Introduction
Toolchain
Evaluation
Beyond the vector model
Conclusion

Fundamentals in Information Retrieval

Jean-Cédric CHAPPELIER
Emmanuel ECKARD

LIA

Information Retrieval

Definition

selection of documents relevant to a query in an unstructured collection of documents.

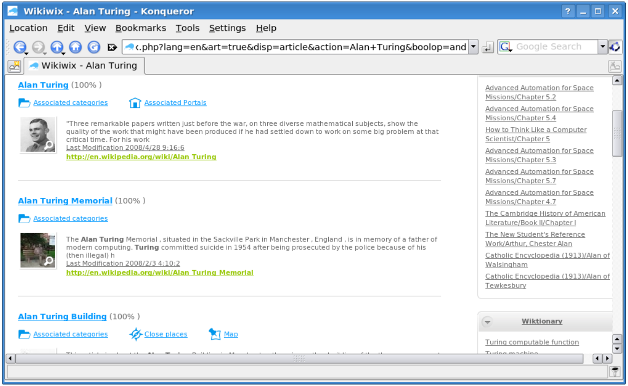
- **unstructured**: not produced with IR in mind, not a database.
- **document**: here, natural language text (but could also be video, audio or images)
- **query**: utterance in natural language (possibly augmented with commands, see later)
- **relevant**:
  1. users-wise: answering the IR requirements
  2. mathematically: maximising a defined “proximity measure”

Example of Information retrieval: issuing a query on an unstructured collection

- query ("Alan Turing")
- search among unstructured collection (Wikipedia articles)

Jean-Cédric Chappelier & Emmanuel Eckard
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Example of Information retrieval: results returned by the system



- list of results with a percentage match
- highest matches first

Ambiguity

Sometimes uninteded results occur

Example

query: "*Chicago school*"

wanted?

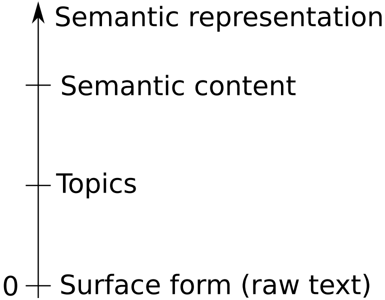
- schools in Chicago (IL)?
- body of works in sociology?
- architectural style?
- where to learn how to play Chicago (game):
  - bridge?
  - or pocker??

Relevance? Content versus topic

"*Relevant*" documents:

What does "*relevant*" mean?

- useful?
- new?
- topically related?
- content related?
  - at word level?
  - at semantic/pragmatic level?



Relevance? Content versus topic

Semantic content:
what the document talks about (topic) *vs* what it says (content).

Example

Document 1:

Note how misty the river banks are.

Document 2:

She got misty by the river of bank notes falling on the table.

Document 3:

Money had never interested her.

Doc. 1 & 2 have similar word content but are not topically related.
Doc. 2 & 3 have similar topics but opposite semantic content.

Introduction
Context and definitions
Simple example: Boolean model
Toolchain
Evaluation
Beyond the vector model
Conclusion

How it IR done?

Tasks

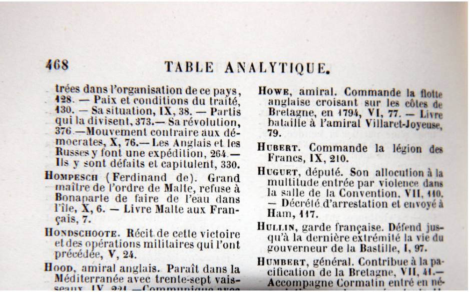
- have the computer represent documens (at the adequate level): preprocessing, indexing, ...
- represent the query, not necessarily the same way as documents (short queries, operators, ...)
- define satisfying relevance measures between representations

Similarities with other NLP tasks

- Classification (no query)
- Data mining (formatted data)
- Information extraction (retrieve *shorts parts* of documents)

IR Before computers

- Colophons on clay tablets of Mesopotamia (3500 BCE)
- Tags on scrolls of Edfu temple (from 237 BCE)
- Middle Age: indexes of key terms of the Bible
- Indexes for important texts: the Bible, Shakespeare's works, ...



468 TABLE ANALYTIQUE.

trées dans l'organisation de ce pays, 128. — Paix et conditions du traité, 130. — Sa situation, IX, 38. — Partis qui la divisent, 373.— Sa révolution, 376.—Mouvement contraire aux démocrates, X, 76.—Les Anglais et les Russes y font une expédition, 264.— Ils y sont défaits et capitulent, 330.

