CAI: Cerca i Anàlisi de la Informació
Grau en Ciència i Enginyeria de Dades, UPC

Introduction. Preprocessing. Laws

September 13, 2021

Slides by Marta Arias, José Luis Balcázar, Ramon Ferrer-i-Cancho, Ricard Gavaldà, Department of Computer Science, UPC

Contents

Information Retrieval

The origins: Librarians, census, government agencies...

Gradually information was digitalized

Now, most information is digital at birth

The web

The web changed everything

Everybody could set up a site and publish information

Now you don't even set up a site

Web search as a comprehensive of Computing

Algorithms, data structures, computer architecture, networking, logic, discrete mathematics, interface design, user modelling, databases, software engineering, programming languages, multimedia technology, image and sound processing, data mining, artificial intelligence, . . .

Web search as a comprehensive of Computing

Algorithms, data structures, computer architecture, networking, logic, discrete mathematics, interface design, user modelling, databases, software engineering, programming languages, multimedia technology, image and sound processing, data mining, artificial intelligence, . . .

Think about it: Search billions of pages and return satisfying results in tenths of a second

Information Retrieval versus Database Queries

In Information Retrieval,

- We may not know where the information is

Information Retrieval versus Database Queries

In Information Retrieval,

- We may not know where the information is
- We may not know whether the information exists

Information Retrieval versus Database Queries

In Information Retrieval,

- We may not know where the information is
- We may not know whether the information exists
- We don't have a schema as in relational DB

Information Retrieval versus Database Queries

In Information Retrieval,

- We may not know where the information is
- We may not know whether the information exists
- We don't have a schema as in relational DB
- We may not know exactly what information we want

Information Retrieval versus Database Queries

In Information Retrieval,

- We may not know where the information is
- We may not know whether the information exists
- We don't have a schema as in relational DB
- We may not know exactly what information we want
  - Or how to define it with a precise query
  - "Too literal" answers may be undesirable

Hierarchical/Taxonomic vs. Faceted Search

Biology:

Animalia $\rightarrow$ Chordata $\rightarrow$ Mammalia $\rightarrow$ Artiodactyla $\rightarrow$ Giraffidae $\rightarrow$ Giraffa

Universal Decimal Classification (e.g. Libraries):

0 Science and knowledge $\rightarrow$
00 Prolegomena. Fundamentals of knowledge and culture. Propaedeutics $\rightarrow$
004 Computer science and technology. Computing $\rightarrow$
004.6 Data $\rightarrow$

004.63 Files

Taxonomic vs. Faceted Search

Faceted search:

By combination of features (facets) in the data

“It is black and yellow & lives near the Equator”





Models

An Information Retrieval Model is specified by:

- A notion of document (= an abstraction of real documents)

Models

An Information Retrieval Model is specified by:

- A notion of document (= an abstraction of real documents)
- A notion of admissible query (= a query language)

Models

An Information Retrieval Model is specified by:

- A notion of document (= an abstraction of real documents)
- A notion of admissible query (= a query language)
- A notion of relevance

Models

An Information Retrieval Model is specified by:

- A notion of document (= an abstraction of real documents)
- A notion of admissible query (= a query language)
- A notion of relevance
  - A function of pairs (document,query)
  - Telling whether / how relevant the document is for the query
  - Range: Boolean, rank, real values, ...

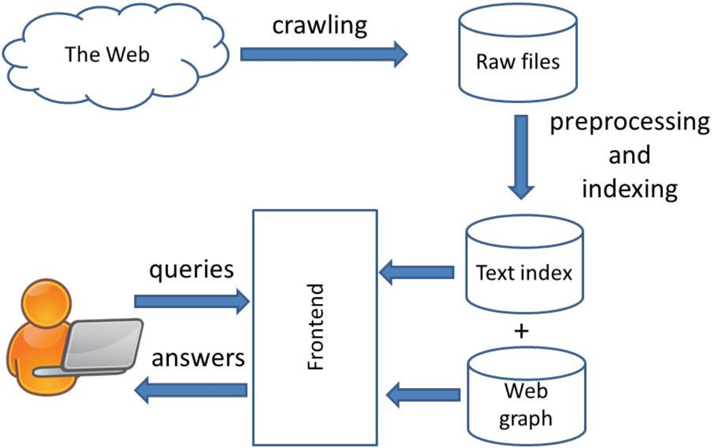
Textual Information

Focus for half the course:

Retrieving (hyper)text documents from the web

- Hypertext documents contain terms and links.
- Users issue queries to look for documents.
- Queries typically formed by terms as well.

