Information Retrieval

Andreas Rauber
http://www.ifs.tuwien.ac.at/~andi

Text IR : Outline

- overview
- pre-processing
- indexing
- term weighting
- retrieval
- text categorization (ATC)
- question answering, summarization

IR Terminology

- document
  - data object
  - text documents, hypertext, (multimedia objects, images, video, audio, 3D-objects -> see separate lecture)
- corpus / collection
  - set of documents, aka collection
- retrieval
  - given a query, return relevant documents
  - several other tasks

IR Tasks

- retrieval
- cross-language retrieval
- XML-retrieval
- text categorization (ATC)
- text clustering
- question answering
- topic detection
- summarization
- filtering

Basic Procedure

- pre-processing
  - collection cleansing
  - identification of relevant document parts
  - stemming
  - stop-word removal
- indexing
  - selecting type of feature set
  - term weighting
  - (feature selection)
  - feature space transformation
- retrieval / organization / analysis

IR vs. DB

	DB	IR
data	structured	unstructured
attrib. semantics	defined	ambiguous
queries	well-defined	free text
retrieval	exact	imprecise

IR and Related Disciplines

- database systems
- machine learning and AI
- natural language processing (NLP)
- forensics, intelligence
- digital libraries
- semantic web
- library and information science

Principles of IR

- take words in document corpus
- weight the words
- take the words in a query
- compare them to the words in corpus
- pick those with the highest match
- doesn't care about
  - meaning of words
  - semantics of a sentence
- simple, but works surprisingly well

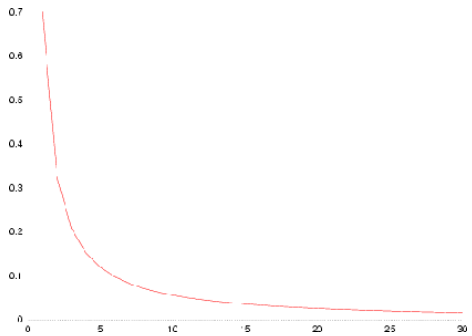
Text Statistics

- stable patterns in language use
- very few words occur very often
- most words occur very rarely
- redundancy in natural language
- word co-occurrence
- word sense diambiguation

Zipf's Law

- relates the frequency *f* of term *t* to its rank *R* in a vocabulary of size *N* when sorted in decending order
- the i-th most frequent term occurs as many times as 1/i times the most frequent one
- exponential distribution of english words; nature of communication is on efficiency;
  -> most common words tend to be short and follow Zipf's distribution
- George K. Zipf, Human Behaviour and the Principle of Least-Effort, Addison-Wesley, Cambridge MA, 1949

Zipf's Law



Zipf's Law: x-axis: rank; y-axis: occurrence frequency

Heap's Law

- predicts the number V of distinct terms:

$$V = K N^{\beta}$$

where

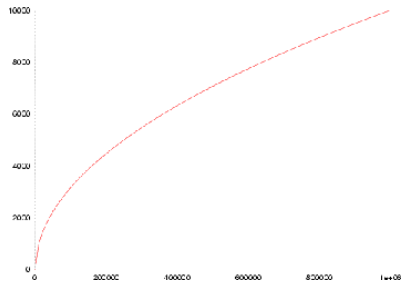
K - corpus constant, commonly $10 < K < 100$

β - corpus constant, commonly $0.4 < \beta < 0.6$ for english

N - number of word occurrences

- H. S. Heaps. Information Retrieval - Computational and Theoretical Aspects. Academic Press, 1978

Heap's Law



Heap's Law: x-axis: vocabulary size; y-axis: text size

- the more text is collected, the smaller the increase in new vocabulary

Some Term Frequencies

term list:

