

# Information retrieval

Evaluation of retrieval systems and learn to rank

Clovis Galiez

Laboratoire Jean Kuntzmann, Statistiques pour les sciences du Vivant et de l'Homme

October 7, 2020

# Objectives of the course

- Acquire a culture in information retrieval
- Master the basics concepts allowing to understand:
  - what is at stake in novel IR methods
  - what are the technical limits

This will allow you to have the basics tools to analyze current limitations or lacks, and imagine novel solutions.

# What's coming ahead (outline)

- Summary of last lectures
- Hands-on: program your own wikipedia search engine.

Next lecture:

- Feedback on the hands-on
- Machine learning in IR:
  - Embeddings
  - Evaluation of IR systems
  - Learning to search

# IR main steps



## Information retrieval

From Wikipedia, the free encyclopedia

**Information retrieval (IR)** is the activity of obtaining information system resources relevant to an information need from a collection of information resources. Searches can be based on full-text



# The tf-idf matrix

# The tf-idf matrix

## Definition

The matrix  $M$  which rows – corresponding to each document – are:

$$D_t = \frac{\# t \text{ in } D}{\# \text{ tokens in } D} \times I(t)$$

is called the **tf-idf** (term frequency-inverse document frequency) representation.

## Question

What are the advantages of the vector model ?

# The tf-idf matrix

## Definition

The matrix  $M$  which rows – corresponding to each document – are:

$$D_t = \frac{\# t \text{ in } D}{\# \text{ tokens in } D} \times I(t)$$

is called the **tf-idf** (term frequency-inverse document frequency) representation.

## Question

What are the advantages of the vector model ?

- Have a direct weightening by information carried by tokens
- Framework for latent semantics

# Latent semantics: low rank approximation

# Latent semantics: low rank approximation

## Theorem

Let  $M$  be the tf matrix:  $M_{ij}$  is the frequency of token  $j$  in document  $i$ .  $M^\top M$  is symmetric and its eigenvectors are orthogonal and form a basis of the token space.

# Latent semantics: low rank approximation

## Theorem

Let  $M$  be the tf matrix:  $M_{ij}$  is the frequency of token  $j$  in document  $i$ .  $M^\top M$  is symmetric and its eigenvectors are orthogonal and form a basis of the token space.

## Question

What are the eigenvectors of  $M^\top M$ ? What do they represent?

# Latent semantics: low rank approximation

## Theorem

Let  $M$  be the tf matrix:  $M_{ij}$  is the frequency of token  $j$  in document  $i$ .  $M^\top M$  is symmetric and its eigenvectors are orthogonal and form a basis of the token space.

## Question

What are the eigenvectors of  $M^\top M$ ? What do they represent?

## Question

What IR latent semantics tackles?

# PageRank

## Question

- What PageRank is good for?
- What data is used as input?
- How does it work?

# PageRank

## Question

What PageRank is good for?

What data is used as input?

How does it work?

**Data:**  $A :=$  graph of the WWW     $A_{ij} = \begin{cases} \frac{1}{N_j} & \text{if link from } j \text{ to } i \\ 0 & \text{else} \end{cases}$

**Result:** Ranking of web pages

$R_0 := S$  ;

**repeat**

$R^{(i+1)} \leftarrow A R^{(i)}$   
 $\delta \leftarrow \|R^{(i)} - R^{(i+1)}\|_1$

**until**  $\delta \leq \epsilon$ ;

**Algorithm 2:** simplified PageRank

# Machine learning in IR

## From (linear) latent semantics to embeddings

In latent semantics, we define base vectors of the vector space of token frequencies that represent

## From (linear) latent semantics to embeddings

In latent semantics, we define base vectors of the vector space of token frequencies that represent **concepts**.

## From (linear) latent semantics to embeddings

In latent semantics, we define base vectors of the vector space of token frequencies that represent **concepts**.

We represent documents by projecting their

## From (linear) latent semantics to embeddings

In latent semantics, we define base vectors of the vector space of token frequencies that represent **concepts**.