The Information Retrieval process, I



The Information Retrieval process, II

Offline process:

- Crawling
- Preprocessing
- Indexing

Goal:

Prepare data structures to make online process fast.

- Can afford long computations. For example, scan each document several times.
- Must produce reasonably compact output (data structure).

The Information Retrieval process, III

Online process:

- Get query
- Retrieve relevant documents
- Rank documents
- Format answer, return to user

Goal:

Instantaneous reaction, useful visualization.

- May use additional info: user location, ads, . . .

Preprocessing

Term extraction

Potential actions:

- Parsing: Extracting structure (if present, e.g. HTML).
- Tokenization: decomposing character sequences into individual units to be handled.
- Enriching: annotating units with additional information.
- Either Lemmatization or Stemming: reduce words to roots.

Tokenization

Group characters

Join consecutive characters into “words”: use spaces and punctuation to mark their borders.

Similar to lexical analysis in compilers.

Tokenization

Group characters

Join consecutive characters into “words”: use spaces and punctuation to mark their borders.

Similar to lexical analysis in compilers.

It seems easy, but. . .

Tokenization

- IP and phone numbers, email addresses, URL's,
- "R+D", "H&M", "C#", "I.B.M.", "753 B.C.",

Tokenization

- IP and phone numbers, email addresses, URL’s,
- “R+D”, “H&M”, “C#”, “I.B.M.”, “753 B.C.”,
- Hyphens:
  - change “afro-american culture” to “afroamerican culture”?

Tokenization

- IP and phone numbers, email addresses, URL's,
- "R+D", "H&M", "C#", "I.B.M.", "753 B.C.",
- Hyphens:
  - change "afro-american culture" to "afroamerican culture"?
  - but not "state-of-the-art" to "stateoftheart",
  - how about "cheap San Francisco-Los Angeles flights".

Tokenization

- IP and phone numbers, email addresses, URL's,
- "R+D", "H&M", "C#", "I.B.M.", "753 B.C.",
- Hyphens:
  - change "afro-american culture" to "afroamerican culture"?
  - but not "state-of-the-art" to "stateoftheart",
  - how about "cheap San Francisco-Los Angeles flights".

A step beyond is Named Entity Recognition.

- "Fahrenheit 451", "The president of the United States", "David A. Mix Barrington", "June 6th, 1944"

Tokenization

Case folding

Move everything into lower case, so searches are case-independent…

Tokenization

Case folding

Move everything into lower case, so searches are case-independent…

But:

- "USA" might not be "usa",
- "Windows" might not be "windows",
- "bush" versus various famous members of a US family…

Tokenization

Stopword removal

Words that appear in most documents, or that do not help.

- prepositions, articles, some adverbs,
- “emotional flow” words like “essentially”, “hence”...
- very common verbs like “be”, “may”, “will”...

May reduce index size by up to 40%.

Tokenization

Stopword removal

Words that appear in most documents, or that do not help.

- prepositions, articles, some adverbs,
- "emotional flow" words like "essentially", "hence"...
- very common verbs like "be", "may", "will"...

May reduce index size by up to 40%.
But note:

- "may", "will", "can" as nouns are not stopwords!
- "to be or not to be", "let there be light", "The Who"

Tokenization

Stopword removal

Words that appear in most documents, or that do not help.

- prepositions, articles, some adverbs,
- “emotional flow” words like “essentially”, “hence”…
- very common verbs like “be”, “may”, “will”…

May reduce index size by up to 40%.
But note:

- “may”, “will”, “can” as nouns are not stopwords!
- “to be or not to be”, “let there be light”, “The Who”

Current tendency: keep everything in index, and filter docs by relevance.

Tokenization

Summary

- Language dependent...
- Application dependent...
  - search on a library?
  - search on an intranet?
  - search on the Web?
- Crucial for efficient retrieval!
- Requires to laboriously hardwire into retrieval systems many many different rules and exceptions.

