
HOUSEKEEPING	HOUSEHOLD	SLAUGHTERHOUSE	TROUSER
▪▪H	▪▪H	▪▪S	▪▪T
▪HO	▪HO	▪SL	▪TR
HOU	HOU	SLA	TRO
OUS	OUS	LAU	ROU
USE	USE	AUG	OUS
SEK	SEH	UGH	USE
EKE	EHO	GHT	SER
KEE	HOL	HTE	ER▪
EEP	OLD	TER	R▪▪
EPI	LD▪	ERH	
PIN	D▪▪	RHO	
ING		HOU	
NG▪		OUS	
G▪▪		USE	
		SE▪	
		E▪▪	
14	11	16	9

T1∩T2	▪▪H	▪HO	HOU	OUS	USE
T1∩T3	HOU	OUS	USE		
T1∩T4	OUS	USE			
T2∩T3	HOU	OUS	USE		
T2∩T4	OUS	USE			
T3∩T4	OUS	USE			

→ Distance

Distance

Similarity measures: Dice, Jaccard, Cosinus, Ochiai, a.s.o.

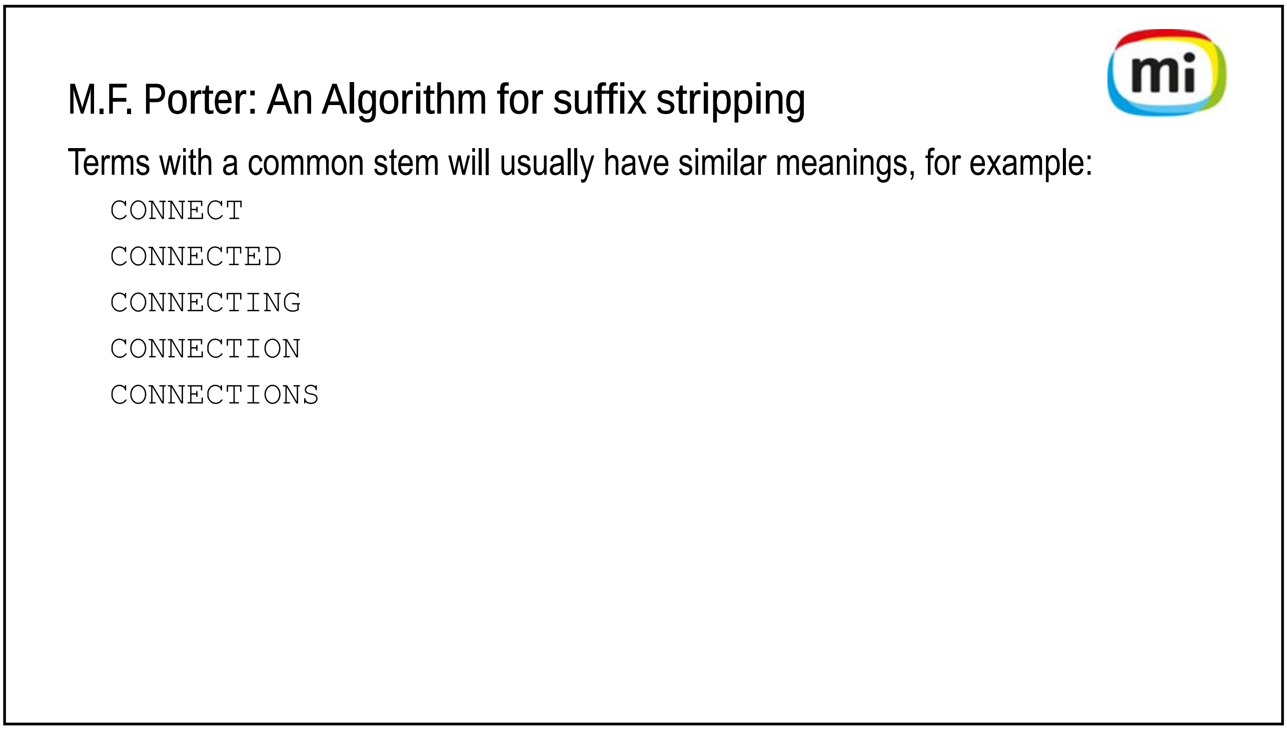
Two very simple measures:

						Count equivalent: **Tx∩Ty**	Count equivalent in relation to length: $\frac{\mathbf{Tx\cap Ty}}{\mathbf{Tx+Ty}}$
T1∩T2	▪▪H	▪HO	HOU	OUS	USE	5	5 / (14+11) = 0,2
T1∩T3	HOU	OUS	USE			3	3 / (14+16) = 0,1
T1∩T4	OUS	USE				2	2 / (14+9) = 0,08

Stemming III: affix stemmers

classical affix-stemmers:

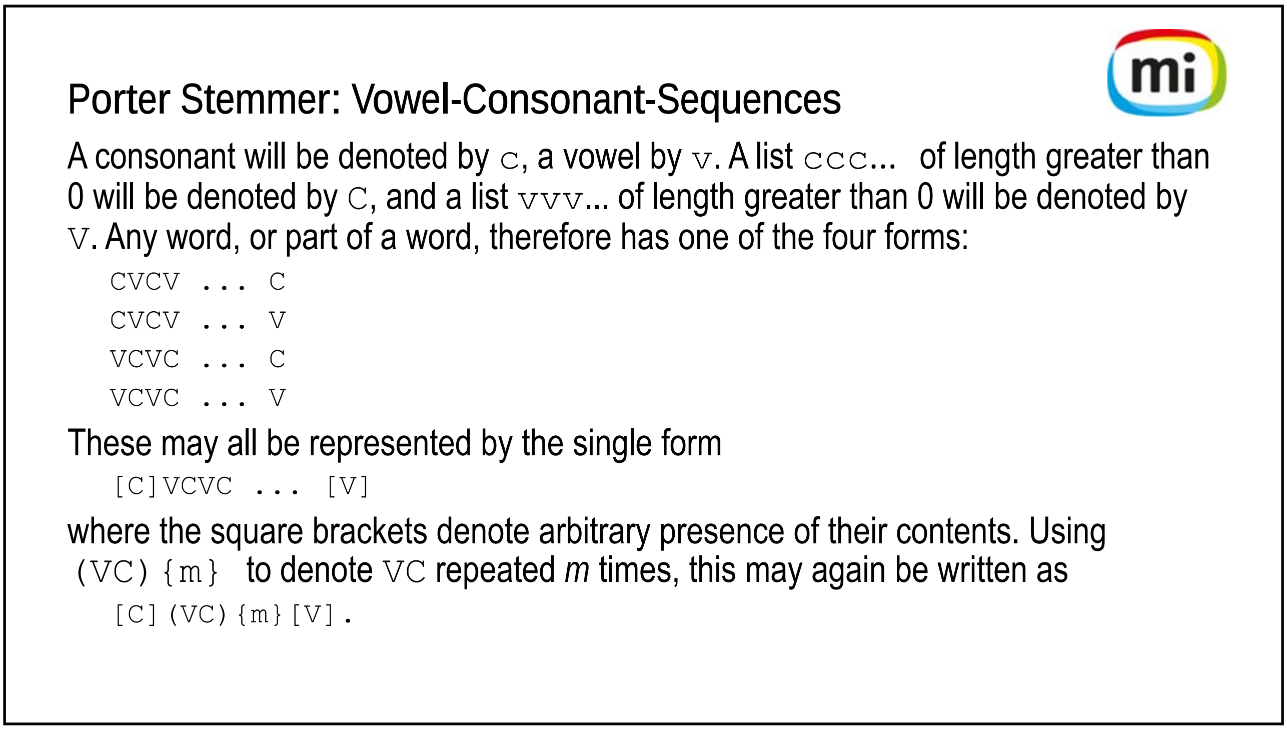
- Porter Stemmer
- Snowball Stemmer
- Krovetz (Kstem)



M.F. Porter: An Algorithm for suffix stripping

Terms with a common stem will usually have similar meanings, for example:

```
CONNECT
CONNECTED
CONNECTING
CONNECTION
CONNECTIONS
```



Porter Stemmer: Vowel-Consonant-Sequences

A consonant will be denoted by `c`, a vowel by `v`. A list `ccc`... of length greater than 0 will be denoted by `C`, and a list `vvv`... of length greater than 0 will be denoted by `V`. Any word, or part of a word, therefore has one of the four forms:

```
CVCV ... C
CVCV ... V
VCVC ... C
VCVC ... V
```

These may all be represented by the single form

```
[C]VCVC ... [V]
```

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].
```

Porter Stemmer: VC - Vowel Consonant Sequences

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:

```
m=0 TR, EE, TREE, Y, BY.
m=1 TROUBLE, OATS, TREES, IVY.
m=2 TROUBLES, PRIVATE,    OATEN,  ORRERY.
    ccvvccvc  ccvcvcv     vvcvc   vccvcv
    CVCVC     CVCVCV      VCVC    VCVCV
    [C]VCVC   [C]VCVC[V]  VCVC    VCVCV
```

Porter Stemmer: *suffix removal*

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

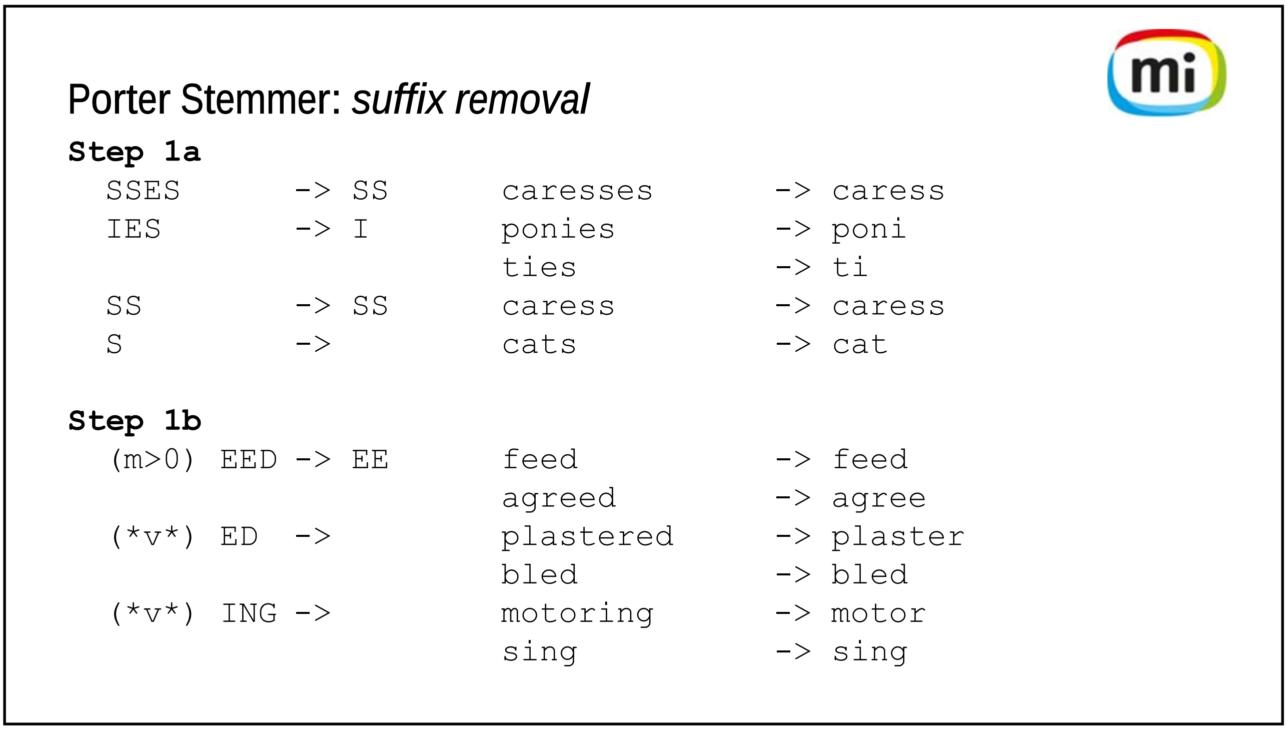
```
(condition) S1 -> S2
```

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 ->
```

Here S1 is `EMENT' and S2 is null. This would map REPLACEMENT to REPLAC, since REPLAC is a word part for which m = 2. 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).
```



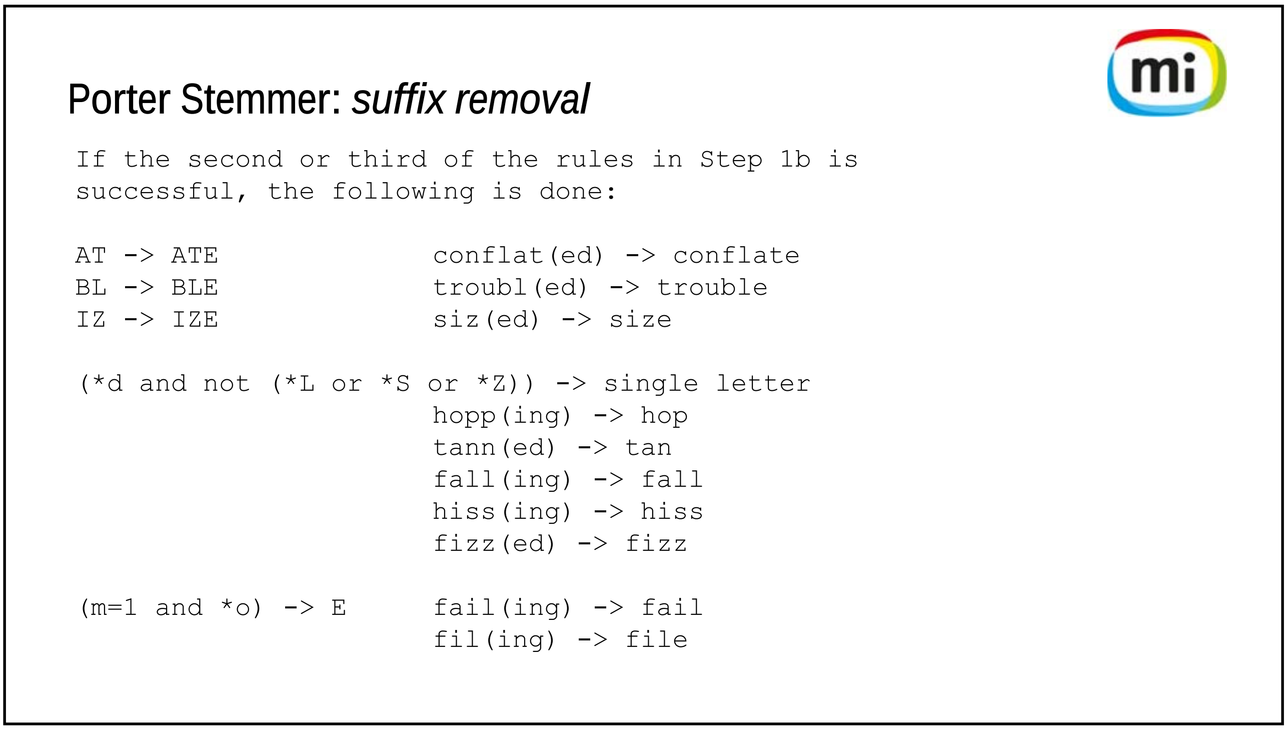
Porter Stemmer: *suffix removal*

Step 1a