We represent documents by projecting their frequency vector in the low dimensional space formed by the important

## From (linear) latent semantics to embeddings

In latent semantics, we define base vectors of the vector space of token frequencies that represent **concepts**.

We represent documents by projecting their frequency vector in the low dimensional space formed by the important **eigen vectors**.

# From (linear) latent semantics to embeddings

In latent semantics, we define base vectors of the vector space of token frequencies that represent **concepts**.

We represent documents by projecting their frequency vector in the low dimensional space formed by the important **eigen vectors**.

Recent techniques (well, mostly since 2013)

Machine learning techniques can be used to **learn better vector representation<sup>a</sup> of tokens**, and more generally of any data (document, sentence, word, image, etc.).

---

<sup>a</sup>aka embeddings

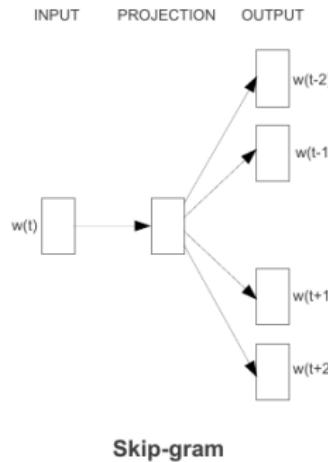
# Embeddings: a general technique with many derivatives

Many models have been developed for representing various type of data.  
Here is a small list of freely available models:

Model	Data represented
word2vec	Tokens
GloVe	Tokens
fastText	Tokens
doc2vec	Documents
dna2vec	Genomic sequences

## Word2vec: predict the context of a token

The core idea of word2vec is to learn a vector representation allows to predict the context of the token. Thereby, tokens appearing in similar context will be encoded closely in the vector space.



[Mikolov, Tomas; et al. (2013)]

# word2vec's latent semantics

The word2vec embeddings have interesting semantic features<sup>1</sup>.

Table 8: Examples of the word pair relationships, using the best word vectors from Table 4 (Skip-gram model trained on 783M words with 300 dimensionality).

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

<sup>1</sup>Note that GloVe is better at this

# Machine learning and IR performance evaluations

# Evaluation of IR systems

How to evaluate the performances of an IR system?

# Evaluation of IR systems

How to evaluate the performances of an IR system?

Need a gold standard  and indicators .

# Evaluation of IR systems

How to evaluate the performances of an IR system?

Need a gold standard  and indicators .

What is a gold standard?

It depends...

# Evaluation of IR systems

How to evaluate the performances of an IR system?

Need a gold standard  and indicators .

What is a gold standard?

It depends...

For measuring correctness: **Collection of documents** associated to a query.

# Evaluation of IR systems

How to evaluate the performances of an IR system?

Need a gold standard  and indicators .

What is a gold standard?

It depends...

For measuring correctness: **Collection of documents** associated to a query.

For measuring relevance: **Collection of documents** ranked by relevance associated to a query.

# Evaluation of IR systems

How to evaluate the performances of an IR system?

Need a gold standard  and indicators .

What is a gold standard?

It depends...

For measuring correctness: **Collection of documents** associated to a query.

For measuring relevance: **Collection of documents** ranked by relevance associated to a query.

Can be seen as a partial function  $g : Q \times D \rightarrow \mathbb{R}$  associating to a couple query-document its quantification of *correctness*, *relevance* or *truth*.

What can be an indicator?

# Indicators for binary gold standards<sup>2</sup>

Special case:  $\text{dom}(g) = \{0, 1\}$ .

## Question

What is a False Positive? True Negative?

---

<sup>2</sup>To be used for assessing correctness for instance.

# Indicators for binary gold standards<sup>2</sup>

Special case:  $\text{dom}(g) = \{0, 1\}$ .

## Question

What is a False Positive? True Negative?

What is a good graphic to have an idea of the performance of a ranking system?

---

<sup>2</sup>To be used for assessing correctness for instance.

# Indicators for binary gold standards<sup>2</sup>

Special case:  $\text{dom}(g) = \{0, 1\}$ .