Enriching

Enriching means that each term is associated to additional information that can be helpful to retrieve the "right" documents. For instance,

- Synonims: gun $\rightarrow$ weapon;
- Related words, definitions: laptop $\rightarrow$ portable computer;
- Categories: fencing $\rightarrow$ sports;
- POS tags (part of speech labels):
  - Part-of-speech (POS) tagging.
  - "Un hombre bajo me acompaña cuando bajo a esconderme bajo la escalera a tocar el bajo."
  - "a ship has sails" vs. "John often sails on weekends".
  - "fencing" as sport or "fencing" as setting up fences?

Enriching

Enriching means that each term is associated to additional information that can be helpful to retrieve the "right" documents. For instance,

- Synonims: gun $\rightarrow$ weapon;
- Related words, definitions: laptop $\rightarrow$ portable computer;
- Categories: fencing $\rightarrow$ sports;
- POS tags (part of speech labels):
  - Part-of-speech (POS) tagging.
  - "Un hombre bajo me acompaña cuando bajo a esconderme bajo la escalera a tocar el bajo."
  - "a ship has sails" vs. "John often sails on weekends".
  - "fencing" as sport or "fencing" as setting up fences?

A step beyond is Word Sense Disambiguation.

Lemmatizing and Stemming

Two alternative options

Stemming: removing suffixes

swim, swimming, swimmer, swimmed $\rightarrow$ swim

Lemmatizing and Stemming

Two alternative options

Stemming: removing suffixes

swim, swimming, swimmer, swimmed $\rightarrow$ swim

Lemmatizing: reducing the words to their linguistic roots.

be, am, are, is $\rightarrow$ be

gave $\rightarrow$ give

feet $\rightarrow$ foot, teeth $\rightarrow$ tooth,

mice $\rightarrow$ mouse, dice $\rightarrow$ die

Lemmatizing and Stemming

Two alternative options

Stemming: removing suffixes

swim, swimming, swimmer, swimmed $\rightarrow$ swim

Lemmatizing: reducing the words to their linguistic roots.

be, am, are, is $\rightarrow$ be

gave $\rightarrow$ give

feet $\rightarrow$ foot, teeth $\rightarrow$ tooth,

mice $\rightarrow$ mouse, dice $\rightarrow$ die

Stemming: Simpler and faster; impossible in some languages.
Lemmatizing: Slower but more accurate.

Probability Review

Fix distribution over probability space. Technicalities omitted.

$Pr(X)$: probability of event X

$Pr(Y|X) = Pr(X \cap Y)/Pr(X)$ = prob. of Y conditioned to X.

Probability Review

Fix distribution over probability space. Technicalities omitted.

$Pr(X)$: probability of event X

$Pr(Y|X) = Pr(X \cap Y)/Pr(X)$ = prob. of Y conditioned to X.

Bayes' Rule (prove it!):

$$Pr(X|Y) = \frac{Pr(Y|X) \cdot Pr(X)}{Pr(Y)}$$

Independence

X and Y are independent if

$$Pr(X \cap Y) = Pr(X) \cdot Pr(Y)$$

equivalently (prove it!) if

$$Pr(Y|X) = Pr(Y)$$

Expectation

$$E[X] = \sum_x (x \cdot Pr[X = x])$$

(In continuous spaces, change sum to integral.)

Major property: Linearity

- $E[X + Y] = E[X] + E[Y]$,
- $E[\alpha \cdot X] = \alpha \cdot E[X]$,

Expectation

$$E[X] = \sum_x (x \cdot Pr[X = x])$$

(In continuous spaces, change sum to integral.)

Major property: Linearity

- $E[X + Y] = E[X] + E[Y]$,
- $E[\alpha \cdot X] = \alpha \cdot E[X]$,
- and, more generally, $E[\sum_i \alpha_i \cdot X_i] = \sum_i (\alpha_i \cdot E[X_i])$.

Expectation

$$E[X] = \sum_x (x \cdot Pr[X = x])$$

(In continuous spaces, change sum to integral.)

Major property: Linearity

- $E[X + Y] = E[X] + E[Y]$,
- $E[\alpha \cdot X] = \alpha \cdot E[X]$,
- and, more generally, $E[\sum_i \alpha_i \cdot X_i] = \sum_i (\alpha_i \cdot E[X_i])$.
- Additionally, if X and Y are independent events, then $E[X \cdot Y] = E[X] \cdot E[Y]$.