HOMPESCH (Ferdinand de). Grand maître de l'ordre de Malte, refuse à Bonaparte de faire de l'eau dans l'île, X, 6. — Livre Malte aux Français, 7.

HONDSCHOOTE. Récit de cette victoire et des opérations militaires qui l'ont précédée, V, 24.

HOOD, amiral anglais. Paraît dans la Méditerranée avec trente-sept vaisseaux, IV, 221.—Communique avec

HOWE, amiral. Commande la flotte anglaise croisant sur les côtes de Bretagne, en 1794, VI, 77. — Livre bataille à l'amiral Villaret-Joyeuse, 79.

HUBERT. Commande la légion des Francs, IX, 210.

HUGUET, député. Son allocation à la multitude entrée par violence dans la salle de la Convention, VII, 110. — Décrété d'arrestation et envoyé à Ham, 117.

HULLIN, garde française. Défend jusqu'à la dernière extrémité la vie du gouverneur de la Bastille, I, 97.

HUMBERT, général. Contribue à la pacification de la Bretagne, VII, 41.— Accompagne Cormatin entré en né-

Index of Thiers' *Histoire de la Révolution française*, 1854

Simple example: Boolean model

Boolean model

- Documents are sets of terms (presence/absence)
- Queries are boolean expressions on terms

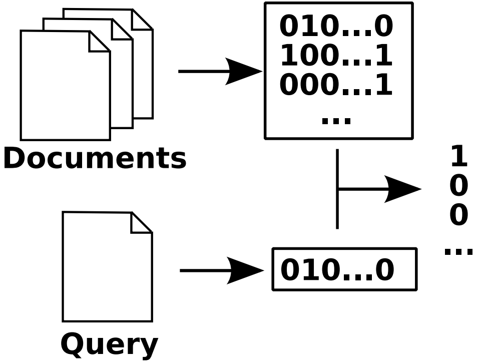
Steps

- V, a finite vocabulary of indexing terms
- R representation space
- $\mathscr{R}_D : V^* \to R$ representation function
- matching between query and documents

Example

- `feeling; ease; pain; feet; pain; ship`
- $\{0;1\}^{|V|}$
- presence/absence
- Boolean operators

Simple example: Boolean model



Introduction
Context and definitions
Simple example: Boolean model
Toolchain
Evaluation
Beyond the vector model
Conclusion

Example: Boolean representation of documents

Example

Document 1:

Come on, now,
I hear you're feeling down.
Well I can ease your pain
Get you on your feet again.

Document 2:

There is no pain you are receding
A distant ship, smoke on the horizon.

→ `Doc1: feeling; ease; pain; feet`
→ `Doc2: pain; ship; smoke; horizon`

Introduction
Context and definitions
Simple example: Boolean model
Toolchain
Evaluation
Beyond the vector model
Conclusion

Example: Boolean representation of queries; retrieval

Example

Query: `pain AND feeling`

`Doc1: feeling; ease; pain; feet`
`Doc2: pain; ship; smoke; horizon`

Results

- `Doc1` matches
- `Doc2` does not match

Limitations of the Boolean model

Example

Query: `pain AND feeling`

`Doc1: feeling; ease; pain; feet`
`Doc2: pain; ship; smoke; horizon`
→ `Doc1` matches; `Doc2` does not.

Limitations

- We might want to return `Doc2` as a second best choice. The boolean model does not allow this.
- What happens with "`pain OR feeling`"? ☞ does not match common layman wisdom

Indexing and represention of documents

Definition

Representation: translating a document (words) into computable data (numbers).
Indexing: selecting relevant elements (features) to support the representation

Themes related to indexing:

- Tokenisation
- Stop words
- Zipf and Luhn
- Stemming and lemmatisation
- Bag of words model

Tokenisation

Definition

Tokenisation: splitting the text into words (Pre-requisite to choosing indexing terms)

Example

- easy: whitespaces

  Now is the winter of our discontent
  Made glorious summer by this son of York

- less easy: space not always indicative of a term segmentation (compounds): *Distributional Semantics Information Retrieval and Latent Semantics Indexing performance comparison*
- agglutinative languages are a problem: *Rinderkennzeichnungs- und Rindfleischetikettierungsüberwachungsaufgabenübertragungsgesetz*
- Technical terms