## Question

What is a False Positive? True Negative?

What is a good graphic to have an idea of the performance of a ranking system?

Query:  $q = \text{results election U.S.}$

Document ( $d$ )	rank	$g(q,d)$
Biology-Wikipedia	0.82	0
laposte.fr	0.01	0
election.gov.us/results	0.9	1
mediapart.fr/America	0.7	1

<sup>2</sup>To be used for assessing correctness for instance.

# Indicators for binary gold standards<sup>2</sup>

Special case:  $\text{dom}(g) = \{0, 1\}$ .

## Question

What is a False Positive? True Negative?

What is a good graphic to have an idea of the performance of a ranking system?

Query:  $q = \text{results election U.S.}$

Document ( $d$ )	rank	$g(q,d)$
Biology-Wikipedia	0.82	0
laposte.fr	0.01	0
election.gov.us/results	0.9	1
mediapart.fr/America	0.7	1

<sup>2</sup>To be used for assessing correctness for instance.

# Indicators for binary gold standards<sup>2</sup>

Special case:  $\text{dom}(g) = \{0, 1\}$ .

## Question

What is a False Positive? True Negative?

What is a good graphic to have an idea of the performance of a ranking system?

Query:  $q = \text{results election U.S.}$

Document ( $d$ )	rank	$g(q,d)$
election.gov.us/results	0.9	1
Biology-Wikipedia	0.82	0
mediapart.fr/America	0.7	1
laposte.fr	0.01	0

<sup>2</sup>To be used for assessing correctness for instance.

# Indicators for binary gold standards<sup>2</sup>

Special case:  $\text{dom}(g) = \{0, 1\}$ .

## Question

What is a False Positive? True Negative?

What is a good graphic to have an idea of the performance of a ranking system?

Query:  $q = \text{results election U.S.}$

Document ( $d$ )	rank	$g(q,d)$
Biology-Wikipedia	0.82	0
laposte.fr	0.01	0
election.gov.us/results	0.9	1
mediapart.fr/America	0.7	1

<sup>2</sup>To be used for assessing correctness for instance.

# Indicators for binary gold standards<sup>2</sup>

Special case:  $\text{dom}(g) = \{0, 1\}$ .

## Question

What is a False Positive? True Negative?

What is a good graphic to have an idea of the performance of a ranking system?

Query:  $q = \text{results election U.S.}$

Document ( $d$ )	rank	$g(q,d)$
election.gov.us/results	0.9	1
Biology-Wikipedia	0.82	0
mediapart.fr/America	0.7	1
laposte.fr	0.01	0

<sup>2</sup>To be used for assessing correctness for instance.

# Indicators for binary gold standards<sup>2</sup>

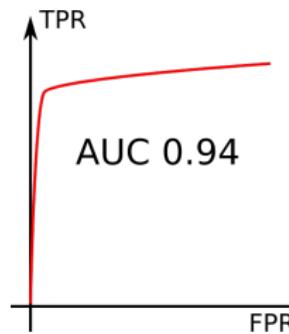
Special case:  $\text{dom}(g) = \{0, 1\}$ .

## Question

What is a False Positive? True Negative?

What is a good graphic to have an idea of the performance of a ranking system?

Query:  $q = \text{results election U.S.}$



AUC 0.94

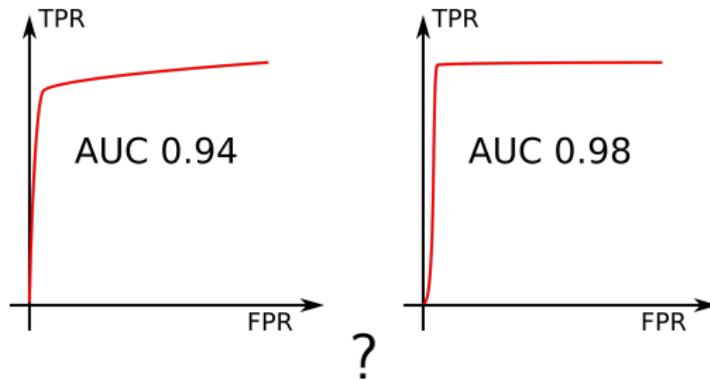
$$TPR_\tau = \frac{TP_\tau}{P}, FPR_\tau = \frac{FP_\tau}{N}$$

<sup>2</sup>To be used for assessing correctness for instance.