0 all 51 108
1 after 223 416
2 nassau 7 11
3 december 14 15
4 first 186 301
5 propos 45 60
6 help 81 135
7 nato 44 116
8 develop 22 25
9 nuclear 45 122
10 strik 45 76
11 forc 161 333
12 europ 72 214
13 mad 134 177
14 attempt 49 55
15 devis 4 5
16 plan 104 165
17 the 266 694

sorted - top df

32 from 354 1223 1 1 1
67 wer 312 845 1 1 1
154 their 309 814 1 1 1
79 hav 297 908 1 1 1
94 year 290 667 1 1 1
111 when 267 578 1 1 1
17 the 266 694 1 1 1
190 been 262 552 1 1 1
145 only 260 503 1 1 1
83 mor 254 556 1 1 1
109 which 251 539 1 1 1
70 governme 240 674 1 1 1

78 into 237 515 1 1 1
204 thi 232 480 1 1 1
176 would 226 575 1 1 1
173 ther 226 433 1 1 1
152 said 225 440 1 1 1
1 after 223 416 1 1 1
35 over 220 399 1 1 1

Text IR : Outline

- overview
- pre-processing
- indexing
- term weighting
- retrieval
- text categorization (ATC)
- question answering, summarization

Pre-Processing

- collection cleansing
  - removal of empty/ill-formatted documents
  - definition of document parts relevant for indexing
  - if necessary split documents (text mining)
  - removal of formatting information
  - transcription of encodings
  - any task-specific preprocessing

Pre-Processing

- remove stop-words
- based on manually created stop-word list
- collection-specific stop-words
- may be defined automatically using term frequency
- usually very few stop-words

Pre-Processing

- Stemming
  - remove word stems
  - conflates morphological variants
  - allows more effective matching
  - linguistically correct stemmers
  - linguistically incorrect stemmers
  - most popular for english: Porter's stemmer

Pre-Processing

- Porter stemmer
- M.F. Porter, 1980, An algorithm for suffix stripping, Program, 14(3):130-137
- determines number m of vowel-consonant sequences:

  [c] (vc){m} [v]

- set of rules to remove suffixes

Pre-Processing

- Porter stemmer rules, 5 stages
- stage 1
  - sses -> ss *(caresses -> caress)*
  - ies -> i *(ponies -> poni)*
  - s -> NULL *(cats -> cat)*
  - y -> i *(study -> studi)*
- stage 2: if m > 0
  - eed -> ee *(agreed -> agree)*
  - v{*}ed -> NULL *(plastered -> plaster)*

Pre-Processing

- Porter stemmer rules, stages 3-5
- remove:
  - icate|ative|alize|iciti|ical|ful|ness
  - al|ance|ence|er|ic|able|ible|ant|ement|ment|ent|ou|ism|ate|iti|ous|ive|ize
- rewrite:
  - 'ational'=>'ate', 'tional'=>'tion', 'enci'=>'ence', 'anci'=>'ance', 'izer'=>'ize', 'bli'=>'ble', 'alli'=>'al', 'entli'=>'ent', 'eli'=>'e', 'ousli'=>'ous', 'ization'=>'ize', 'ation'=>'ate', 'ator'=>'ate', 'alism'=>'al', 'iveness'=>'ive', 'fulness'=>'ful', 'ousness'=>'ous', 'aliti'=>'al', 'iviti'=>'ive', 'biliti'=>'ble', 'logi'=>'log'
  - ('icate'=>'ic', 'ative'=>'', 'alize'=>'al', 'iciti'=>'ic', 'ical'=>'ic', 'ful'=>'', 'ness'=>'');

Pre-Processing

- stemming example:
- ever since world war ii even through ten years of soviet occupation the austrian government has been run jointly by the conservative people's party and the socialist party, like two polite, equally weighted cousins on an inert seesaw . so scrupulously balanced is the coalition and the proporz system of dividing up the jobs that, according to viennese table talk, if there is one people's party putzfrau (charwoman) in a government building, there must be a socialist putzfrau too.
- ever sinc world war ii even through ten year of soviet occup the austrian govern ha been run jointli by the conserv peopl's parti and the socialist parti, like two polit, equal weight cousin on an inert seesaw . so scrupul balanc is the coalit and the proporz system of divid up the job that, accord to viennes tabl talk, if there is on peopl's parti putzfrau (charwoman) in a govern build, there must be a socialist putzfrau too.