Tokenisation of technical terms

e.g. in Chemistry

Methionyl-glutaminyl-arginyl-tyrosyl-glutamyl-seryl-leucyl-phenyl-alanyl-alanyl-glutaminyl-leucyl-lysyl-glutamyl-arginyl-lysy-glutamyl-gycyl-alanyl-phenyl-alanyl-valyl-prolyl-phenyl-alanyl-valyl-threonyl-leucyl-glycyl-aspartyl-prolyl-glycyl-isoleucyl-glutamyl-glutaminyl-seryl-leucyl-lysyl-isoleucyl-aspartyl-threonyl-leucyl-isoleucyl-glutamyl-alanyl-glycyl-alanyl-aspartyl-alanyl-leucyl-glutamyl-leucyl-glycyl-isoleucyl-prolyl-phenyl-alanyl-seryl-aspartyl-prolyl-leucyl-alanyl-aspartyl-glycyl-prolyl-threonyl-isoleucyl-glutaminyl-asparaginyl-alanyl-threonyl-leucyl-arginyl-alanyl-phenyl-alanyl-alanyl-alanyl-glycyl-valyl-threonyl-prolyl-alanyl-glutaminyl-cysteinyl-phenyl-alanyl-glutamyl-methionyl-leucyl-alanyl-leucyl-isoleucyl-arginyl-glutaminyl-lysyl-histidyl-prolyl-threonyl-isoleucyl-prolyl-isoleucyl-glycyl-leucyl-leucyl-methionyl-tyrosyl-alanyl-asparaginyl-leucyl-valyl-phenyl-...

Word Entities

Definition

Semantic entity: compound word (group of words) bearing a semantic meaning

Example

- “Information retrieval”
- “rendez-vous”
- “radio antenna”
- “Singing Lily” (a type of pastry)
- “Dolphin striker” (a spar [part of boat])

Conclusion on Tokenisation

Tokenisation is actually a NLP issue (use NLP techniques)

Choice of indexing terms

Filtering

Automated choice of indexing terms using filters:

- on morpho-syntactic categories (e.g.: prepositions have no semantic content; nouns do)
- on stop-words
- on frequencies

Jean-Cédric Chappelier
& Emmanuel Eckard
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Stop words

Definition

Stop word: term explicitely to be excluded from indexing.

Example

stoplist: `the; a; ’s; in; but; I; we; my; your; their; then`

> *Young men’s love then lies*
> *Not truly in their hearts, but in their eyes.*

Document: `Young men love lies truly hearts eyes`

Stop words

- Benefits:
  - more informative indexes
  - cheap way to remove classes of words without semantic content
  - smaller indexes (tractability)
- Problems:

  To be or not to be

  $\longrightarrow$ this sentence would be entirely stopped.

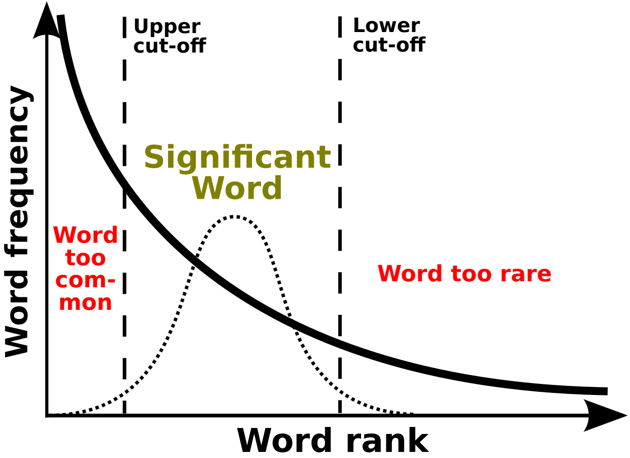
Choice of indexing terms: frequencies

Zipf and Luhn

If r is the rank of a term and n is its number of occurences (frequency) in the collection:

- Zipf (1949): $n \sim 1/r$
- Luhn (1958): mid-rank terms are the best indicators of topics

Choice of indexing terms: frequencies



Introduction
Toolchain
Indexing
Vector Space model
Queries
Evaluation
Beyond the vector model
Conclusion

Stemming and lemmatisation

Definition

Stem: morphological root of a word.
Stemming: Process of reducing words to their *stem*.

Example

- `prepaid, paid` $\longrightarrow$ `paid`
- `interesting, uninteresting` $\longrightarrow$ `interest`

Stemming and lemmatisation

Benefits

Reduces lexical variability $\Rightarrow$ reduces index size
increases information value of each indexing term.

Non-trivial process

`factual` $\longrightarrow$ `fact`	OK
`equal` $\longrightarrow$ `eq`	wrong ("eq" is too short)

Desequentialisation: bag of words model

Assumption

Positions of the terms are ignored. Term distribution is indicative enough of the meaning.