# Beware of summary indicators!



Use the right summary indicator (e.g. AUC-ROC/AUC-ROC5).



## Indicators for rank gold standards<sup>3</sup>

Ranking functions  $f_1, f_2 : \mathcal{Q} \times \mathcal{D} \rightarrow \mathbb{R}_+^*$  (the higher, the more relevant the doc to the query).

How can we say that  $f_1$  is better than  $f_2$ ?

---

<sup>3</sup>To be used for evaluation of relevance ranking for instance.

## Indicators for rank gold standards<sup>3</sup>

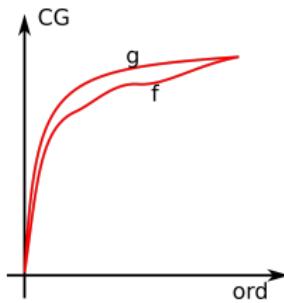
Ranking functions  $f_1, f_2 : \mathcal{Q} \times \mathcal{D} \rightarrow \mathbb{R}_+^*$  (the higher, the more relevant the doc to the query).

How can we say that  $f_1$  is better than  $f_2$ ?

Given a gold standard  $g : \mathcal{Q} \times \mathcal{D} \rightarrow \mathbb{R}_+^*$ , we define the cumulative gain:

$$\text{CG}_n(q, f) = \sum_{k:\text{ord}_n(q,f)} g(q, k)$$

where  $\text{ord}_n(q, f)$  are the  $n$  first elements of  $\mathcal{D}$  when sorting by  $f(q, -)$ .



<sup>3</sup>To be used for evaluation of relevance ranking for instance.

## Indicators for ranking, stressing first results

In the same spirit of the difference AUC/AUC5, one can stress more the first results, by weighting the relevance by a *discount*<sup>4</sup> function:

$$\text{DCG}(q, f) = \sum_{k: \text{ord}_{N_q}(q, f)} \frac{g(q, k)}{\log(i+1)}$$

where  $N_Q$  is  $\text{card}\{d | (q, d) \in \text{dom}(g)\}$  and  $i$  is the index in the summation.

The normalized DCG is defined as:

$$\text{NDCG}(q, f) = \frac{\text{DCG}_{N_Q}(q, f)}{\text{DCG}_{N_Q}(q, g)}$$

---

<sup>4</sup>One can choose different discount functions, but  $\frac{1}{\log i}$  has nice theoretical foundations [Wang et al. JMLR 13] and is good in practice.

## Comparing ranking strategies

Ranking functions  $f_1, f_2 : \mathcal{Q} \times \mathcal{D} \rightarrow \mathbb{R}_+^*$ . How can we say that  $f_1$  is better than  $f_2$ ?

Can use the expected NDCG( $q, f_i$ ) summary statistic:

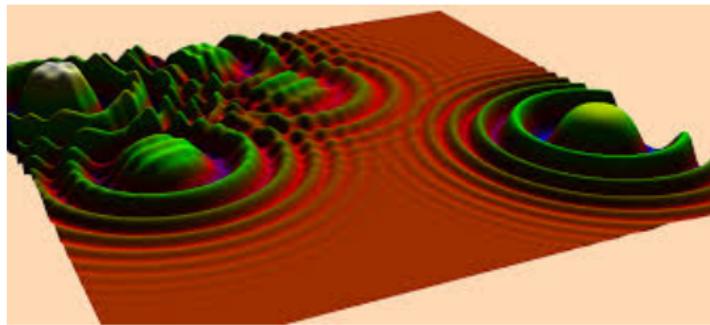
$$\rho_i = \frac{1}{Q} \sum_q \text{NDCG}(q, f_i)$$

As usual, averaging can hide bad performance when the gold standard is dominated by high performance on many similar queries.