```
  SSES      -> SS      caresses      -> caress
  IES       -> I       ponies        -> poni
                       ties          -> ti
  SS        -> SS      caress        -> caress
  S         ->         cats          -> cat
```

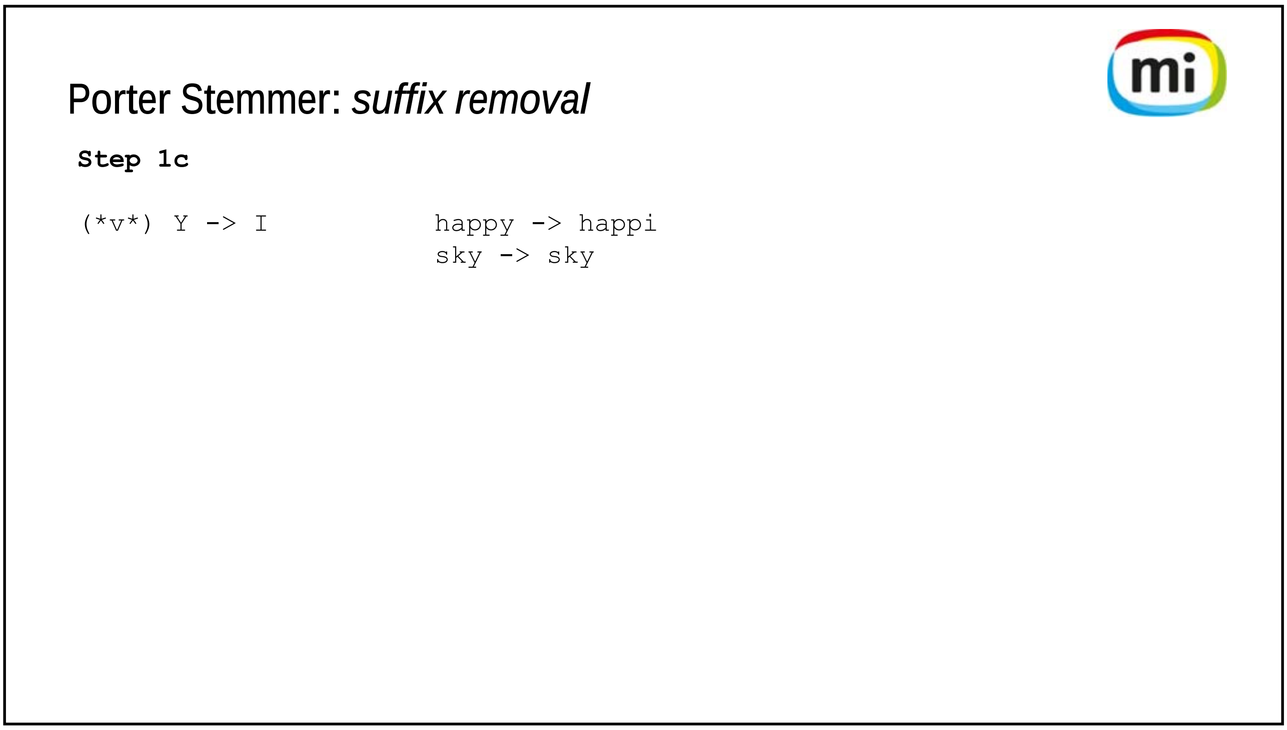
Step 1b

```
  (m>0) EED -> EE      feed          -> feed
                       agreed        -> agree
  (*v*) ED  ->         plastered     -> plaster
                       bled          -> bled
  (*v*) ING ->         motoring      -> motor
                       sing          -> sing
```



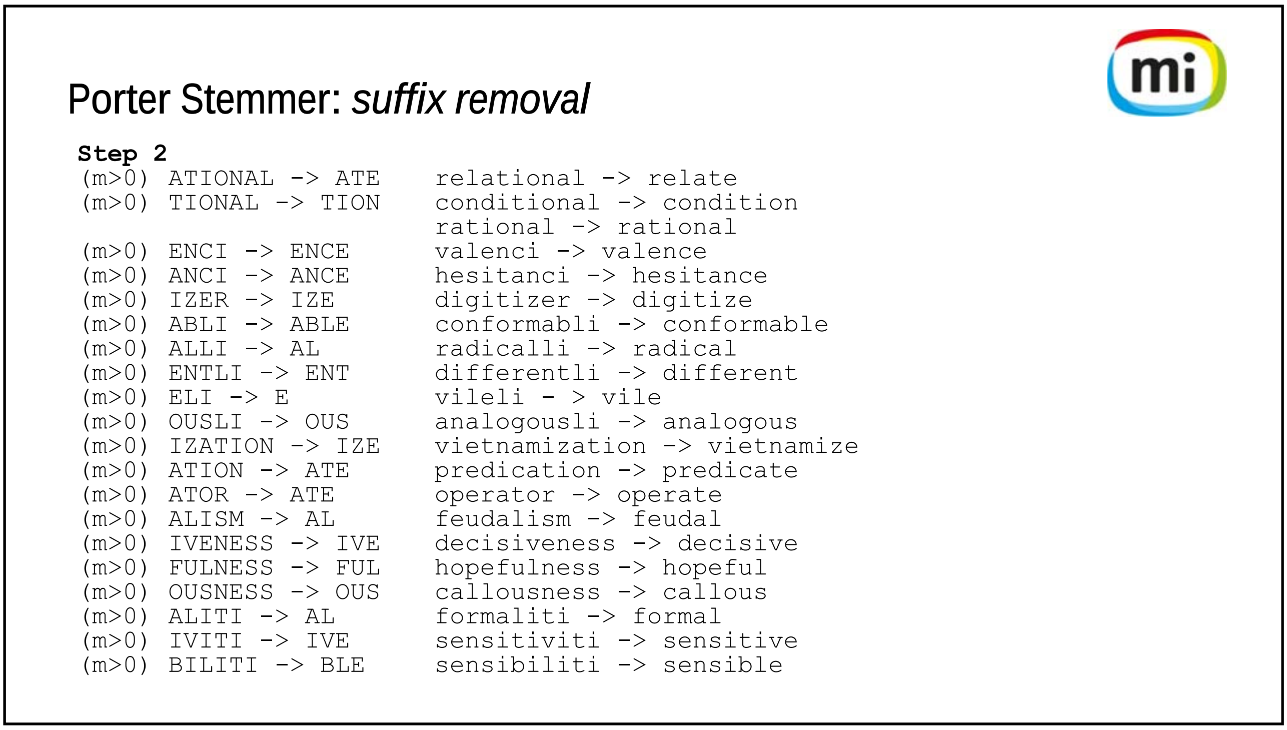
Porter Stemmer: *suffix removal*

```
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

(*d and not (*L or *S or *Z)) -> single letter
                         hopp(ing)  -> hop
                         tann(ed)  -> tan
                         fall(ing)  -> fall
                         hiss(ing)  -> hiss
                         fizz(ed)  -> fizz

(m=1 and *o) -> E        fail(ing)  -> fail
                         fil(ing)  -> file
```



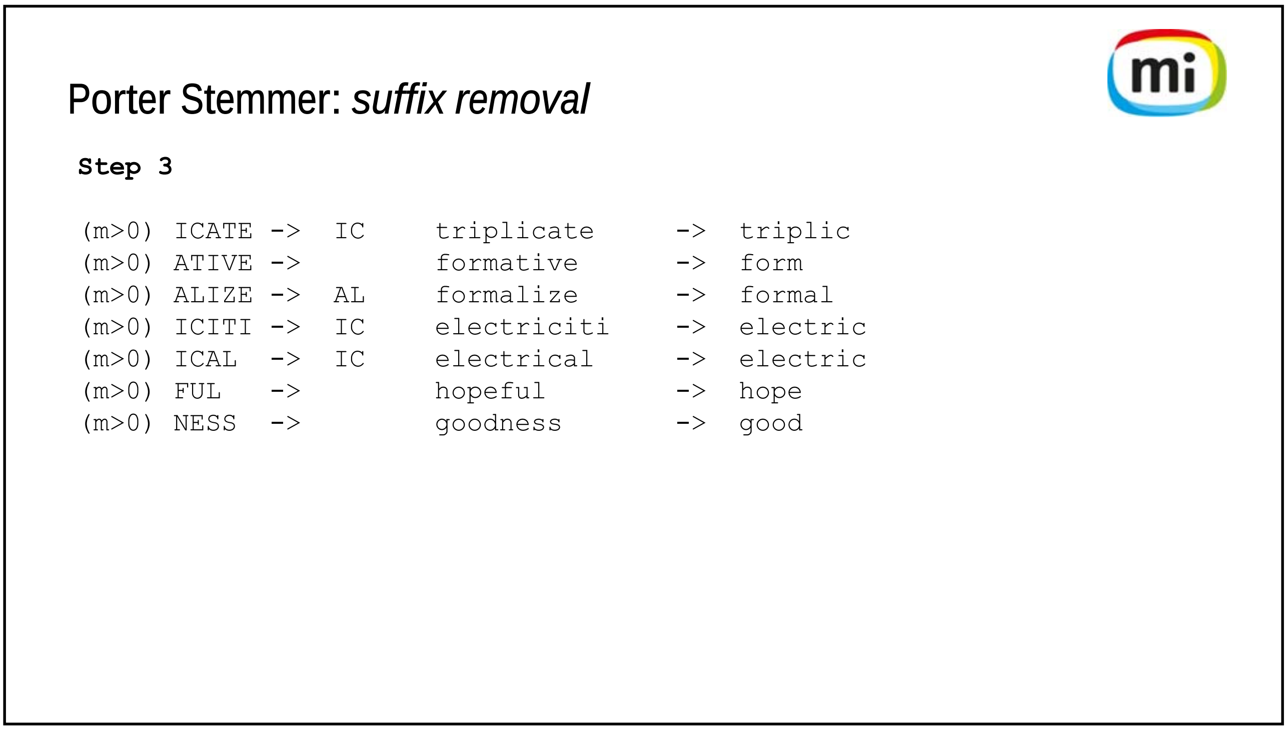
Porter Stemmer: *suffix removal*