Model

$$d_1 = \{(t_1, n(d_1, t_1)); (t_2, n(d_1, t_2)); \ldots\}$$

$$d_2 = \{(t_1, n(d_2, t_1)); (t_2, n(d_2, t_2)); \ldots\}$$

A document is a multiset of terms

Example

Now so long, Marianne ; it's time that we began
to laugh and cry and cry ; and laugh about it all again.

$\rightarrow$ (`[begin,1] [cry,2] [laugh,2] [long,1] [Marianne,1] [time,1]`)

Introduction
Toolchain
Indexing
Vector Space model
Queries
Evaluation
Beyond the vector model
Conclusion

Phrases, neighbourhoods: beyond the words

Position could be kept to allow

- litteral search (quotations):

  `"more things in heaven and earth"`

- search by proximity:

  `dreamt WITHIN 5 philosophy`

Introduction
Toolchain
Indexing
Vector Space model
Queries
Evaluation
Beyond the vector model
Conclusion

Conclusions on indexing

- Bad indexing can ruin the performances of an otherwise sophisticated IR system
- Good indexing is anything but trivial

Jean-Cédric Chappelier
& Emmanuel Eckard
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Introduction
Toolchain
Indexing
Vector Space model
Queries
Evaluation
Beyond the vector model
Conclusion

Vector Space model

Objective

Overcome the limitations of the Boolean model by representing documents with vector describing term distributions.

Principle

- V, a finite vocabulary of indexing terms
- R **representation space**
- $\mathscr{R}_D : V^* \to R$ **representation function**
- **similarity:** $\mathscr{M}_{\text{prox}} : R \times R \to \mathbb{R}^+$

Note: choose similarity measure well behaved for the representation (depends on the representation)
☞ more in the “Textual Data Analysis” lecture

Vocabulary of indexing terms

Example

- *Now so long, Marianne*
  it's time that we began
  to laugh and cry and cry
  and laugh about it all again.
- V, a finite vocabulary: `aardvark, begin, cry, information, laugh, long, Marianne, retrieval, time, ...`

$\rightarrow$ Now so long Marianne it's time that we began to laugh and cry and cry and laugh about it all again.

In practice

the vocabulary is several thousands of terms large

Introduction
Toolchain
Indexing
Vector Space model
Queries
Evaluation
Beyond the vector model
Conclusion

Characterisation

Definition

characterisation: projection of the document into the representation space

Example

- *Now so long, Marianne*
  it's time that we began
  to laugh and cry and cry
  and laugh about it all again.
- R representation space: $\mathbb{R}^{|V|}$

$\rightarrow$ (`[aardvark,?]` `[begin,?]` `[cry,?]` `[information,?]` `[laugh,?]` `[long,?]` `[Marianne,?]` `[retrieval,?]` `[time,?]`)

Weightings

Term Frequency

$\mathrm{tf}(w_i, d_j) =$ nb of occurences of term w_i in document d_j

Sometimes $1 + \log(\mathrm{tf}(w_i, d_j))$ is used in place of $\mathrm{tf}(w_i, d_j)$

Term Frequency - Inverse Document Frequency

$$\text{tf-idf}(w_i, d_j) = \mathrm{tf}(w_i, d_j) \cdot \mathrm{idf}(w_i)$$

with

$$\mathrm{idf}(w_i) = \log\left(\frac{|D|}{nb(d_k \supset w_i)}\right)$$

$|D|$: number of documents
$nb(d_k \supset w_i)$: number of documents which contain term w_i

Weighting

Example

- *Now so long, Marianne*
  it's time that we began
  to laugh and cry and cry
  and laugh about it all again.
- $\mathscr{R}_D : V^* \rightarrow R$ representation function: here: Term Frequency

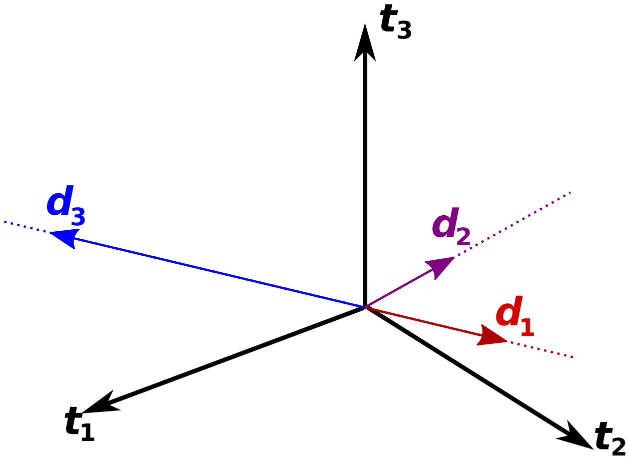
$\longrightarrow$ (`[aardvark,0] [begin,1] [cry,2] [information,0] [laugh,2] [long,1] [Marianne,1] [retrieval,0] [time,1]`)