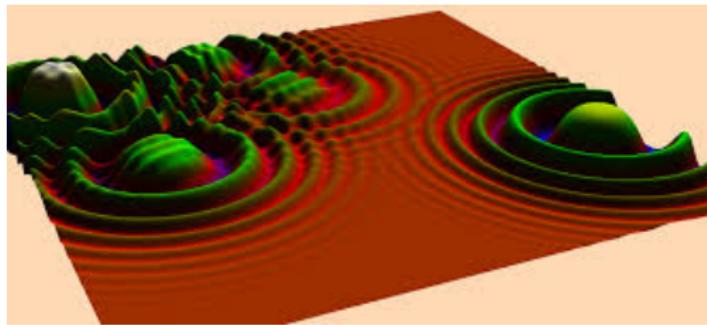
## Learning the parameters

Having a gold standard not only allows evaluation, but also optimization of the parameters.



## Learning the parameters

Having a gold standard not only allows evaluation, but also optimization of the parameters.



What parameters are we talking about?

## Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)

## Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming

## Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal

## Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal
  - too much: police - policy

## Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal
  - too much: police - policy
- Important text scoring

# Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal
  - too much: police - policy
- Important text scoring
- The scoring system (cosine similarity, Euclidean distance, etc.)

# Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal
  - too much: police - policy
- Important text scoring
- The scoring system (cosine similarity, Euclidean distance, etc.)
- Latent semantics (e.g. size of the space)

# Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal
  - too much: police - policy
- Important text scoring
- The scoring system (cosine similarity, Euclidean distance, etc.)
- Latent semantics (e.g. size of the space)

At the ranking level:

# Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal
  - too much: police - policy
- Important text scoring
- The scoring system (cosine similarity, Euclidean distance, etc.)
- Latent semantics (e.g. size of the space)

At the ranking level:

- Source vector choice

# Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal
  - too much: police - policy
- Important text scoring
- The scoring system (cosine similarity, Euclidean distance, etc.)
- Latent semantics (e.g. size of the space)

At the ranking level:

- Source vector choice
- Prior distribution on links in a web page (e.g. important links)

## Some parameters to tune

At the semantics level:

- Tokenization (parameters in *phrase as tokens* cf. patent in 1st tutorial session)
- Stemming
  - not enough: mother - maternal
  - too much: police - policy
- Important text scoring
- The scoring system (cosine similarity, Euclidean distance, etc.)
- Latent semantics (e.g. size of the space)

At the ranking level:

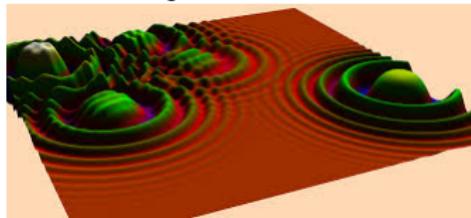
- Source vector choice
- Prior distribution on links in a web page (e.g. important links)

But also...

...the **trade-off** between the semantic scores (e.g. tf-idf vector model, text importance score) and the authority ranking score.

# Optimize the objective function by tuning the parameters

$$\text{obj} : \mathbb{R}^p \rightarrow \mathbb{R}$$



This is an optimization problem:

$$\arg \max_{\mathbb{R}^p} \text{obj}$$

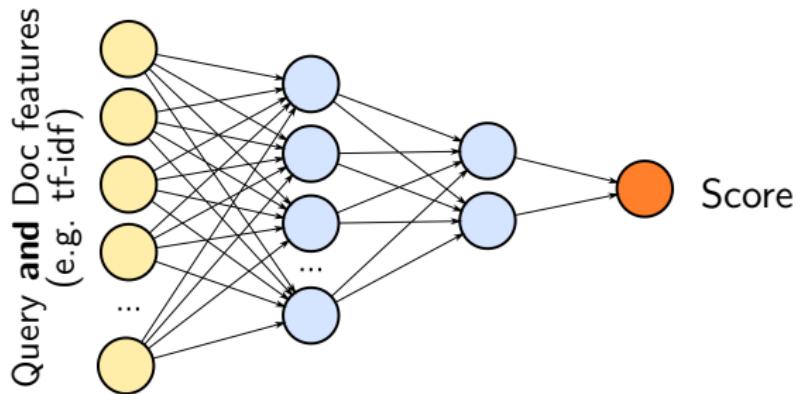
One can therefore use optimization strategies to maximize performance indicators by tuning  $p$  parameters.