Harmonic Series

And its relatives

The harmonic series is $\sum_i \frac{1}{i}$:

- It diverges:

  $$\lim_{N\to\infty} \sum_{i=1}^{N} \frac{1}{i} = \infty.$$

- Specifically, $\sum_{i=1}^{N} \frac{1}{i} \approx \gamma + \ln(N)$,
  where $\gamma \approx 0.5772\ldots$ is known as Euler's constant.

Harmonic Series

And its relatives

The harmonic series is $\sum_i \frac{1}{i}$:

- It diverges:

  $\lim_{N\to\infty} \sum_{i=1}^{N} \frac{1}{i} = \infty$.
- Specifically, $\sum_{i=1}^{N} \frac{1}{i} \approx \gamma + \ln(N)$,

  where $\gamma \approx 0.5772\ldots$ is known as Euler's constant.

However, for $\alpha > 1$, $\sum_i \frac{1}{i^\alpha}$ converges to Riemann's function $\zeta(\alpha)$

For example $\sum_i \frac{1}{i^2} = \zeta(2) = \frac{\pi^2}{6} \approx 1.6449\ldots$

How are texts constituted?

Obviously, some terms are very frequent and some are very infrequent.

How are texts constituted?

Obviously, some terms are very frequent and some are very infrequent.
Basic questions:

- How many different words do we use frequently?
- How much more frequent are frequent words?

How are texts constituted?

Obviously, some terms are very frequent and some are very infrequent.
Basic questions:

- How many different words do we use frequently?
- How much more frequent are frequent words?
- Can we formalize what we mean by all this?

There are quite precise empirical laws in most human languages.

Text Statistics

Heavy tails

In many natural and artificial phenomena, the probability distribution “decreases slowly” compared to Gaussians or exponentials.

This means: very infrequent objects have substantial weight in total.

Text Statistics

Heavy tails

In many natural and artificial phenomena, the probability distribution "decreases slowly" compared to Gaussians or exponentials.

This means: very infrequent objects have substantial weight in total.

- texts, where they were observed by Zipf;
- distribution of people's names;
- website popularity;
- wealth of individuals, companies, and countries;
- number of links to most popular web pages;
- earthquake intensity.

Text Statistics

The frequency of words in a text follows a powerlaw.
For (corpus-dependent) constants a, b, c

$$\text{Frequency of } i\text{-th most common word} \approx \frac{c}{(i+b)^a}$$

(Zipf-Mandelbrot equation).

Text Statistics

The frequency of words in a text follows a powerlaw.
For (corpus-dependent) constants a, b, c

$$\text{Frequency of } i\text{-th most common word} \approx \frac{c}{(i+b)^a}$$

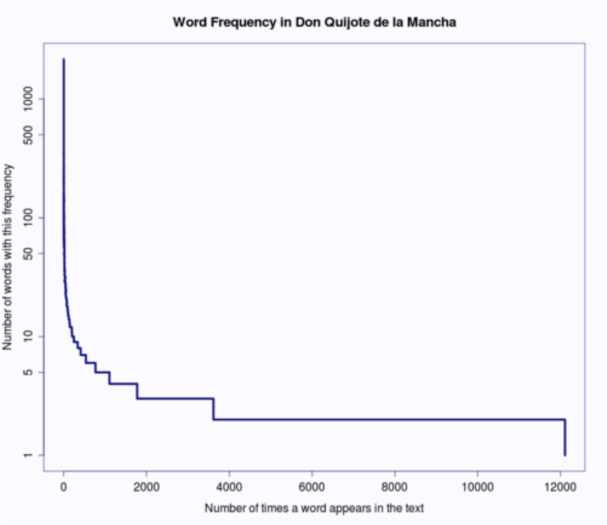
(Zipf-Mandelbrot equation).

Postulated by Zipf with $a = 1$ in the 30's.

$$\text{Frequency of } i\text{-th most common word} \approx \frac{c}{i^a}.$$

Further studies: a varies above and below 1.

Word Frequencies in Don Quijote



[https://www.r-bloggers.com/don-quijote-word-statistics/]

How to detect power laws?