$$\longrightarrow (0\ 1\ 2\ 0\ 2\ 1\ 1\ 0\ 1\ \ldots)$$

In practice

the vector is very sparse

Vector space model



- indexing terms define axis
- documents are point in the vector space (representing directions)

Proximity measure between documents

Cosine similarity

$$\cos(\mathbf{d_1},\mathbf{d_2}) = \frac{\mathbf{d_1}}{||\mathbf{d_1}||} \cdot \frac{\mathbf{d_2}}{||\mathbf{d_2}||} = \frac{\sum_{j=1}^{N} d_{1j}\, d_{2j}}{\sqrt{\left[\sum_j {d_{1j}}^2\right]\, \left[\sum_j {d_{2j}}^2\right]}}$$

- bounded ($0 < \cos(\mathbf{d_1},\mathbf{d_2}) < 1, \forall \mathbf{d_1},\mathbf{d_2}$)
- it is a similarity: the greater, the more similar the documents (as opposed to a *metric*)
- independent on the length of the document

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Proximity measure between documents

Document 1

- Now so long, Marianne, it's time that we began to laugh and cry and cry and laugh about it all again.
- `..., [long,1] [Marianne,1] [time,1] [begin,1] [laugh,2] [cry,2], ...`
- $\mathbf{d_1} = (\ldots, 1, 1, 1, 1, 2, 2, \ldots)$

Document 2

- I haven't seen Marianne laugthing for some time, is she crying all day long ?
- `..., [long,1] [Marianne,1] [time,1] [begin,0] [laugh,1] [cry,1], ...`
- $\mathbf{d_2} = (\ldots, 1, 1, 1, 0, 1, 1, \ldots)$

Example

$$\cos(\mathbf{d_1}, \mathbf{d_2}) = 7/(\sqrt{12} \cdot \sqrt{5}) = 0.904$$

Summary

Choices depending on the application

- **Weighting:** allows to translate semantic notions into computable models
- **Proximity measure:** fixes the topology of the representation space

Constants

- $|V|$-dimensional vector space
- very sparse vectors

Queries: definition

Definition

Queries (or "topics") are "questions" asked to the system

Typically *keywords*, possibly augmented with operators

```
dreamt WITHIN 5 philosophy
```

Supposed unknown at indexing time (difference between IR and classification or clustering)
Visit `http://www.google.com/trends` for real-life examples

Query representation

Example

- easy: as for documents

  `more things in heaven and earth`

- less easy (verbatim sentence)

  `"more things in heaven and earth"`

- quite different from the document (positional information)

  `dreamt WITHIN 5 philosophy`

Conclusion:

Query representation is not necessarly trivial (not always the same as representation of documents).

Problem of short queries

Web queries

On the web,

- the average query length is under three words
- very few users use operators

Language being ambiguous, three-word queries are difficult to satisfy.

Solutions

- *query expansion*: use knowledge about the query terms to associate them with other terms and improve the query.
- *query term reweighting*: weight the terms of the query as to obtain maximum retrieval performance.
- *relevance feedback*: User provides the system an evaluation of the relevance of its answers.

Evaluation campaigns

Evaluation set

1. Document collection
2. Query set
3. Referential

Definition

Referential: list of documents of a collection to be retrieved for one given query (handmade).

Examples of evaluation campains

- Smart (1970s)
- TREC (since the 1990s; large collections)
- AMARYLLIS (French)

Performances of IR systems

Reminder:
Given an IR system, a document collection, queries, referential and an answer by the system:

Definition

Precision is the proportion of the documents retrieved by the system that are relevant (according to the referential)

Definition

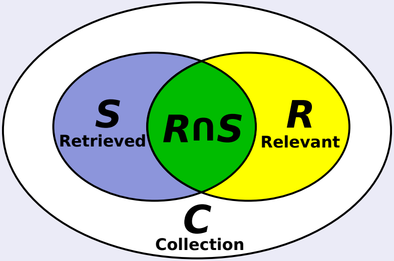
Recall is the proportion of the relevant documents which were retrieved by the system