Pre-Processing

- problems with stemming:
  - sometimes too aggressive, conflating words not to be conflated *(organization and organ)*
  - does not succeed to conflate wherever desired
    (europe and european, matrix and matrices)
  - stems are not gramatically correct words

Text IR : Outline

- overview
- pre-processing
- indexing
- term weighting
- retrieval
- text categorization (ATC)
- question answering, summarization

Indexing

- some terminology
  - d - document
  - t - term
  - N - number of documents in corpus
  - tf(d) - term frequency:
    number of times term t appears in document d
  - df(t) - document frequency:
    number of documents in collection that term t occurs in

Indexing

- selecting words to represent document collection
- manual:
  - controlled vocabulary, thesaurus
  - cost-intensive
- automatic:
  - automatically select words
  - fast
  - comparable retrieval performance

Indexing

- decide on type of features (terms):
  - n-grams
  - words (word stems)
  - word co-occurrences, (word n-grams)
  - concepts

Indexing

- "bag of words" (BOW)
- most common indexing technique
- simply take list of all terms
- independence assumption
- creates a high-dimensional vector space
- every document occupies a position in this space
- similar documents close to each other

Indexing

- n-grams as terms
- sequence of n characters (n = 3, 4)
- including or excluding white space, punctuation marks, etc.
- eliminates need for stemming
- features not interpretable
- *ngr, gra, ram, ams, msa, sas, ste,...*

Indexing

- words or word stems as terms
- list of all words
- polysemy taken care of by redundancy
- most common form of indexing
- common settings:
  - minimum word length: 3
  - ignore numbers, punctuation
- *word stem term list all polysemy*

Indexing

- phrases / co-occurrences as terms
- phrases of length 2, 3
- bigrams of co-occurrences within window
  - window 5: phrase-level
  - window 20: sentence-level
  - window 250: paragraph-level
- may include capitalization: proper names
- *co occurrence, window level, white house*

Indexing

- semantic concepts
- indexers for certain concepts, such as
  - dates
  - locations
  - persons
  - proper nouns, acronyms
  - topics
- includes NLP processing
- highly domain-specific

Text IR : Outline

- overview
- pre-processing
- indexing
- term weighting
- retrieval
- text categorization (ATC)
- question answering, summarization

Term Weighting

- different terms carry different amounts of semantics
- stop-words, VERY rare words, misspelled words
- assign different weights to individual terms
- also used for term selection
- based on
  - location of term occurrence *(title,...)*
  - type of term *(nouns, proper nouns, ...)*
  - term statistics (term frequency, corpus frequency)

Term Weighting

- binary: 0 | 1
- term frequency
- tfidf or: tf x idf
- term discrimination model

Term Weighting - tf

- tf - term frequency
- the more often a term occurs in a document, the more important it is
- often includes normalization based on
  - maximum term frequency
  - document length
- OKAPI-tf: $$wtf = \frac{tf}{tf + 0.5 + 1.5 \frac{doc_{length}}{avg_{doc_{length}}}}$$

Term Weighting - idf

- most common model
- **tf:** the more often a term occurs in a document, the more important it is
- **df:** the more documents in a collection contain the term, the less important it is
- many variants of tfidf
  - tfidf = tf * 1/df
  - tfidf = tf * ln(N/df) (most common)
  - tfidf = tf * ln ((N-df)/df)

Term Weighting - disc

- compute discrimination value of term
- compare density of document space (average similarity of documents) with or without a term

$$D = k \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} s(d_i, d_j)$$

- D average density
- D_t density with term t removed
- DISC = D - D_t

Term Weighting - disc

- DISC > 0: good discriminator
- DISC < 0: poor discriminator
- does not consider relevant / not relevant discrimination, but general discrimination
- advanced discriminator measures if relevance information is available
  -> see ATC

Term Weighting - df-Selection

- document frequency to select terms
- remove most common terms
  - df > 10% - 50%
  - collection-specific stop-word list
  - very few words removed
- remove very rare terms
  - df > 2-10 docs
  - reduces dimensionality significantly

Text IR : Outline

- overview
- pre-processing
- indexing
- term weighting
- retrieval
- text categorization (ATC)
- question answering, summarization