```
Step 1c

(*v*) Y -> I           happy -> happi
                       sky -> sky
```



Porter Stemmer: *suffix removal*

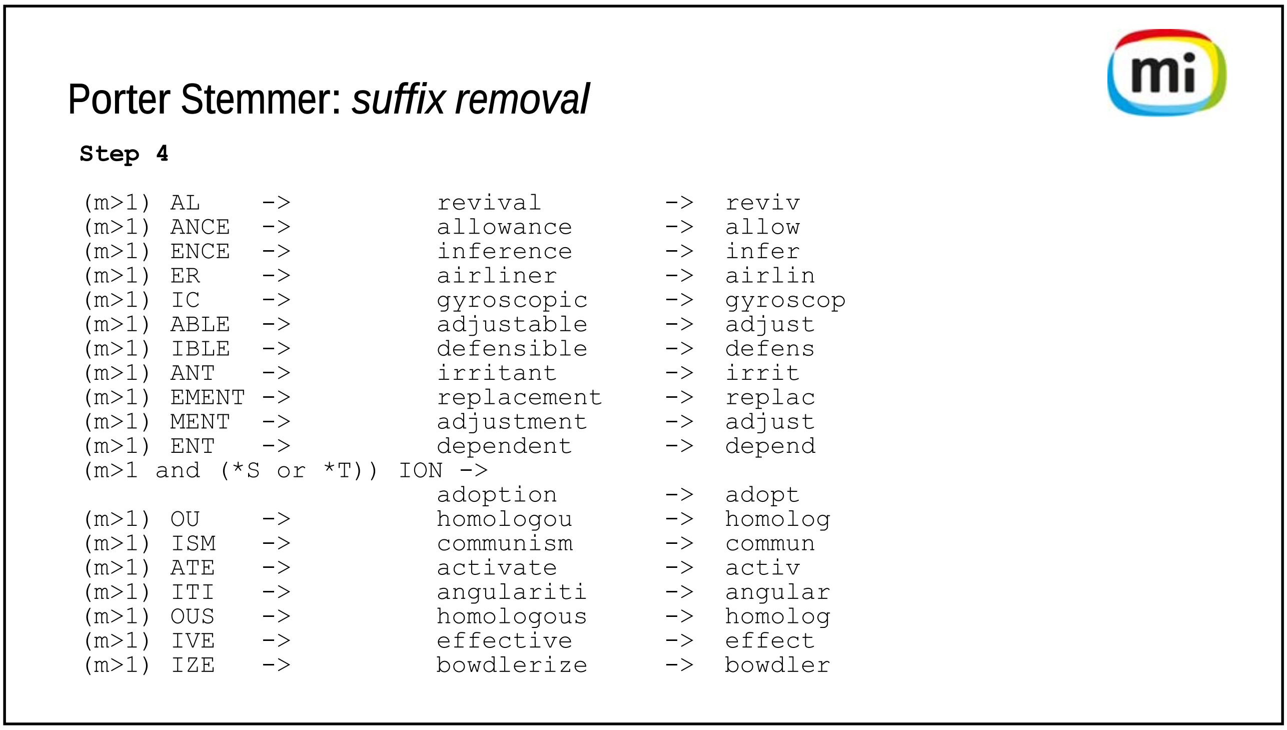
```
Step 2
(m>0) ATIONAL -> ATE   relational -> relate
(m>0) TIONAL -> TION   conditional -> condition
                       rational -> rational
(m>0) ENCI -> ENCE     valenci -> valence
(m>0) ANCI -> ANCE     hesitanci -> hesitance
(m>0) IZER -> IZE      digitizer -> digitize
(m>0) ABLI -> ABLE     conformabli -> conformable
(m>0) ALLI -> AL       radicalli -> radical
(m>0) ENTLI -> ENT     differentli -> different
(m>0) ELI -> E         vileli - > vile
(m>0) OUSLI -> OUS     analogousli -> analogous
(m>0) IZATION -> IZE   vietnamization -> vietnamize
(m>0) ATION -> ATE     predication -> predicate
(m>0) ATOR -> ATE      operator -> operate
(m>0) ALISM -> AL      feudalism -> feudal
(m>0) IVENESS -> IVE   decisiveness -> decisive
(m>0) FULNESS -> FUL   hopefulness -> hopeful
(m>0) OUSNESS -> OUS   callousness -> callous
(m>0) ALITI -> AL      formaliti -> formal
(m>0) IVITI -> IVE     sensitiviti -> sensitive
(m>0) BILITI -> BLE    sensibiliti -> sensible
```



Porter Stemmer: *suffix removal*

Step 3

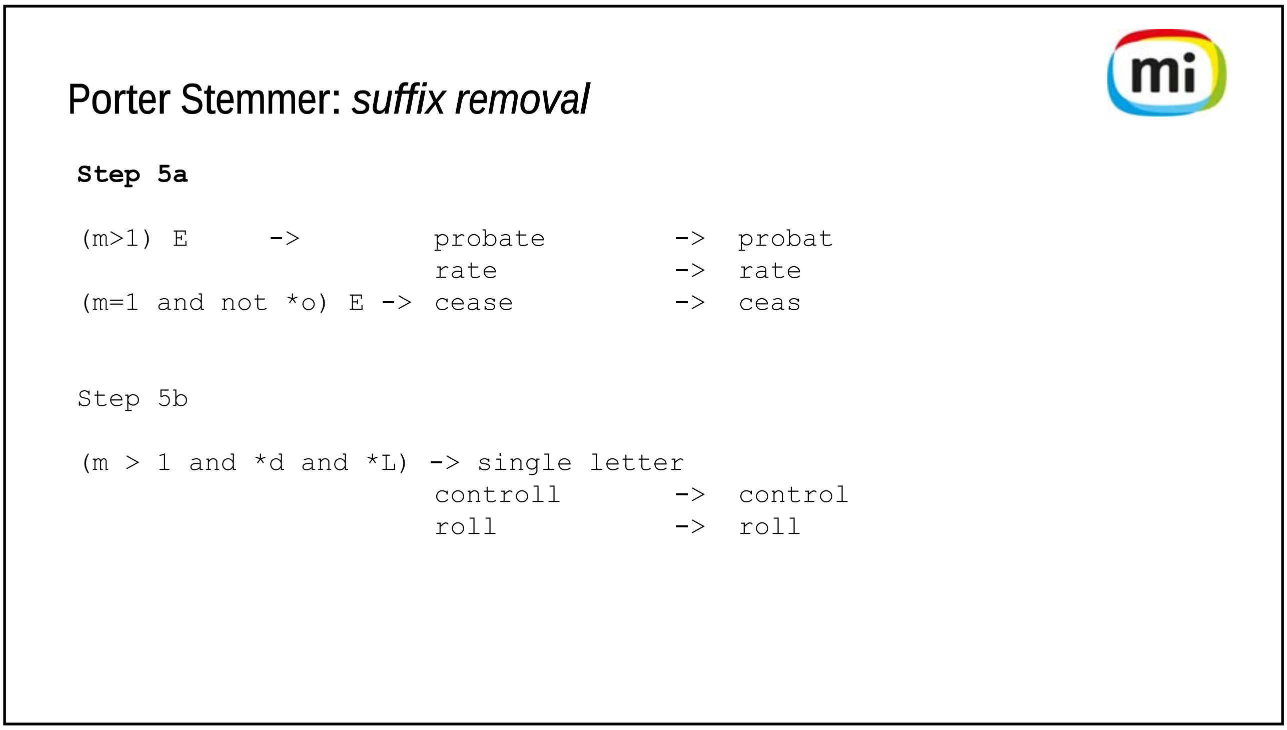
```
(m>0) ICATE ->  IC     triplicate     ->  triplic
(m>0) ATIVE ->         formative      ->  form
(m>0) ALIZE ->  AL     formalize      ->  formal
(m>0) ICITI ->  IC     electriciti    ->  electric
(m>0) ICAL  ->  IC     electrical     ->  electric
(m>0) FUL   ->         hopeful        ->  hope
(m>0) NESS  ->         goodness       ->  good
```



Porter Stemmer: *suffix removal*

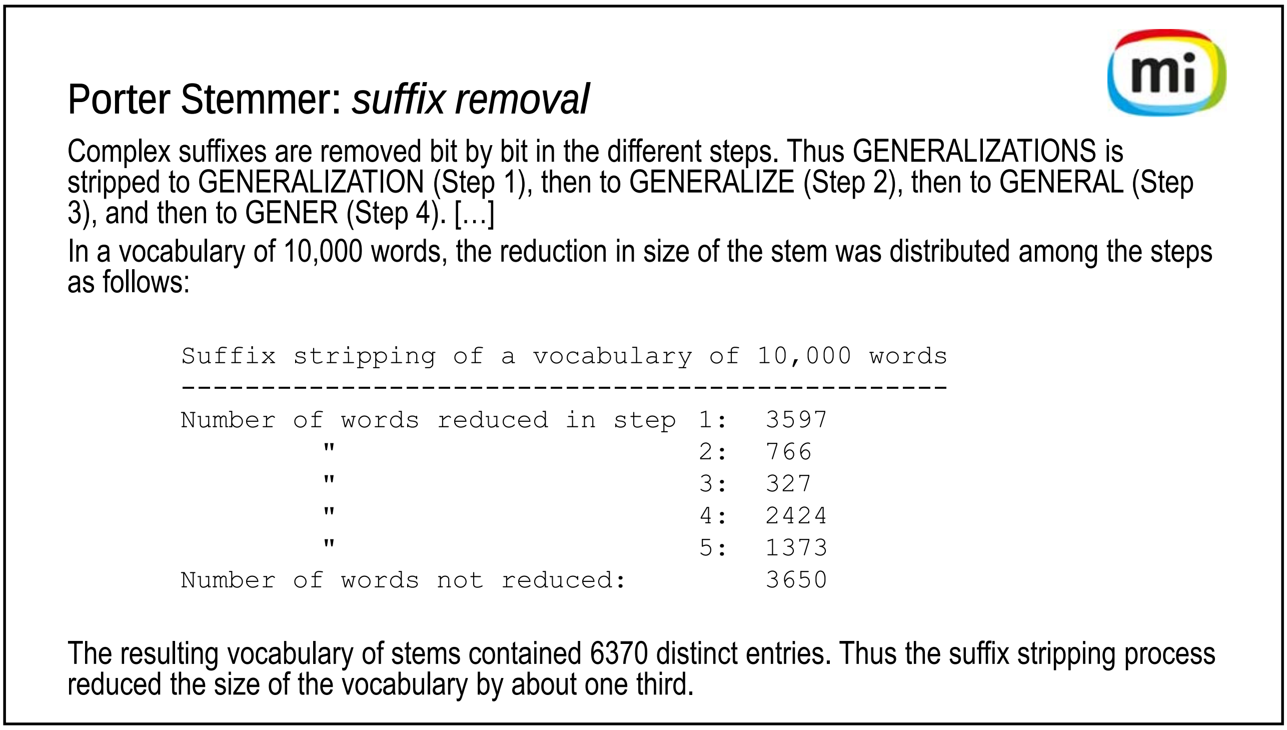
Step 4

```
(m>1) AL    ->          revival        ->  reviv
(m>1) ANCE  ->          allowance      ->  allow
(m>1) ENCE  ->          inference      ->  infer
(m>1) ER    ->          airliner       ->  airlin
(m>1) IC    ->          gyroscopic     ->  gyroscop
(m>1) ABLE  ->          adjustable     ->  adjust
(m>1) IBLE  ->          defensible     ->  defens
(m>1) ANT   ->          irritant       ->  irrit
(m>1) EMENT ->          replacement    ->  replac
(m>1) MENT  ->          adjustment     ->  adjust
(m>1) ENT   ->          dependent      ->  depend
(m>1 and (*S or *T)) ION ->
                        adoption       ->  adopt
(m>1) OU    ->          homologou      ->  homolog
(m>1) ISM   ->          communism      ->  commun
(m>1) ATE   ->          activate       ->  activ
(m>1) ITI   ->          angulariti     ->  angular
(m>1) OUS   ->          homologous     ->  homolog
(m>1) IVE   ->          effective      ->  effect
(m>1) IZE   ->          bowdlerize     ->  bowdler
```



Porter Stemmer: *suffix removal*

```
Step 5a

(m>1) E      ->         probate         ->  probat
                        rate            ->  rate
(m=1 and not *o) E ->  cease           ->  ceas


Step 5b

(m > 1 and *d and *L) -> single letter
                        controll        ->  control
                        roll            ->  roll
```



Porter Stemmer: *suffix removal*

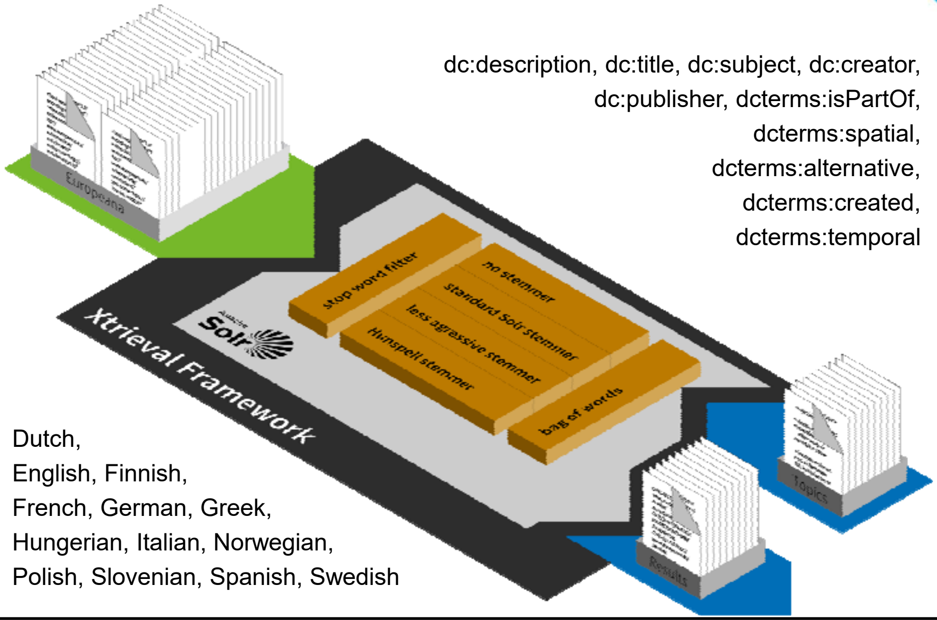
Complex suffixes are removed bit by bit in the different steps. Thus GENERALIZATIONS is stripped to GENERALIZATION (Step 1), then to GENERALIZE (Step 2), then to GENERAL (Step 3), and then to GENER (Step 4). […]

In a vocabulary of 10,000 words, the reduction in size of the stem was distributed among the steps as follows:

```
Suffix stripping of a vocabulary of 10,000 words
------------------------------------------------
Number of words reduced in step 1:   3597
         "                      2:   766
         "                      3:   327
         "                      4:   2424
         "                      5:   1373
Number of words not reduced:         3650
```

The resulting vocabulary of stems contained 6370 distinct entries. Thus the suffix stripping process reduced the size of the vocabulary by about one third.

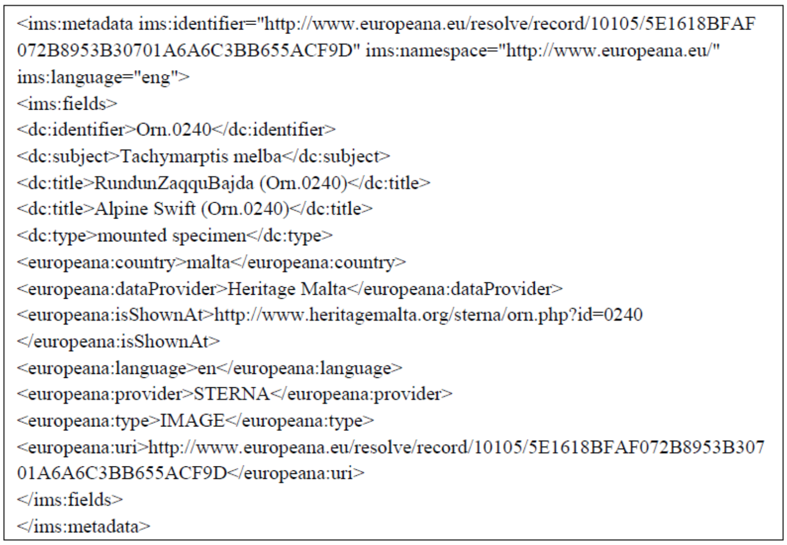
Example: Cultural Heritage



dc:description, dc:title, dc:subject, dc:creator,
dc:publisher, dcterms:isPartOf,
dcterms:spatial,
dcterms:alternative,
dcterms:created,
dcterms:temporal

Dutch,
English, Finnish,
French, German, Greek,
Hungerian, Italian, Norwegian,
Polish, Slovenian, Spanish, Swedish

Example Cultural Heritage