- Precision can be cheated by returning no document
- Recall can be cheated by returning all documents

Performances of IR systems

Given an IR system, a document collection and a referential; for a query q, the results returned by the system is evaluated with:

- Precision: $\Pr(q) = \frac{|R(q) \cap S(q)|}{|S(q)|}$
- Recall: $\mathrm{Rec}(q) = \frac{|R(q) \cap S(q)|}{|R(q)|}$



Performance measures: R-Precision

Definition

Precision at n document:

$$\Pr_n(q) = \frac{|R(q) \cap S_n(q)|}{|S_n(q)|}$$

with $S_n(q) = n$ first documents to be retrieved

R-Precision

precision obtained after retrieving as many documents as there are relevant documents, averaged over queries

$$\text{R-Precision} = \frac{1}{N} \sum_{i=1}^{N} \Pr_{|R(q_i)|}(q_i)$$

Performance measures: Mean Average Precision

Average Precision

Average of the precisions whenever all relevant documents below rank $\mathrm{rk}(d,q)$ are retrieved:

$$\mathrm{AvgP}(q) = \frac{1}{|R(q)|} \sum_{d \in R(q)} \mathrm{Pr}_{\mathrm{rk}(d,q)}(q)$$

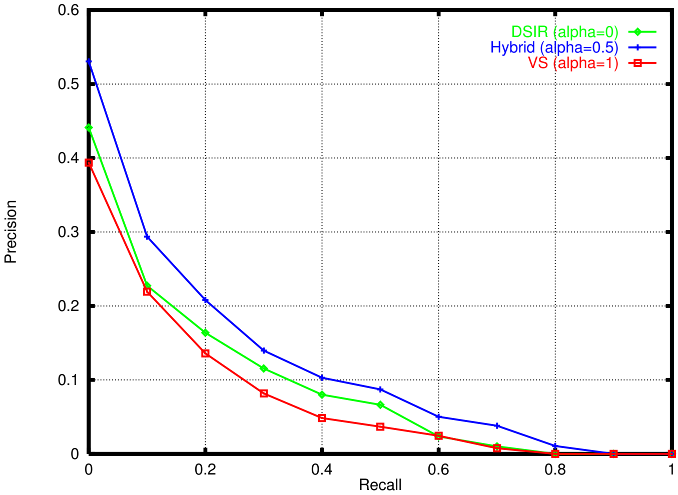
Mean Average Precision

Mean over the queries of the Average Precisions

$$\frac{1}{N} \sum_{i} \mathrm{AvgP}(q_i)$$

MAP measures the tendency of the system to retrieve relevant documents first.

Plotting average Precision and Recall



Aim of the game: push the curve towards the upper right corner

Introduction
Toolchain
Evaluation
Beyond the vector model
Probabilistic models
Topic-based models
LSI
DSIR
Conclusion

Probabilistic models

Idea

The best possible ranking returns documents sorted by probability to be relevent given a query.

for instance: Sparck-Jones' model

- Estimate the probability that a given document d_i is relevant ($d_i \in R(q)$) to given query q: $P(d_i \in R(q) | d_i, q)$
- Invert the probability (here R is a boolean variable, standing for $d_i \in R(q)$) : $P(d_i | R, q)$
- Write $P(d_i | R, q)$ as a function of the probabilities of occurence of the terms (assuming that terms are conditionally independant): $P(t_i \in d | R, q)$

Sparck-Jones' model

Document d contains term t_i (of the query)

$$w(t_i, d) = \log \frac{p(t_i \in d | d \in R)}{p(t_i \in d | d \notin R)}$$

Document d does not contain term t_i (of the query)

$$w(t_i, d) = \log \frac{p(t_i \notin d | d \in R)}{p(t_i \notin d | d \notin R)} = \log \frac{1 - p(t_i \in d | d \in R)}{1 - p(t_i \in d | d \notin R)}$$

Combining the two

$$\begin{aligned} w(t_i, d) &= \log \frac{p(t_i \in d | d \in R)}{p(t_i \in d | d \notin R)} - \log \frac{p(t_i \notin d | d \in R)]}{p(t_i \notin d | d \notin R)} \\ &= \log \frac{p(t_i \in d | d \in R)[1 - p(t_i \in d | d \notin R)]}{p(t_i \in d | d \notin R)[1 - p(t_i \in d | d \in R)]} \end{aligned}$$

Okapi BM25

Idea

Refine Sparck-Jones' model by including term frequencies

$$w = \log \frac{p(\mathrm{freq}(t,d) = \mathrm{tf} | d \in R) p(t \notin d | d \notin R)}{p(\mathrm{freq}(t,d) = \mathrm{tf} | d \notin R) p(t \notin d | d \in R)}$$