## Why not going further?

Why learning only few parameters?

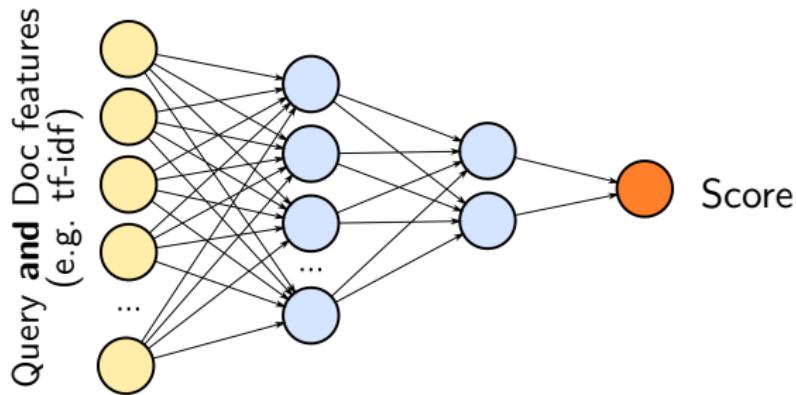
# Why not going further?

Why learning only few parameters? Why not learning directly:



## Why not going further?

Why learning only few parameters? Why not learning directly:

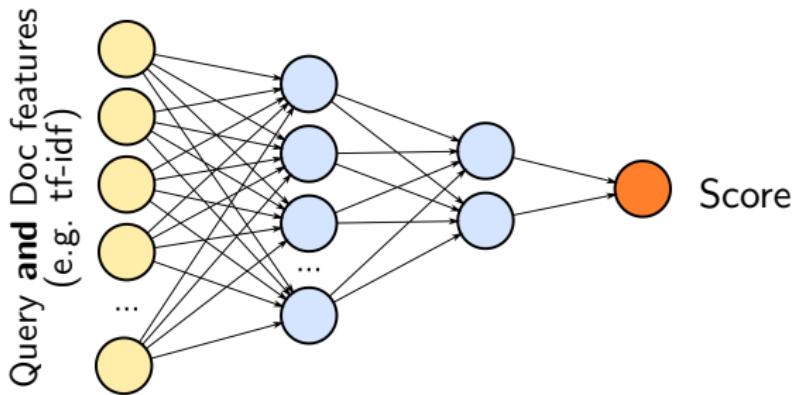


### Issue

Curse of dimensionality, training set limitation, overtraining.

# Why not going further?

Why learning only few parameters? Why not learning directly:



## Issue

Curse of dimensionality, training set limitation, overtraining.

## Way out

**Decrease the number of features or/and expand the training set.**

## Increasing the training set

If you own a popular search engine, imagine a simple way of increasing your training set.

## Increasing the training set

If you own a popular search engine, imagine a simple way of increasing your training set.

Click-through strategy:  $g : (q, d) \mapsto \text{nb of clicks on } d$

# Reducing the dimensionality

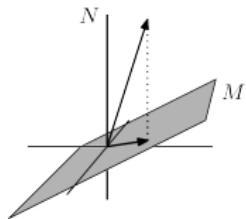
Remember

What dimensionality reduction technique have we seen before?

# Reducing the dimensionality

Remember

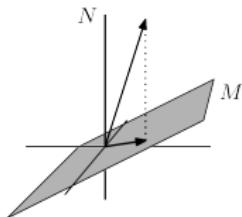
What dimensionality reduction technique have we seen before?