```
<ims:metadata ims:identifier="http://www.europeana.eu/resolve/record/10105/5E1618BFAF
072B8953B30701A6A6C3BB655ACF9D" ims:namespace="http://www.europeana.eu/"
ims:language="eng">
<ims:fields>
<dc:identifier>Orn.0240</dc:identifier>
<dc:subject>Tachymarptis melba</dc:subject>
<dc:title>RundunZaqquBajda (Orn.0240)</dc:title>
<dc:title>Alpine Swift (Orn.0240)</dc:title>
<dc:type>mounted specimen</dc:type>
<europeana:country>malta</europeana:country>
<europeana:dataProvider>Heritage Malta</europeana:dataProvider>
<europeana:isShownAt>http://www.heritagemalta.org/sterna/orn.php?id=0240
</europeana:isShownAt>
<europeana:language>en</europeana:language>
<europeana:provider>STERNA</europeana:provider>
<europeana:type>IMAGE</europeana:type>
<europeana:uri>http://www.europeana.eu/resolve/record/10105/5E1618BFAF072B8953B307
01A6A6C3BB655ACF9D</europeana:uri>
</ims:fields>
</ims:metadata>
```

Fig.1. Europeana CHiC Collection Sample Record

Vivien Petras, Toine Bogers, Nicola Ferro and Ivano Masiero (2013). Cultural Heritage in CLEF (CHiC) 2013 – Multilingual Task Overview: http://ceur-ws.org/Vol-1179/CLEF2013wn-CHiC-PetrasEt2013.pdf

Alpine Swift (*Rundun Zaqqu Bajda*)

Example Cultural Heritage

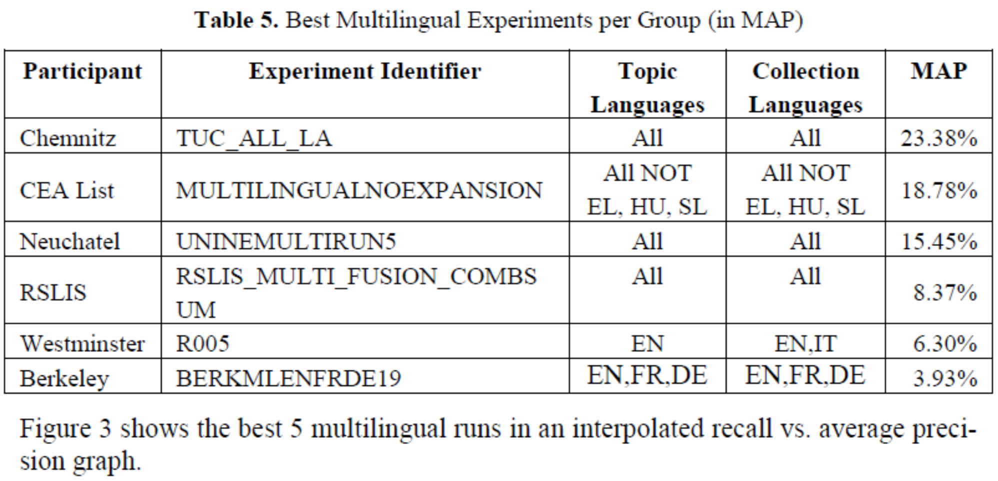


Table 5. Best Multilingual Experiments per Group (in MAP)

Participant	Experiment Identifier	Topic Languages	Collection Languages	MAP
Chemnitz	TUC_ALL_LA	All	All	23.38%
CEA List	MULTILINGUALNOEXPANSION	All NOT EL, HU, SL	All NOT EL, HU, SL	18.78%
Neuchatel	UNINEMULTIRUN5	All	All	15.45%
RSLIS	RSLIS_MULTI_FUSION_COMBSUM	All	All	8.37%
Westminster	R005	EN	EN,IT	6.30%
Berkeley	BERKMLENFRDE19	EN,FR,DE	EN,FR,DE	3.93%

Figure 3 shows the best 5 multilingual runs in an interpolated recall vs. average precision graph.

Vivien Petras, Toine Bogers, Nicola Ferro and Ivano Masiero (2013). Cultural Heritage in CLEF (CHiC) 2013 – Multilingual Task Overview: http://ims-sites.dei.unipd.it/documents/71612/430938/CLEF2013wn-CHiC-PetrasEt2013.pdf

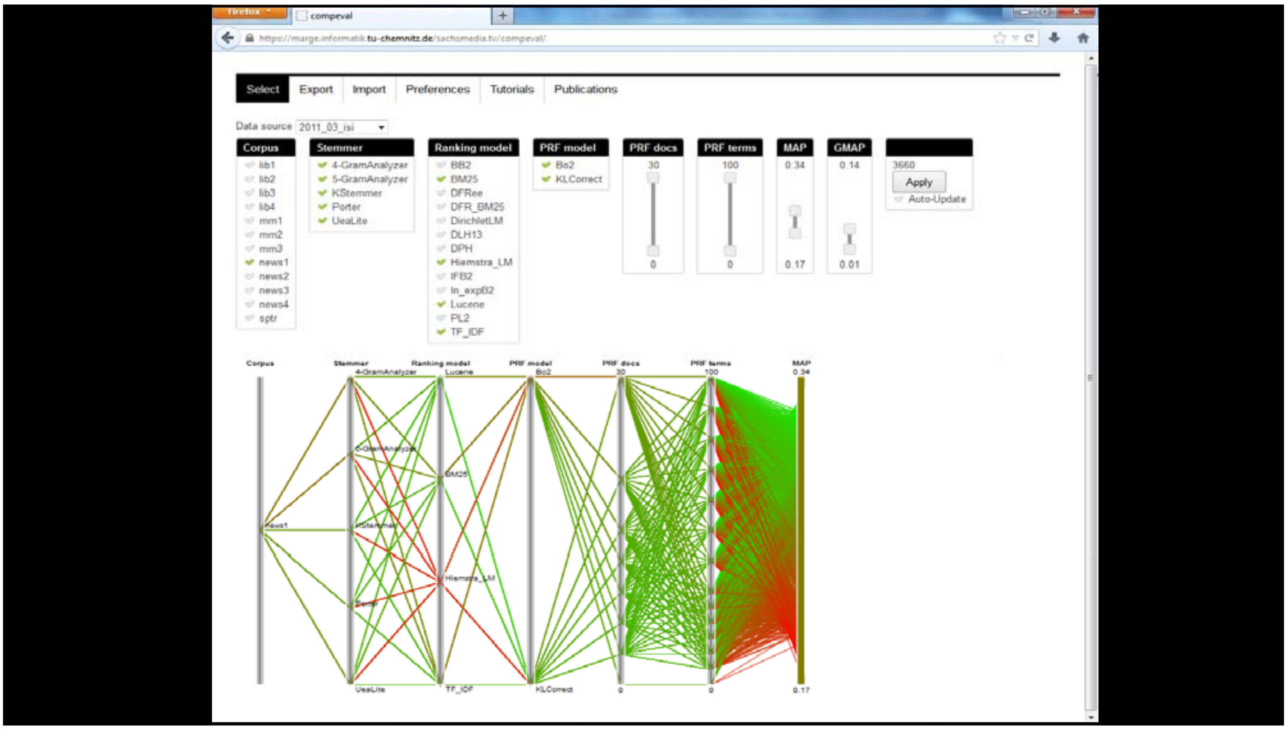
Example Cultural Heritage

Stemmer	MAP	GMAP	BPref	R-Precision
Solr standard	0.2583	0.1603	0.3538	0.3329
less aggressive	0.2590	0.1552	0.3686	0.3253
Hunspell	0.2466	0.1314	0.2914	0.3160
no stemmer	0.2684	0.1587	0.3031	0.3444
Snowball	0.2604	0.1591	0.3576	0.3360
no stop words extraction	0.1597	0.0621	0.2251	0.2297

The higher the result the better (on a 0 to 1 range)

→ Obviously hughe impact of stop word list: Why?

→ Obviously negative impact of stemmer: Why?



Data source 2011_03_isi

Corpus
- ☐ lib1
- ☐ lib2
- ☐ lib3
- ☐ lib4
- ☐ mm1
- ☐ mm2
- ☐ mm3
- ☑ news1
- ☐ news2
- ☐ news3
- ☐ news4
- ☐ sptr

Stemmer
- ☑ 4-GramAnalyzer
- ☑ 5-GramAnalyzer
- ☑ KStemmer
- ☑ Porter
- ☑ UeaLite

Ranking model
- ☐ BB2
- ☑ BM25
- ☐ DFRee
- ☐ DFR_BM25
- ☐ DirichletLM
- ☐ DLH13
- ☐ DPH
- ☑ Hiemstra_LM
- ☐ IFB2