BM25 weight for term *i*

$$w_i^{\mathrm{BM25}} = \frac{\mathrm{tf}_i (k_1 + 1)}{k_1((1-b) + b\frac{dl}{avdl}) + \mathrm{tf}_i} \cdot \mathrm{idf}_i$$

with *dl* = document length
avfl = average document length

BM25 is a very good model and used as reference for comparison with new models

Introduction
Toolchain
Evaluation
Beyond the vector model
Probabilistic models
Topic-based models
LSI
DSIR
Conclusion

Introduction to topic-based models

Problem

Information retrieval has problems notably with

- Polymesy
- Synonymy

Polymesy

Example

Query includes term `Bank`
$\rightarrow$ Bank of England? Bank of fishes? Grand bank? Airplane bank?

Consequences

Negative impact on precision

Synonymy

Example

Query includes term `freedom`
$\rightarrow$ *liberty* will not be seen as relevant

Consequences

Negative impact on recall

Topic-based models

Idea

Apply a transformation to the representation space as to emphasise the most relevant features: index senses rather than mere words

Note

Stemming is already a step in this direction (less dependent on mere words)

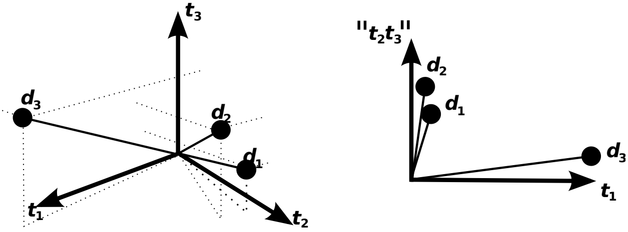
Reminder

Occurence matrix: term $\times$ document matrix containing the weights $\{w_{ij}\}$ associated to document d_i and term t_j

Latent Semantic Indexing

Idea

Reduction of dimensionality of the original representation space
Create a matrix close to the occurence matrix but of smaller rank



Before **After**

Latent Semantic Indexing

Reduction of dimensionality

- approximation of the occurence matrix
- filtering of the occurence matrix

Example

Singular Matrix Decomposition with k values (k between 100 to 300).

Latent Semantic Indexing

Illustration

$$\underset{(D \times T)}{[w_{ij}]} \rightarrow \underset{(D \times k)}{[\overline{w_{ij}}]}$$

$$\left.\begin{array}{l}\texttt{[flower]}\\ \texttt{[car]}\\ \texttt{[truck]}\end{array}\right\} \begin{array}{l}\texttt{[flower]}\\ \sqrt{12}\cdot\texttt{[car]} + \pi\cdot\texttt{[truck]} \quad (\cong \texttt{[vehicle]}\ ?)\end{array}$$

Latent Semantic Indexing

Advantages

- More significant representation

Drawbacks

- Out-performed by other models
- Too expensive to compute on large bases (requires iterative methods)
- Meaning of axis ??
- Query projection is problematic

Distributional Semantics Information Retrieval

Idea

There is a high degree of correlation between the observable distributional caracteristics of a term and its meaning:
"*a word is characterized by the company it keeps*";
Z. Harris (1954), J.R. Firth (1957)

Example

- Some *X*, for instance, naturally attack rats.
- The *X* on the roof was exposing its back to the shine of the sun.
- He heard the mewings of *X* in the forest .
- *X* is a: . . .

X is a ...

Bertil Videt, GFDL & CC-by-2.5

Co-occurence profile

Definition

Co-occurence profile: caracterisation of a word by its co-occurences with indexing terms

Example

Document 1

Now so long, Marianne,
it's time that we began
to laugh and cry and cry
and laugh about it all again.

Document 2

it seems so long ago,
Nancy was alone
looking at the Late Late show
through a semi-precious stone.

$\rightarrow$ Co-occurence profile of `long` = (`[cry,2] [begin,1] [Marianne,1] [Nancy,1] [time,1] [late,2] [laugh,2]`)

Co-occurence matrix

Definition

Co-occurence matrix: words $\times$ terms matrix of the co-occurence profiles with terms
f_{ij}: number of times that the word w_i and the indexing term t_j occur together.

