INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

Text Cleaning & Vectorisation

IA Frameworks

Brendan Guillouet - 02/11/2020

Institut National des Sciences Appliquées de Toulouse

Google Cloud Platform

ML Python Libraries

scikit learn
Keras
gensim
surprise
Gym
TensorFlow
NLTK

Python Environment

PC
jupyter
ANACONDA

Viz' Python Libraries

matplotlib
Version 3.1.1
seaborn
plotly

Framework & Tool

docker
Google Cloud Platform
git

Table of contents

Introduction

Text use case in Artificial intelligence

There are multiple applications of artificial intelligence on text data:

- **Information retrieval**: on text or content-based *(Google, Yahoo etc.)*
- **Pattern recognition**: Information/Named extraction.
- **Sentiment analysis**: Marketing. Website comments.
- **Text generation**: Chatbot. Newspaper article.
- **Text translation**: Google Translate. DeepL.
- **Disambiguation**: Security.
- And many others…

Text processing does not always mean Natural Language Processing (NLP)

Example: Text classification

Objective: Automate the categorisation of text product within discount website.
Data come from datascience contest **website**.

Difficulties:

- Text data require processing to used machine learning model on it.
- Big amount of data (15M of text description).
- Highly unbalanced classes.
- High number of classes (more than 5000).
- Real dataset that requires a lot of cleaning.

Data

Train file contains 15.786.885 products. Answer of test file not furnished.

Three levels classification:

- 47 categories of level 1.
- 536 categories of level 2.
- 5789 categories of level 3.

Field	Type	Description
product id	String	Unique identifiant du produit
Catégorie 1	String	Catégorie de niveau 1
Catégorie 2	String	Catégorie de niveau 2
Catégorie 3	String	Catégorie de niveau 3
Description	String	Description produit
Libelle	String	Description courte
Marque	String	Marque du produit

Data example

Categorie1	
ANIMALERIE - NEW	Lit Mijou, 48 × 37 Pouces, Crème - -imitation ...
ARME DE COMBAT - ARME DE SPORT	Réplique chargeur STI DUTY ONE (CPG1945) - Rép...
ART DE LA TABLE - ARTICLES CULINAIRES	Mugs Alchemy (king 13) (Taille unique) - Mugs...
ARTICLES POUR FUMEUR	E-PACK FRUITÉ 'EXPERT' (Titanium bleu - Mixte ...
AUTO - MOTO (NEW)	Tube de fourche Tarozzi KYMCO X-CITING 500 - 0...
BAGAGERIE	portefeuille porte cartes billets compagnon fe...
BATEAU MOTEUR - VOILIER	Echelle pour plateforme70086 - Fabrication i...
BIJOUX - LUNETTES - MONTRES	Seiko SFP599 Hommes Montre - Acheter Authentiq...
BRICOLAGE - OUTILLAGE - QUINCAILLERIE	Clé polygonale double contre-coudée - 20x22 - ...
CHAUSSURES - ACCESSOIRES	Bottes bi-matière à talons bleu - Zaza Pata -...
CONDITIONNEMENT	EMBALLAGE Ruban adhésif d'emballage PVC colle ...
CULTURE / JEUX	De Keenen Ivory Wayans avec Shannon Elizabeth,...
DECO - LINGE - LUMINAIRE	Cars Poster Reproduction Sur Toile, Tendue Sur...
DROGUERIE (NEW)	PERCHE TELESCOPIQUE SECURITY LOCK 3X2M - PERCH...
ELECTROMENAGER	Filtre metal antigraisse (x1) AD546BE11 AD546W...
ELECTRONIQUE	Bloc de jonction à fusible Contenu: 20 pc(s) p...
EPICERIE	Sel de Guérande aux épices, Verrine 150 gr - S...
HYGIENE - BEAUTE - PARFUM	Uriage AquaPRÉCIS Crème Confort 40 ml - Les mi...
INFORMATIQUE	Batterie Acer Aspire One 751H-52Yr - Li-Ion 11...

Difficulties

- Noise linked to spelling, grammar, conjugaison mistake.
- Non significant terms.
- Mining of terms depends of context. *(Clothes / Dolls' clothes)*
- Different from one language to another.
- Transcription to machine learning.

Implies a lot of cleaning.

Text Cleaning

Stemming

Problem: Term can be written in different way (accentuation, genre, conjugaison, plurals, etc.) but still have the same meaning.

Solution: replace word with their stems.

Example:

- Épée, épee, épées, épée = epe
- vert, verts, vertes, vertes = vert
- mange, manger, mangez, mangent, = mang

Algorithm that generate steming from words are rules-based and depends of the language.

The one used on nltk for French language is the Snowball algorithm.