- ☐ In_expB2
- ☑ Lucene
- ☐ PL2
- ☑ TF_IDF

PRF model
- ☑ Bo2
- ☑ KLCorrect

PRF docs 30 – 0

PRF terms 100 – 0

MAP 0.34 – 0.17

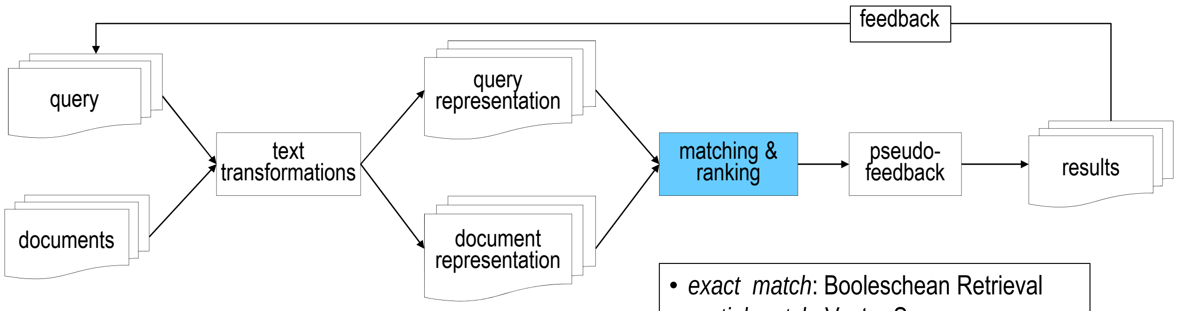
GMAP 0.14 – 0.01

3660

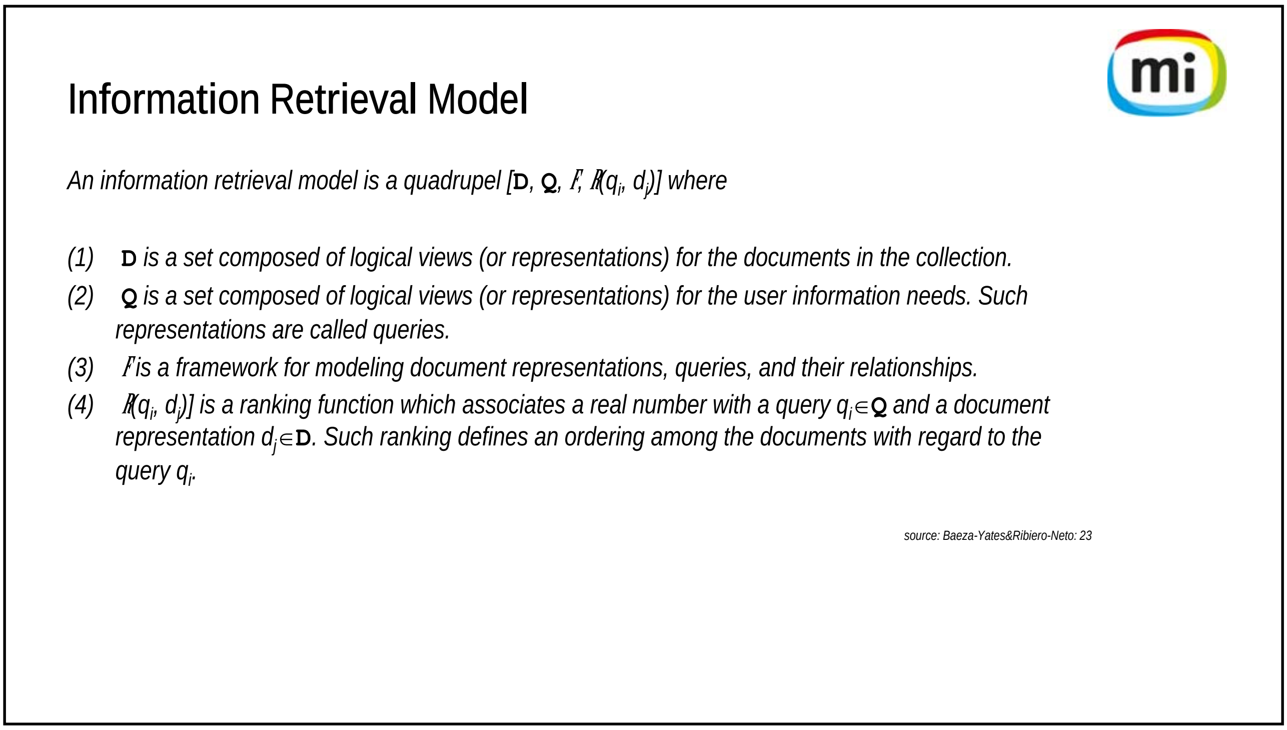
Apply

☐ Auto-Update

Matching & Ranking



- *exact match*: Booleschean Retrieval
- *partial match*: Vector Space, Probabilistic, Fuzzy, Clustering, a.s.o.



Information Retrieval Model

An information retrieval model is a quadrupel $[\mathbf{D}, \mathbf{Q}, \mathcal{F}, \mathcal{R}(q_i, d_j)]$ *where*

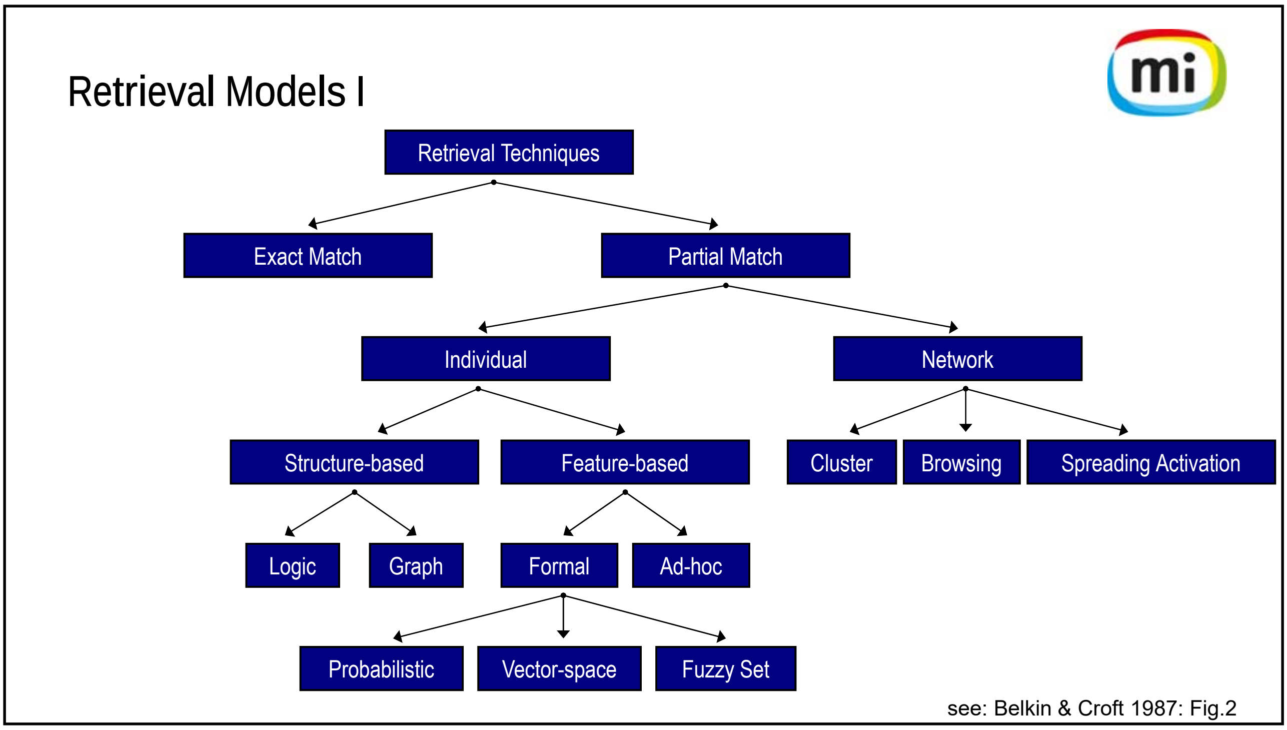
(1) $\mathbf{D}$ *is a set composed of logical views (or representations) for the documents in the collection.*

(2) $\mathbf{Q}$ *is a set composed of logical views (or representations) for the user information needs. Such representations are called queries.*

(3) $\mathcal{F}$ *is a framework for modeling document representations, queries, and their relationships.*

(4) $\mathcal{R}(q_i, d_j)]$ *is a ranking function which associates a real number with a query* $q_i \in \mathbf{Q}$ *and a document representation* $d_j \in \mathbf{D}$*. Such ranking defines an ordering among the documents with regard to the query* q_i*.*

source: Baeza-Yates&Ribiero-Neto: 23



Retrieval Models I

Retrieval Techniques

Exact Match

Partial Match

Individual

Network

Structure-based

Feature-based

Cluster

Browsing

Spreading Activation

Logic

Graph

Formal

Ad-hoc

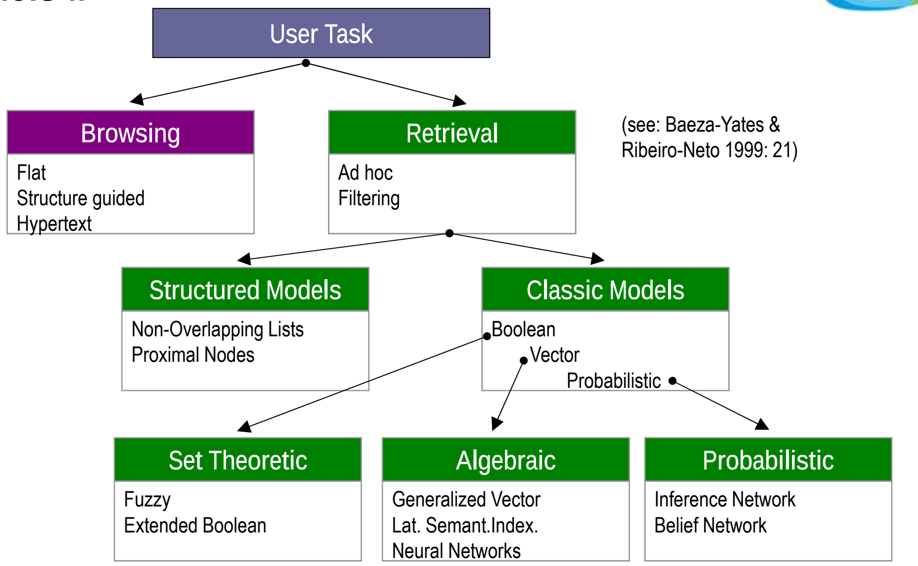
Probabilistic

Vector-space

Fuzzy Set

see: Belkin & Croft 1987: Fig.2

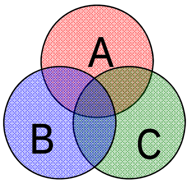
Retrieval Models II



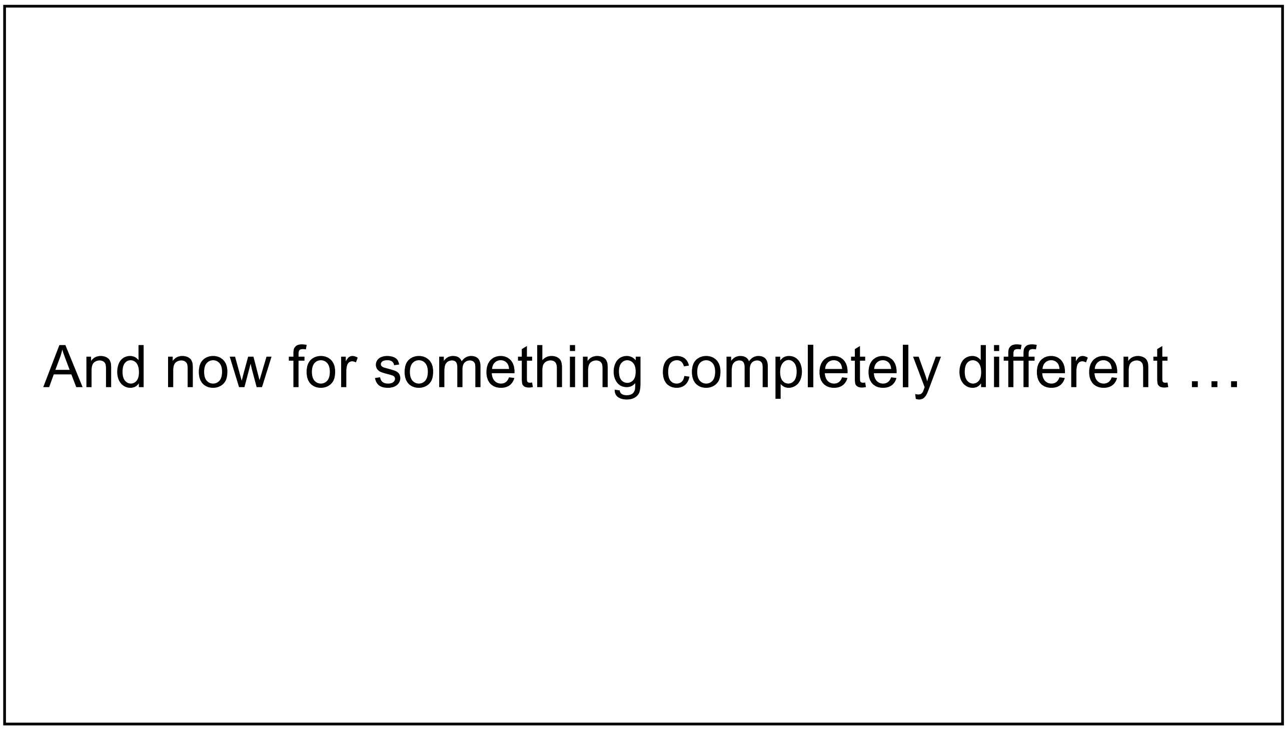
(see: Baeza-Yates & Ribeiro-Neto 1999: 21)

Boolean Retrieval

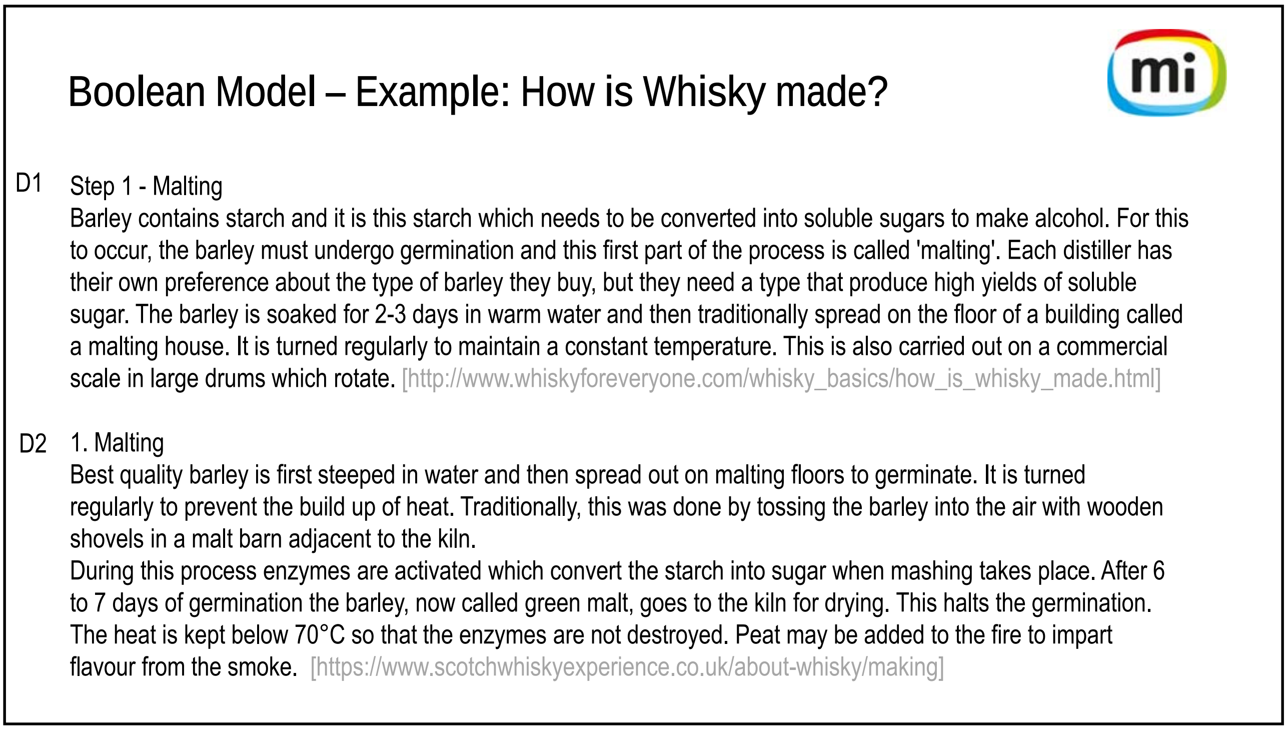
- Boolean Logic
  - Conjunction: AND ∧
  - Disjunction 1: OR ∨ (Adjunction, not excluding)
  - Disjunction 2: XOR (Contravalence, excluding)
  - Negation: NOT ¬
- Binary weighting: $w_{i,j} \in \{0,1\}$



A	B	A∧B	A∧¬B	¬A∧B	¬A∧¬B	A∨B	A∨¬B	¬A∨B	¬A∨¬B	AXORB	AXOR¬B	¬AXORB	¬AXOR¬B
w	w	w	f	f	f	w	w	w	f	f	w	w	f
w	f	f	w	f	f	w	w	f	w	w	f	f	w
f	w	f	f	w	f	w	f	w	w	w	f	f	w
f	f	f	f	f	w	f	w	w	w	f	w	w	f



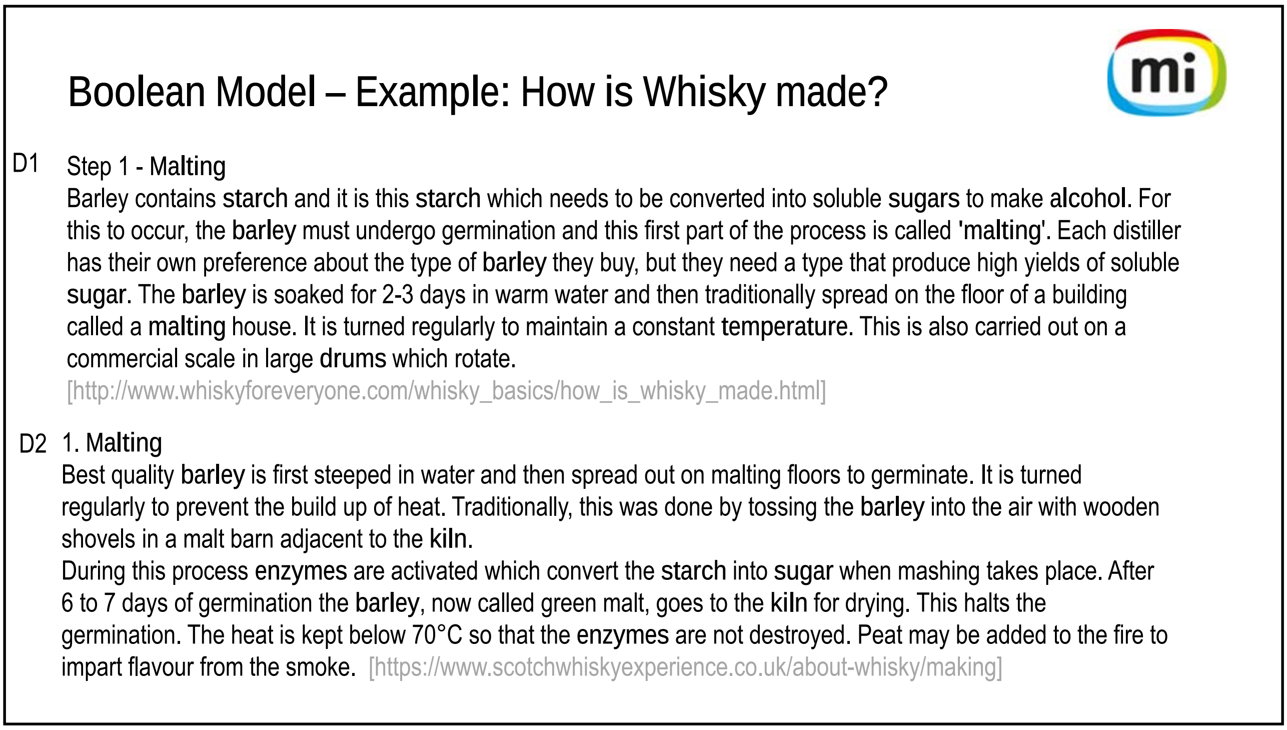
And now for something completely different …



Boolean Model – Example: How is Whisky made?

D1 Step 1 - Malting
Barley contains starch and it is this starch which needs to be converted into soluble sugars to make alcohol. For this to occur, the barley must undergo germination and this first part of the process is called 'malting'. Each distiller has their own preference about the type of barley they buy, but they need a type that produce high yields of soluble sugar. The barley is soaked for 2-3 days in warm water and then traditionally spread on the floor of a building called a malting house. It is turned regularly to maintain a constant temperature. This is also carried out on a commercial scale in large drums which rotate. [http://www.whiskyforeveryone.com/whisky_basics/how_is_whisky_made.html]