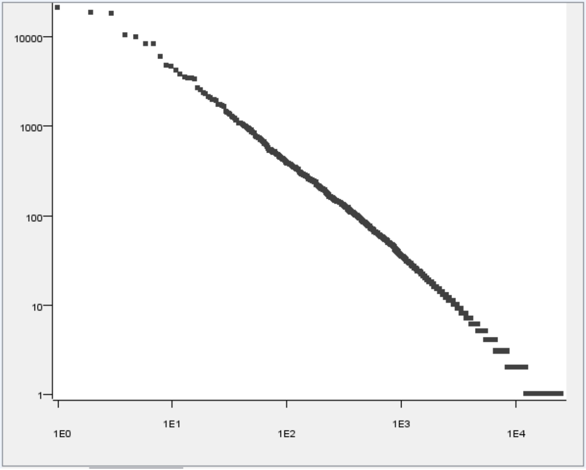
Try to estimate the exponent of an harmonic sequence.

- Sort the items by decreasing frequency.
- Plot them against their position in the sorted sequence (rank).
- Probably you do not see much until adjusting to get a log-log plot:

  That is, running both axes at log scale.
- Then you should see something close to a straight line.
- Beware the rounding to integer absolute frequencies.
- Use this plot to identify the exponent.

Text Statistics

Zipf's law in action



Word frequencies in Don Quijote (log-log scales).

Text Statistics

Amount of terms in use

Naturally, longer texts tend to use wider lexicon.

However,

the longer the text already seen, the lesser the chances of finding novel terms.

Naturally, longer texts tend to use wider lexicon.

However,

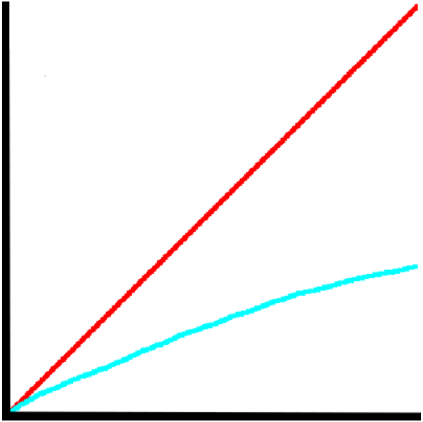
the longer the text already seen, the lesser the chances of finding novel terms.

- The first 2500 words of Don Quijote include slightly over 1100 different words.
- The total text of Don Quijote reaches about 383,000 words, but only less than 40,000 different ones.

Text Statistics

The first 2500 words of a random journal paper

(The blue line indicates number of different words.)



Text Statistics

Herdan's law, also known as Heaps' law

The number of different words

is described by a polynomial of degree less than 1.

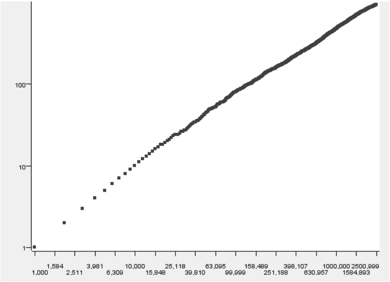
Text Statistics

Herdan's law, also known as Heaps' law

The number of different words

is described by a polynomial of degree less than 1.

Again this can be seen by resorting to log-log plots. The blue curve in the previous slide then becomes "more straight":



Text Statistics

Herdan's or Heaps' law

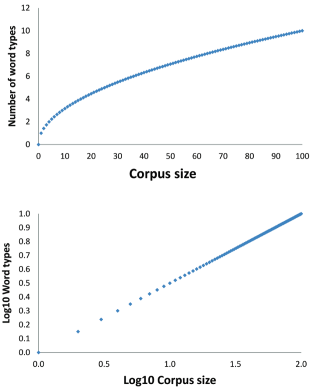


FIGURE 1 | Figures illustrating Herdan's or Heap's law. The **(Top)** figure shows how the number of word types would increase if the law were a power law with exponent 0.5 (i.e., square root). The **(Bottom)** figure shows the same information when the axes are $\log_{10}$ transformed. Then the concave function becomes a linear function, which is easier to work with.

From `https://www.frontiersin.org/articles/10.3389/fpsyg.2016.01116/full`

Text Statistics

Deriving the formula for Heaps' law

For a text of length N:

Say that we tend to find d words; how to relate d to N?

As a straight line in the log-log plot, we get:

$$\log d = k_1 + \beta \cdot \log N, \text{ that is, } d = k \cdot N^{\beta}$$

- The value of β varies with language and type of text.
- for Don Quijote, we find $\beta \approx 0.806$.
- In English, lower values of β, down to 0.5, are common.
- Finite vocabulary implies no further growth for very large N (but note: misspellings, proper names, foreign words…).