# Reducing the dimensionality

## Remember

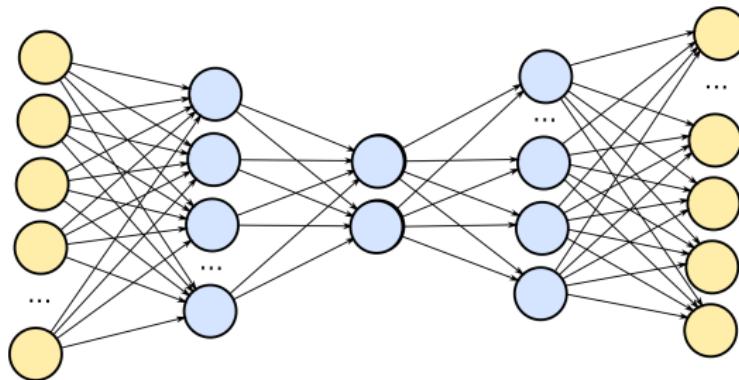
What dimensionality reduction technique have we seen before?



Note that for reducing the dimension we do not lack of data! The Internet is big enough :)

→ More powerful techniques such as autoencoders.

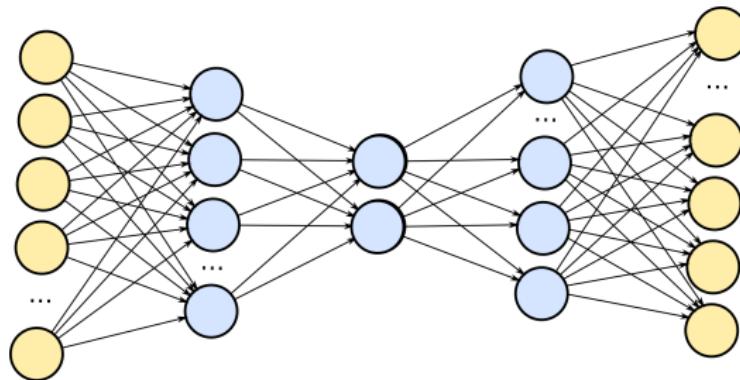
# Autoencoders



If you were an autoencoder...

If you were an autoencoder with **few** intermediate neurons, how would you encode a document?

# Autoencoders



If you were an autoencoder...

If you were an autoencoder with **few** intermediate neurons, how would you encode a document?

If one wants to minimize the loss between the input and the output, a neuron of the intermediate layer represents a topic, or a concept.

# Wrap-up |



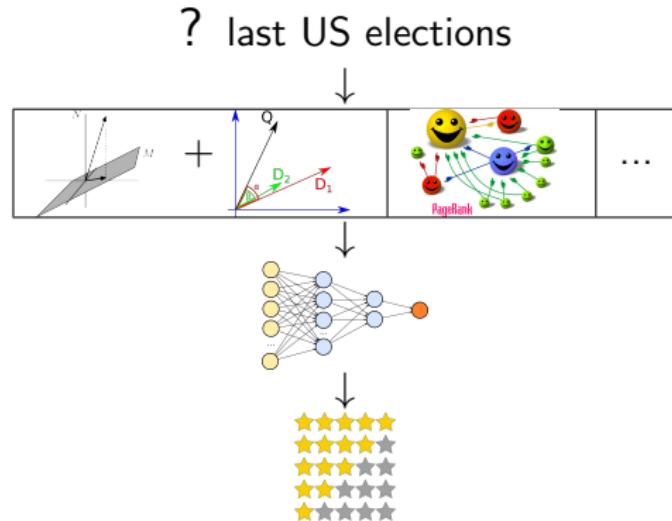
## Information retrieval

From Wikipedia, the free encyclopedia

**Information retrieval (IR)** is the activity of obtaining **information system** resources relevant to an information need from a collection of information resources. Searches can be based on **full-text**



## Wrap-up II



# Hope you enjoyed.

Find the material on <http://clovisg.github.io>

# Extras