D2 1. Malting
Best quality barley is first steeped in water and then spread out on malting floors to germinate. It is turned regularly to prevent the build up of heat. Traditionally, this was done by tossing the barley into the air with wooden shovels in a malt barn adjacent to the kiln.
During this process enzymes are activated which convert the starch into sugar when mashing takes place. After 6 to 7 days of germination the barley, now called green malt, goes to the kiln for drying. This halts the germination. The heat is kept below 70°C so that the enzymes are not destroyed. Peat may be added to the fire to impart flavour from the smoke. [https://www.scotchwhiskyexperience.co.uk/about-whisky/making]



Boolean Model – Example: How is Whisky made?

D1 Step 1 - Malting

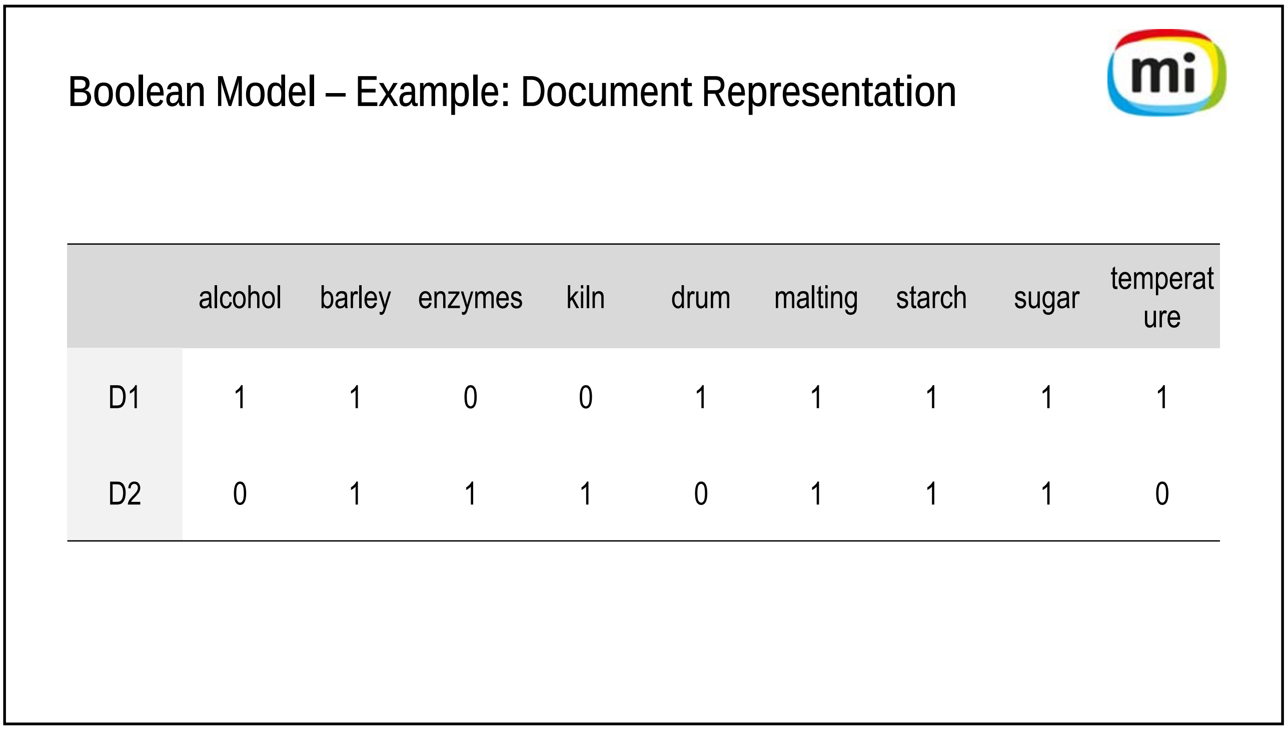
Barley contains **starch** and it is this **starch** which needs to be converted into soluble **sugars** to make **alcohol**. For this to occur, the **barley** must undergo germination and this first part of the process is called **'malting'**. Each distiller has their own preference about the type of **barley** they buy, but they need a type that produce high yields of soluble **sugar**. The **barley** is soaked for 2-3 days in warm water and then traditionally spread on the floor of a building called a **malting** house. It is turned regularly to maintain a constant **temperature**. This is also carried out on a commercial scale in large **drums** which rotate.

[http://www.whiskyforeveryone.com/whisky_basics/how_is_whisky_made.html]

D2 1. Malting

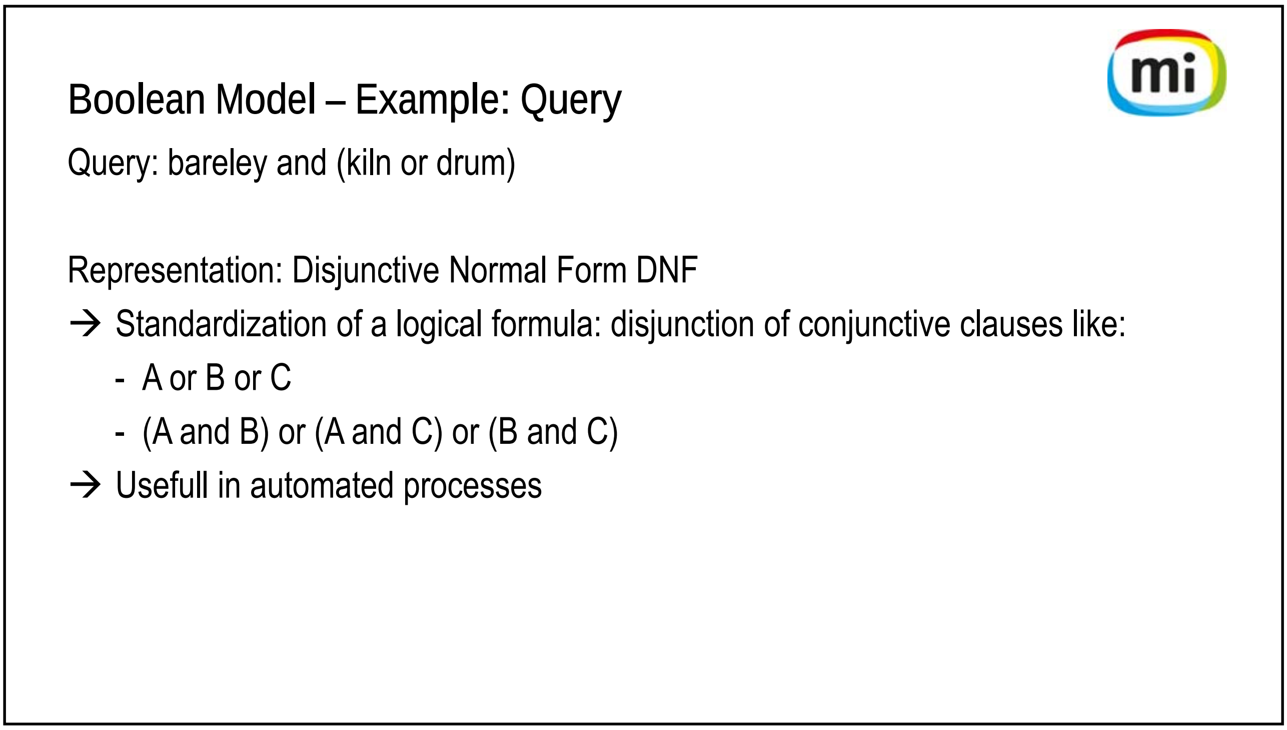
Best quality **barley** is first steeped in water and then spread out on malting floors to germinate. It is turned regularly to prevent the build up of heat. Traditionally, this was done by tossing the **barley** into the air with wooden shovels in a malt barn adjacent to the **kiln**.

During this process **enzymes** are activated which convert the **starch** into **sugar** when mashing takes place. After 6 to 7 days of germination the **barley**, now called green malt, goes to the **kiln** for drying. This halts the germination. The heat is kept below 70°C so that the **enzymes** are not destroyed. Peat may be added to the fire to impart flavour from the smoke. [https://www.scotchwhiskyexperience.co.uk/about-whisky/making]



Boolean Model – Example: Document Representation

	alcohol	barley	enzymes	kiln	drum	malting	starch	sugar	temperature
D1	1	1	0	0	1	1	1	1	1
D2	0	1	1	1	0	1	1	1	0



Boolean Model – Example: Query

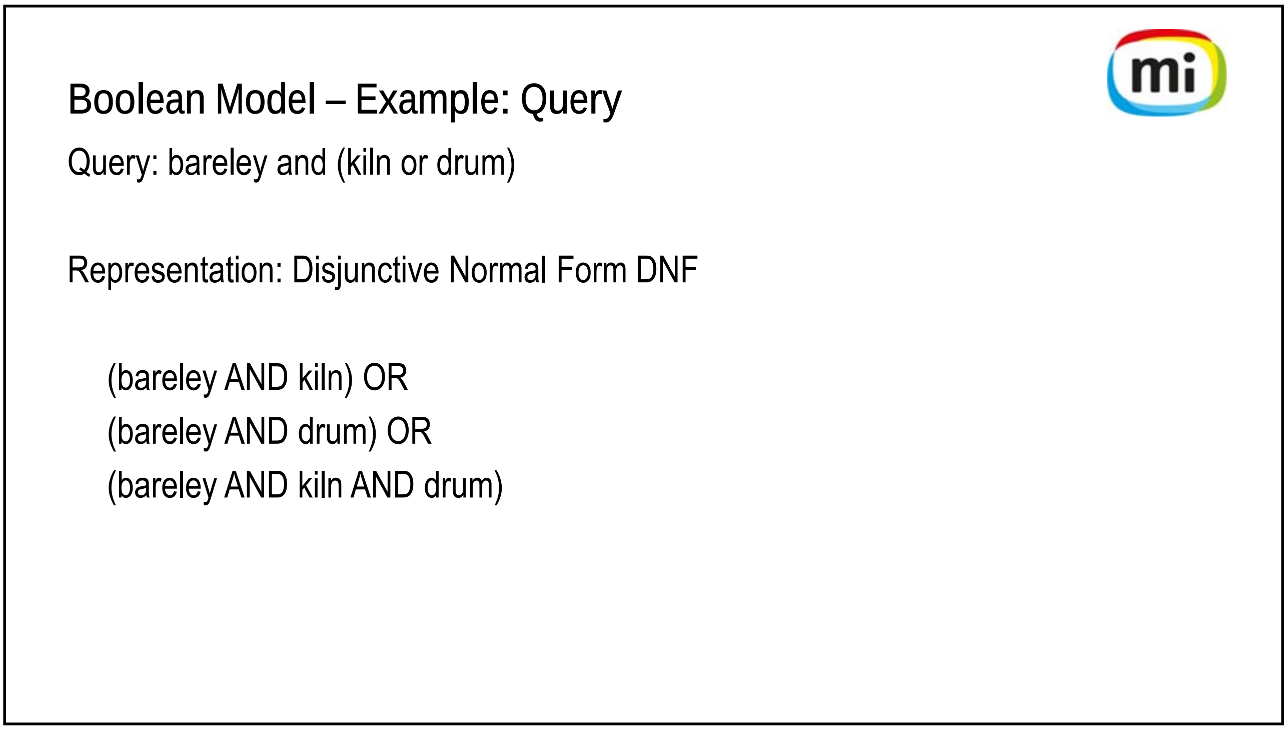
Query: bareley and (kiln or drum)