DSIR Document representation

$$F_D = F_{\text{occurence}} \cdot F_{\text{co-occurence}}$$

$\rightarrow$ ponderation of the words in documents by the co-occurences

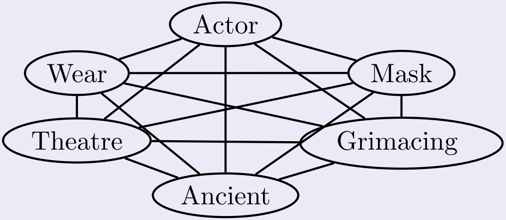
Note

When indexing a collection C, the co-occurence matrix would typically be evaluated on a control collection representative of the language/domain (could be C itself, but not necessarily)

Computing co-occurencies

The actor was wearing a grimacing mask of ancient theatre

```
actor
wear
grimacing
mask
ancient
theatre
```



Question

How often is a theatre grimacing?
$\rightarrow$ focus on relevant co-ocurrencies (Mask-Grimacing, Theatre-Ancient, ...)

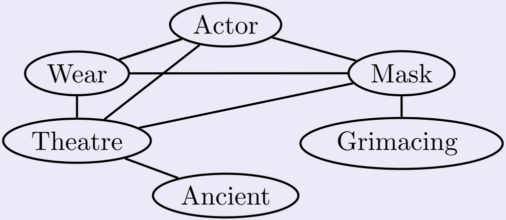
☞ Beyond mere co-occurencies: syntactic features

Co-occurencies and syntactic features

Use heads of phrases

(The actor) (was wearing) (a grimacing mask) (of ancient theatre)

```
⋆actor
⋆wear
grimacing ⋆mask
ancient ⋆theatre
```

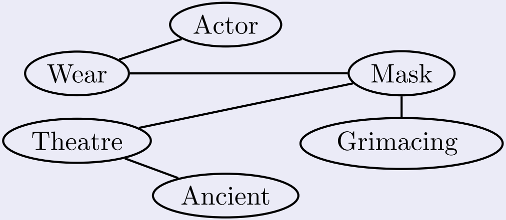


Co-occurencies augmented with Part-of-Speech

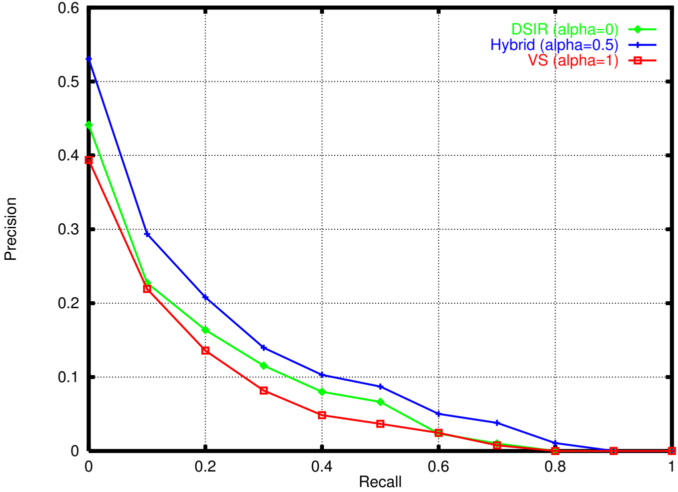
Using syntactic rules and semantic roles

The actor was wearing a grimacing mask of ancient theatre

```
SUBJ(actor,wear)
OBJ(wear,mask)
ADJ(mask,grimacing)
ADJ(theatre,ancient)
CNOUN(mask,theatre)
```



DSIR results



Other more advanced Topic Models

LDA: Latent Dirichlet Allocation (Blei, Ng, Jordan 2003)
(not to be confused with Linear discriminant analysis!!)

☞ probabilistic model with hidden states ("topics")

Reference:

- D. Blei. Probabilistic topic models. Communications of the ACM, 55(4):77–84, 2012.
- J.-C. Chappelier, Topic-based Generative Models for Text Information Access, In Textual Information Access – Statistical Models, E. Gaussier and F. Yvon eds, ch. 5, pp. 129-178, Wiley-ISTE, April 2012.

Summary / Keypoints

- Vector-space model;
- Indexing (and its important role);
- Weighting schemes, tf-idf;
- Evaluation: Precision and Recall.

References

[1] C. D. Manning, P. Raghavan and H. Schütze, "*Introduction to Information Retrieval*", Cambridge University Press. 2008.

[2] R. Baeza-Yates and B. Ribeiro-Neto, "*Modern Information Retrieval*", Addison Wesley, 1999.

[3] "*Topics in Information Retrieval*", chap. 15 in "Foundations of Statistical Natural Language Processing", C. D. Manning and H. Schütze, MIT Press, 1999.