Representation: Disjunctive Normal Form DNF

→ Standardization of a logical formula: disjunction of conjunctive clauses like:

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

→ Usefull in automated processes

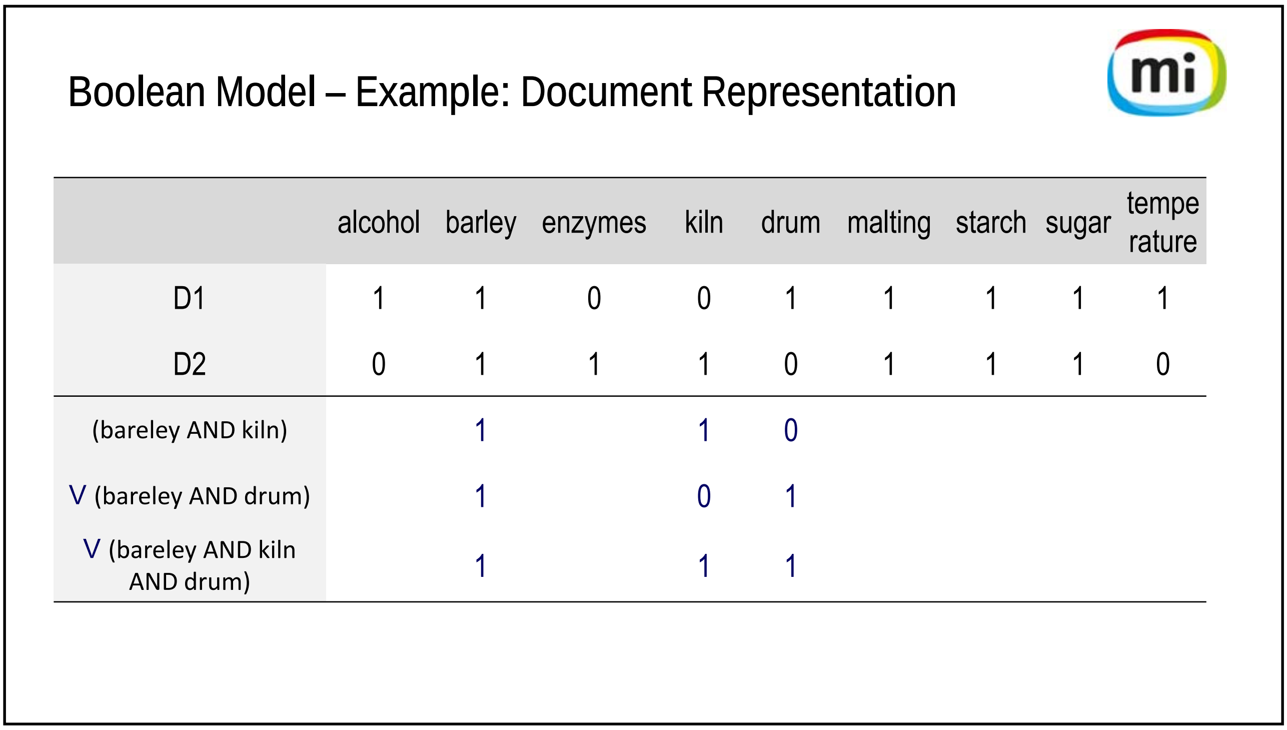


Boolean Model – Example: Query

Query: bareley and (kiln or drum)

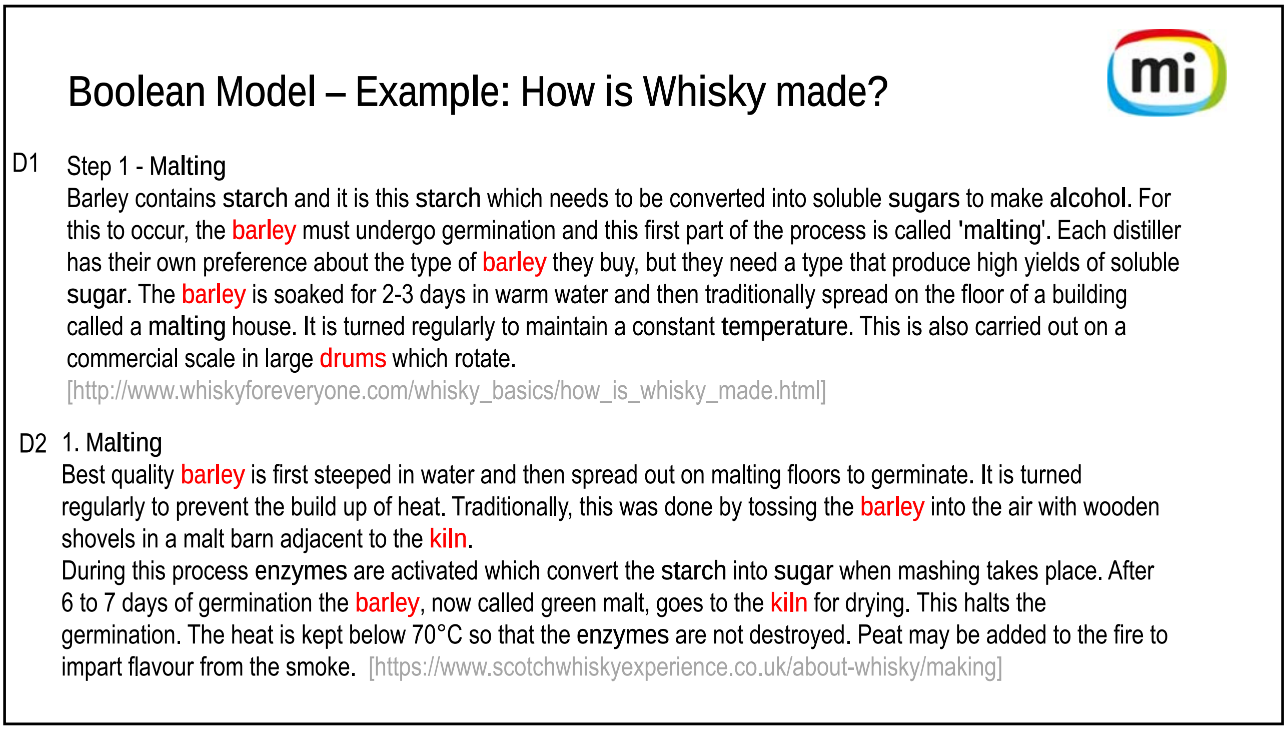
Representation: Disjunctive Normal Form DNF

(bareley AND kiln) OR
(bareley AND drum) OR
(bareley AND kiln AND drum)



Boolean Model – Example: Document Representation

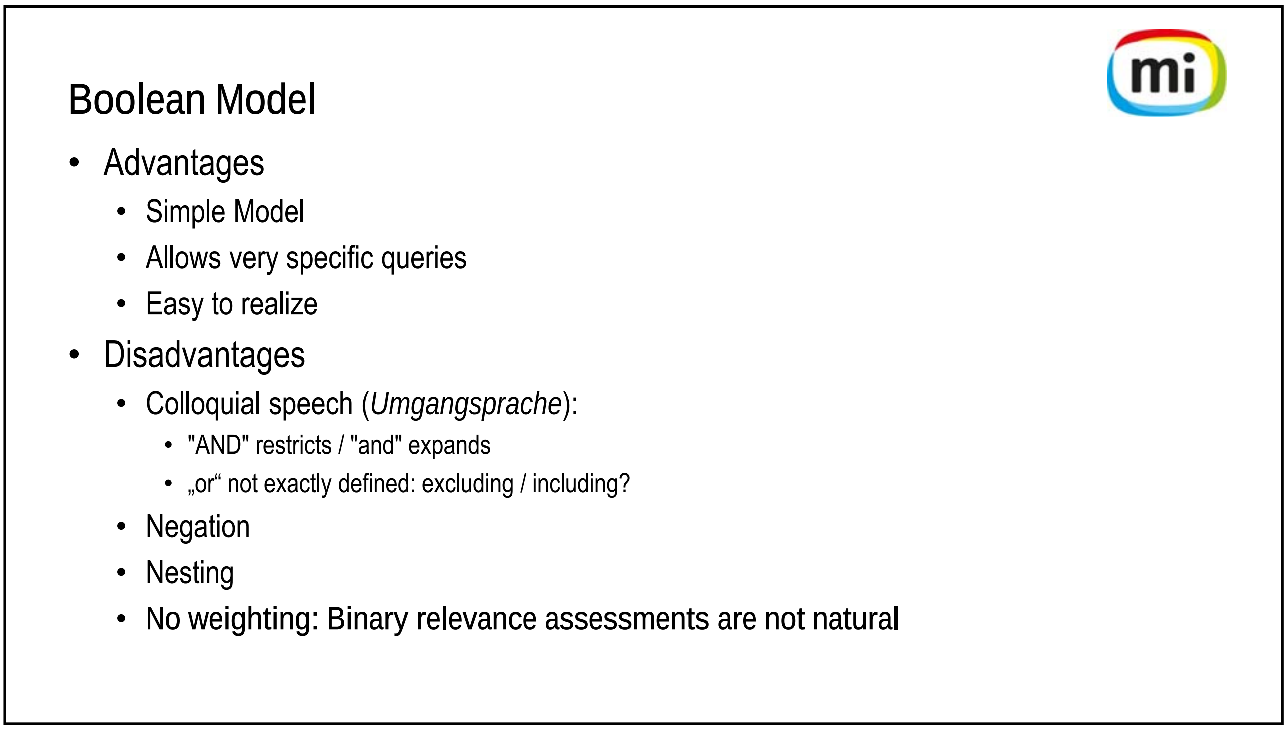
	alcohol	barley	enzymes	kiln	drum	malting	starch	sugar	temperature
D1	1	1	0	0	1	1	1	1	1
D2	0	1	1	1	0	1	1	1	0
(bareley AND kiln)		1		1	0				
∨ (bareley AND drum)		1		0	1				
∨ (bareley AND kiln AND drum)		1		1	1				



Boolean Model – Example: How is Whisky made?

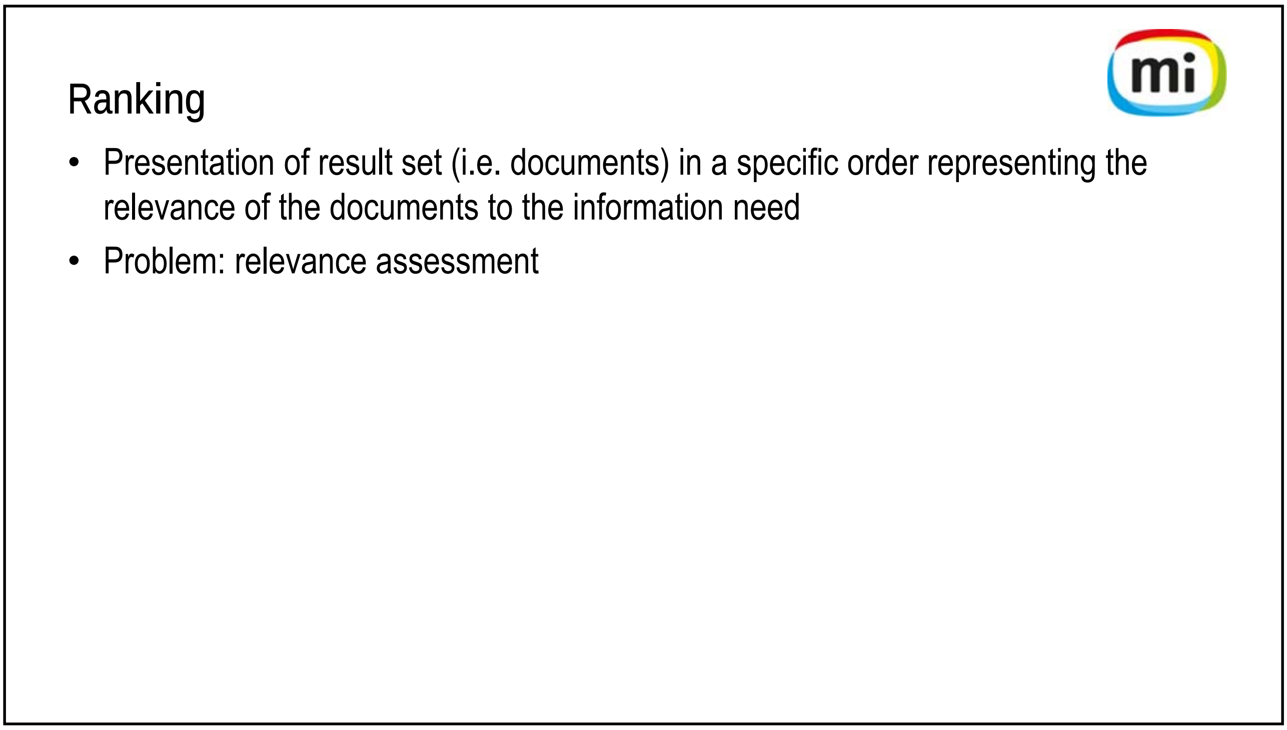
D1 Step 1 - Malting
Barley contains **starch** and it is this **starch** which needs to be converted into soluble **sugars** to make **alcohol**. For this to occur, the barley must undergo germination and this first part of the process is called **'malting'**. Each distiller has their own preference about the type of barley they buy, but they need a type that produce high yields of soluble **sugar**. The barley is soaked for 2-3 days in warm water and then traditionally spread on the floor of a building called a **malting** house. It is turned regularly to maintain a constant **temperature**. This is also carried out on a commercial scale in large drums which rotate.
[http://www.whiskyforeveryone.com/whisky_basics/how_is_whisky_made.html]

D2 1. Malting
Best quality barley is first steeped in water and then spread out on malting floors to germinate. It is turned regularly to prevent the build up of heat. Traditionally, this was done by tossing the barley into the air with wooden shovels in a malt barn adjacent to the kiln.
During this process **enzymes** are activated which convert the **starch** into **sugar** when mashing takes place. After 6 to 7 days of germination the barley, now called green malt, goes to the kiln for drying. This halts the germination. The heat is kept below 70°C so that the **enzymes** are not destroyed. Peat may be added to the fire to impart flavour from the smoke. [https://www.scotchwhiskyexperience.co.uk/about-whisky/making]



Boolean Model

- Advantages
  - Simple Model
  - Allows very specific queries
  - Easy to realize
- Disadvantages
  - Colloquial speech (*Umgangsprache*):
    - "AND" restricts / "and" expands
    - „or“ not exactly defined: excluding / including?
  - Negation
  - Nesting
  - No weighting: Binary relevance assessments are not natural



Ranking

- Presentation of result set (i.e. documents) in a specific order representing the relevance of the documents to the information need
- Problem: relevance assessment

Term Weighting I – tf

- Statistical assumption 1: The more often a term is used in a document, the better it describes the document.

- Examples:
  - D1: Apples are typical fruits of central Europe. In southern Europe you will find fruits which are more depending on sun light and mild climate like Oranges, Pineapples, …
  - D2: Apples are typical fruits of central Europe. There are many sorts of apples like the *Pink Lady*, a very sweet apple, or sour apples like *Granny Smith*. You can use apples in many ways. Just think about apple pie, apple juice, apple tea or apple crumble.

- Which document is about apples and which is about fruits in general?

Term Weighting I - tf

Term Frequency (tf)

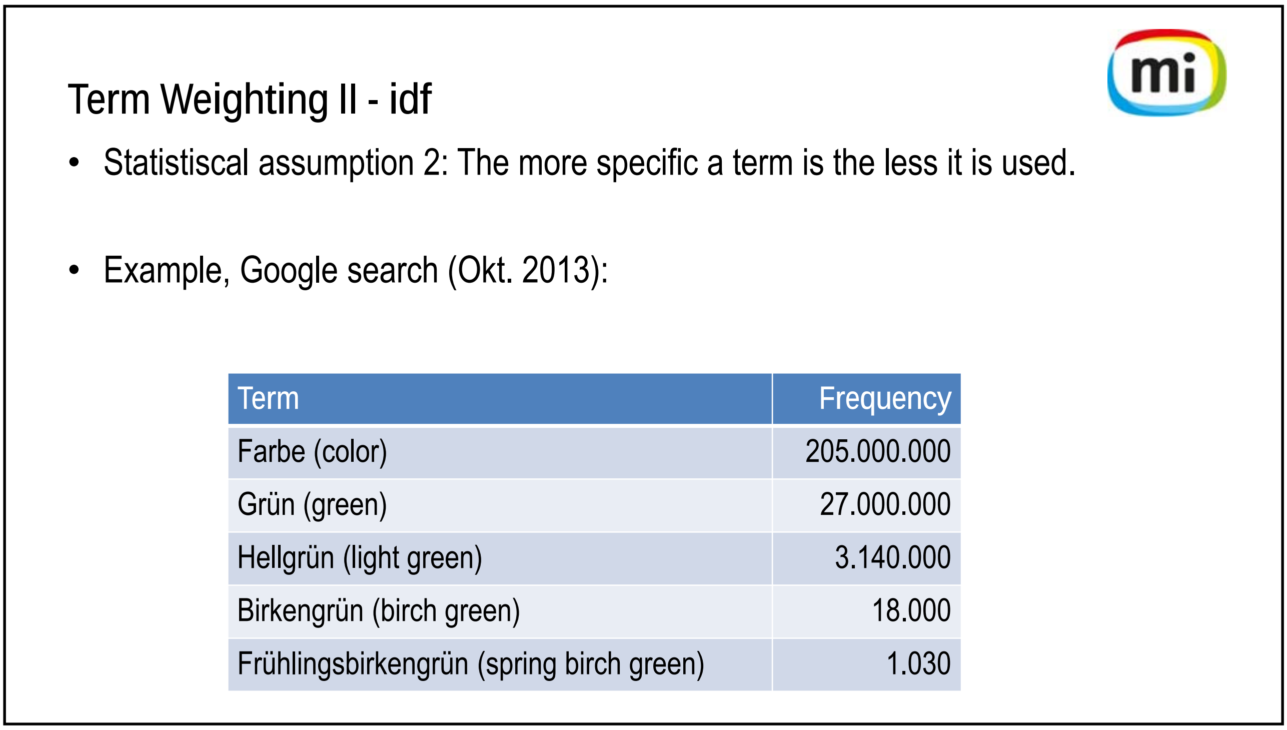
$$f_{i,m} = \frac{freq_{i,m}}{\max_l \; freq_{l,m}}$$

with:

$freq_{i,m}$ = frequency of term t_i in document d_m

$\max_l freq_{l,m}$ = frequency of the term with the highest fequency in document d_m

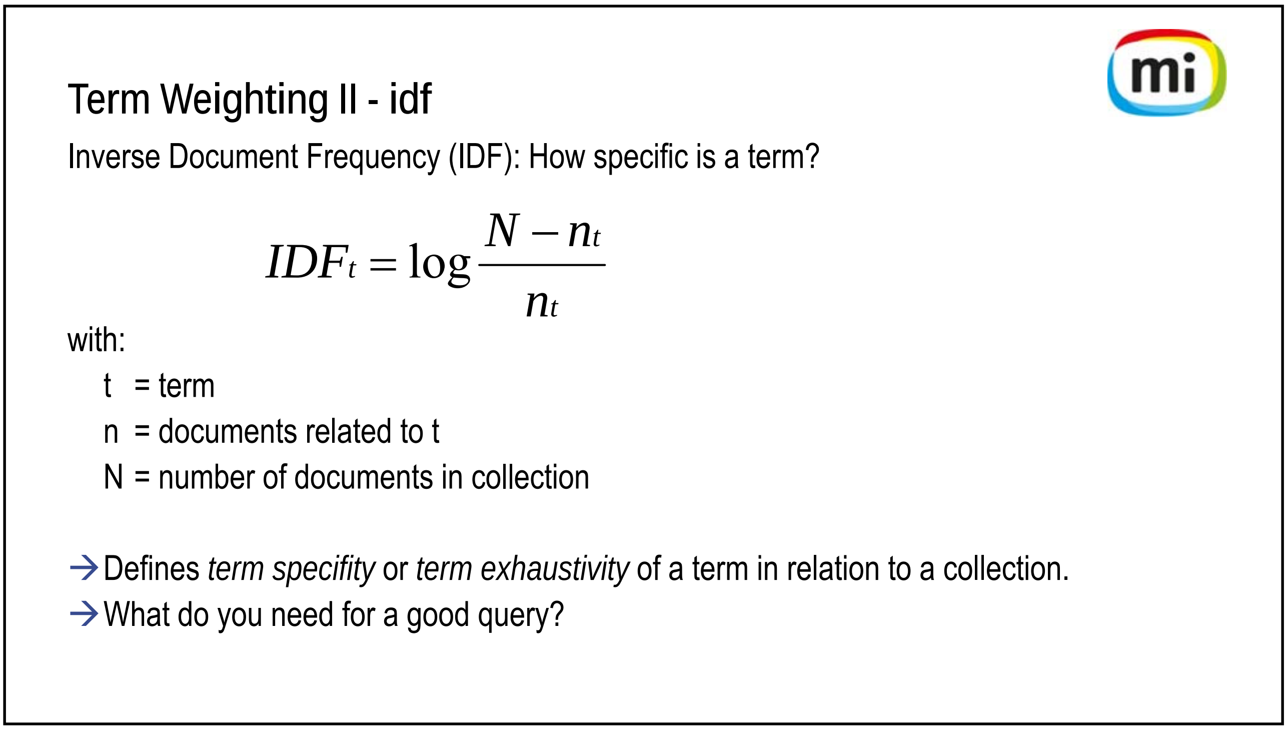
$f_{i,m}$ = normalized frequency of term t_i in document d_m



Term Weighting II - idf

- Statistiscal assumption 2: The more specific a term is the less it is used.
- Example, Google search (Okt. 2013):

Term	Frequency
Farbe (color)	205.000.000
Grün (green)	27.000.000
Hellgrün (light green)	3.140.000
Birkengrün (birch green)	18.000
Frühlingsbirkengrün (spring birch green)	1.030



Term Weighting II - idf

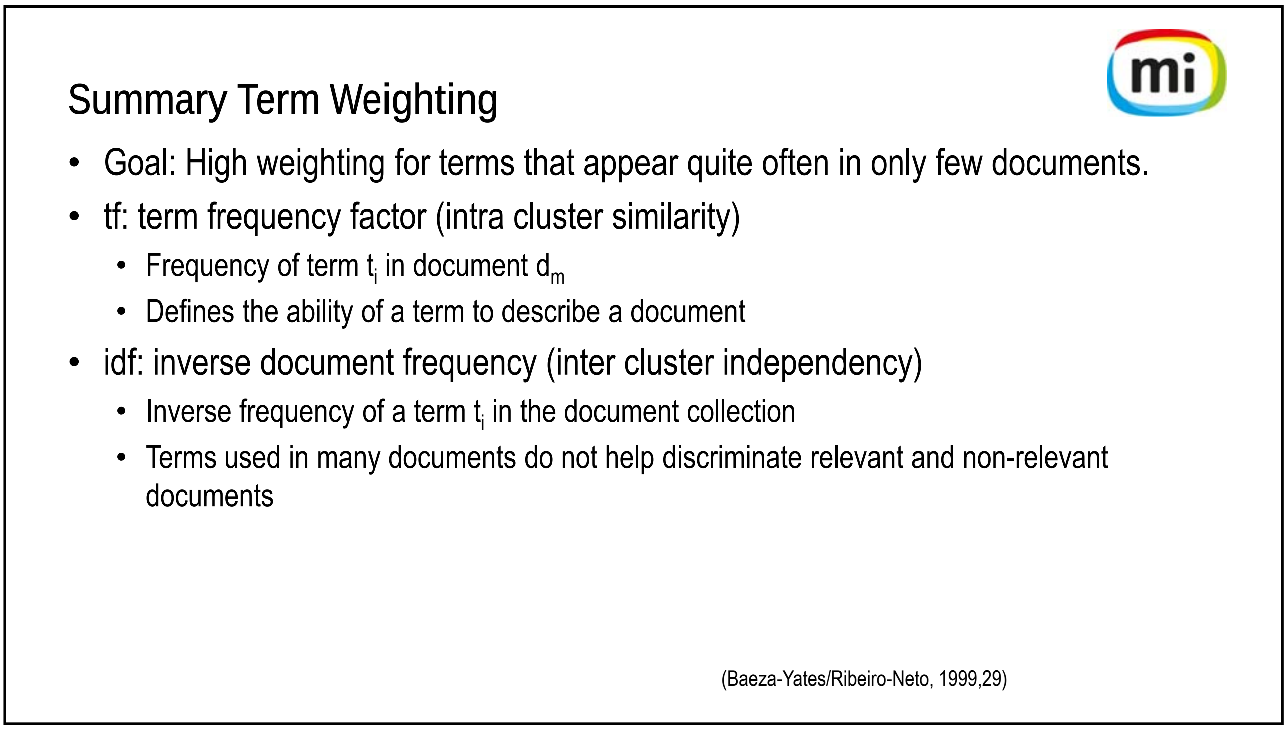
Inverse Document Frequency (IDF): How specific is a term?

$$IDF_t = \log \frac{N - n_t}{n_t}$$

with:

t = term
n = documents related to t
N = number of documents in collection

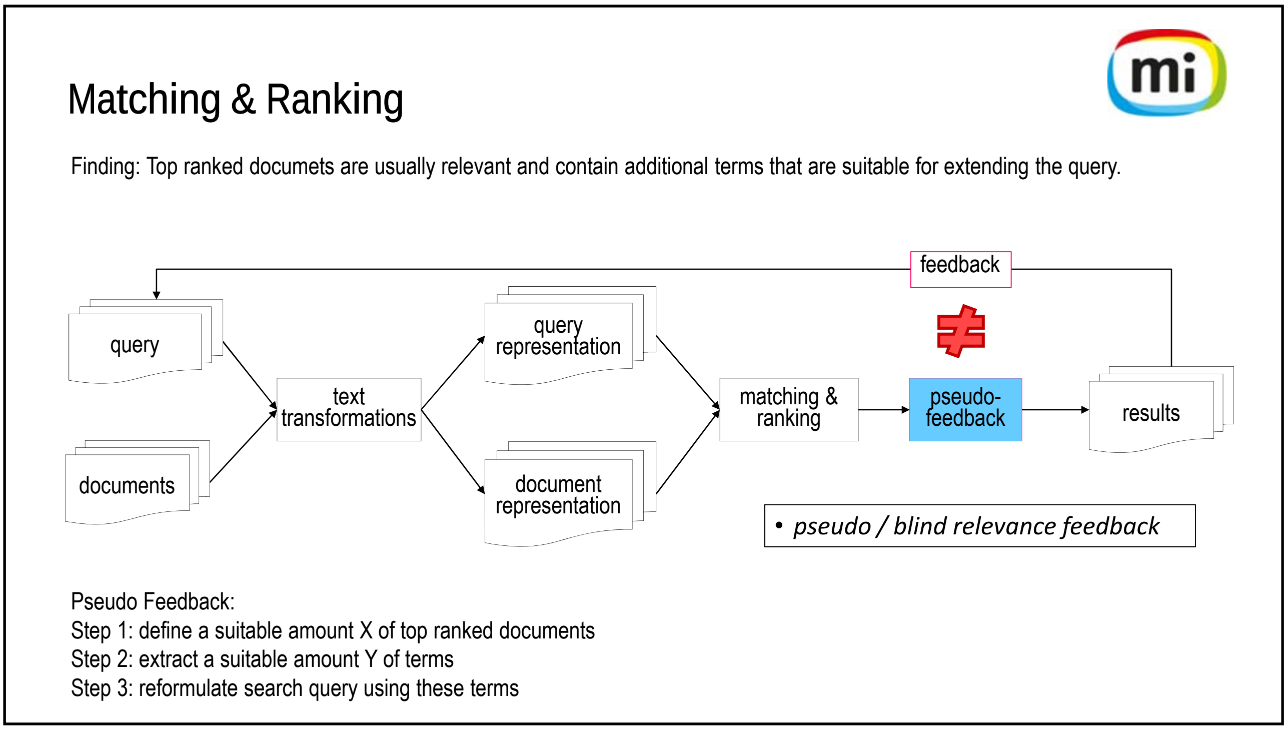
→Defines *term specifity* or *term exhaustivity* of a term in relation to a collection.
→What do you need for a good query?



Summary Term Weighting

- Goal: High weighting for terms that appear quite often in only few documents.
- tf: term frequency factor (intra cluster similarity)
  - Frequency of term t_i in document d_m
  - Defines the ability of a term to describe a document
- idf: inverse document frequency (inter cluster independency)
  - Inverse frequency of a term t_i in the document collection
  - Terms used in many documents do not help discriminate relevant and non-relevant documents

(Baeza-Yates/Ribeiro-Neto, 1999,29)



Matching & Ranking

Finding: Top ranked documets are usually relevant and contain additional terms that are suitable for extending the query.

feedback

≠

query

documents

text transformations

query representation

document representation

matching & ranking

pseudo-feedback

results

- *pseudo / blind relevance feedback*

Pseudo Feedback:
Step 1: define a suitable amount X of top ranked documents
Step 2: extract a suitable amount Y of terms
Step 3: reformulate search query using these terms