Stopwords

Problem: Terms that are very common does not help to classify data and can even disturb the training.

Solution: Most common words are removed. This words are called stopword.

Example:

- **French:** ‘au’, 'aux', 'avec', 'ce', 'ces', 'dans', 'de', 'des', 'du', 'elle', 'en', 'et', 'eux', 'il', 'ils', 'je', 'la', 'le', 'les', 'leur', ‘lui', 'ma', 'mais', 'me', 'même', 'mes', 'moi', ‘mon’, etc.

- **English:** ['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', ‘him’, etc.

Other clean steps

- Remove punctuation, number or other non-letter symbol.
- Increment stopwords list with domain words.
- Removed technical noise (HTML code).
- Lower case.

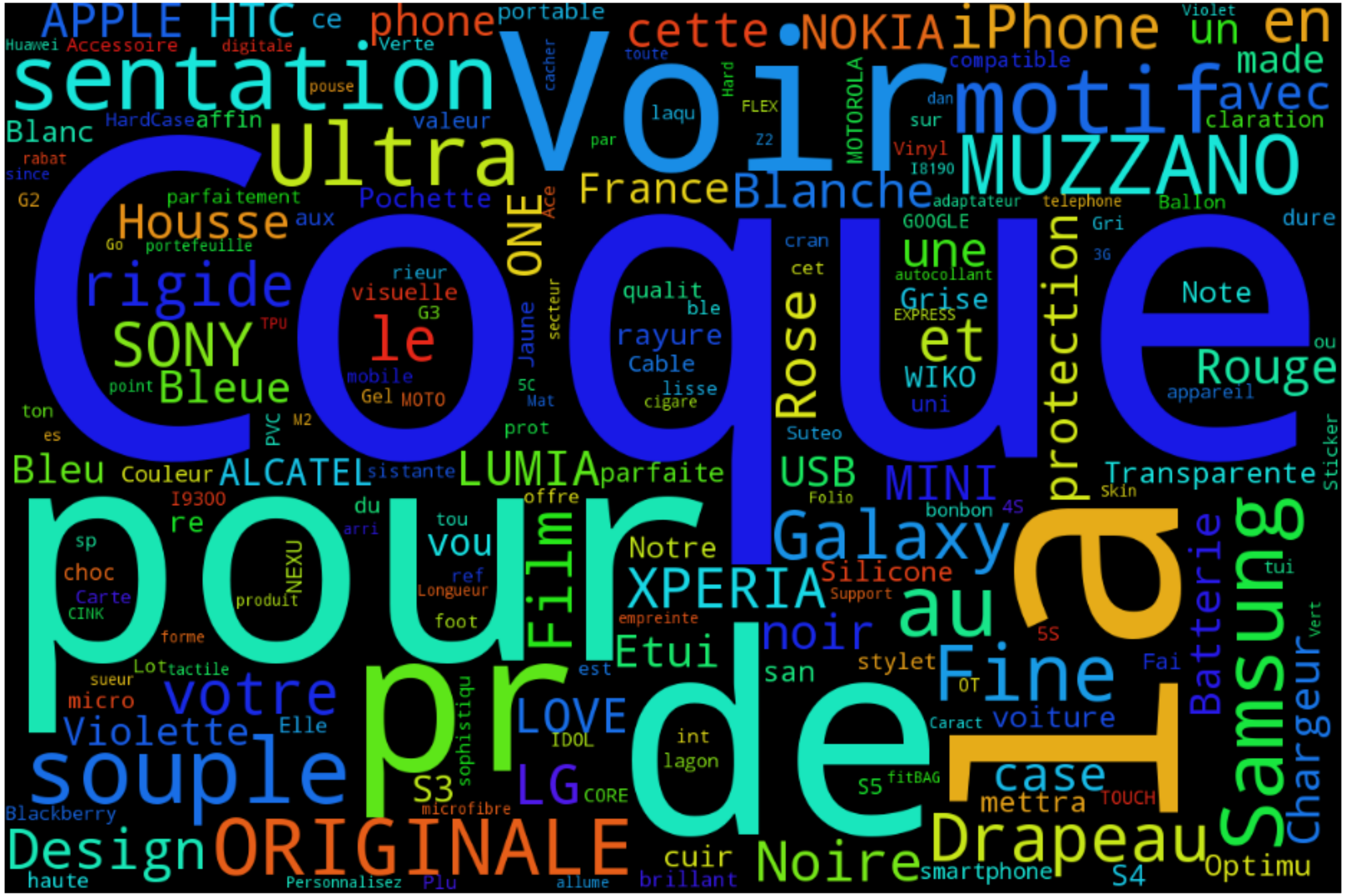
Most of these steps depend of the objectives you want to achieve.

Example:

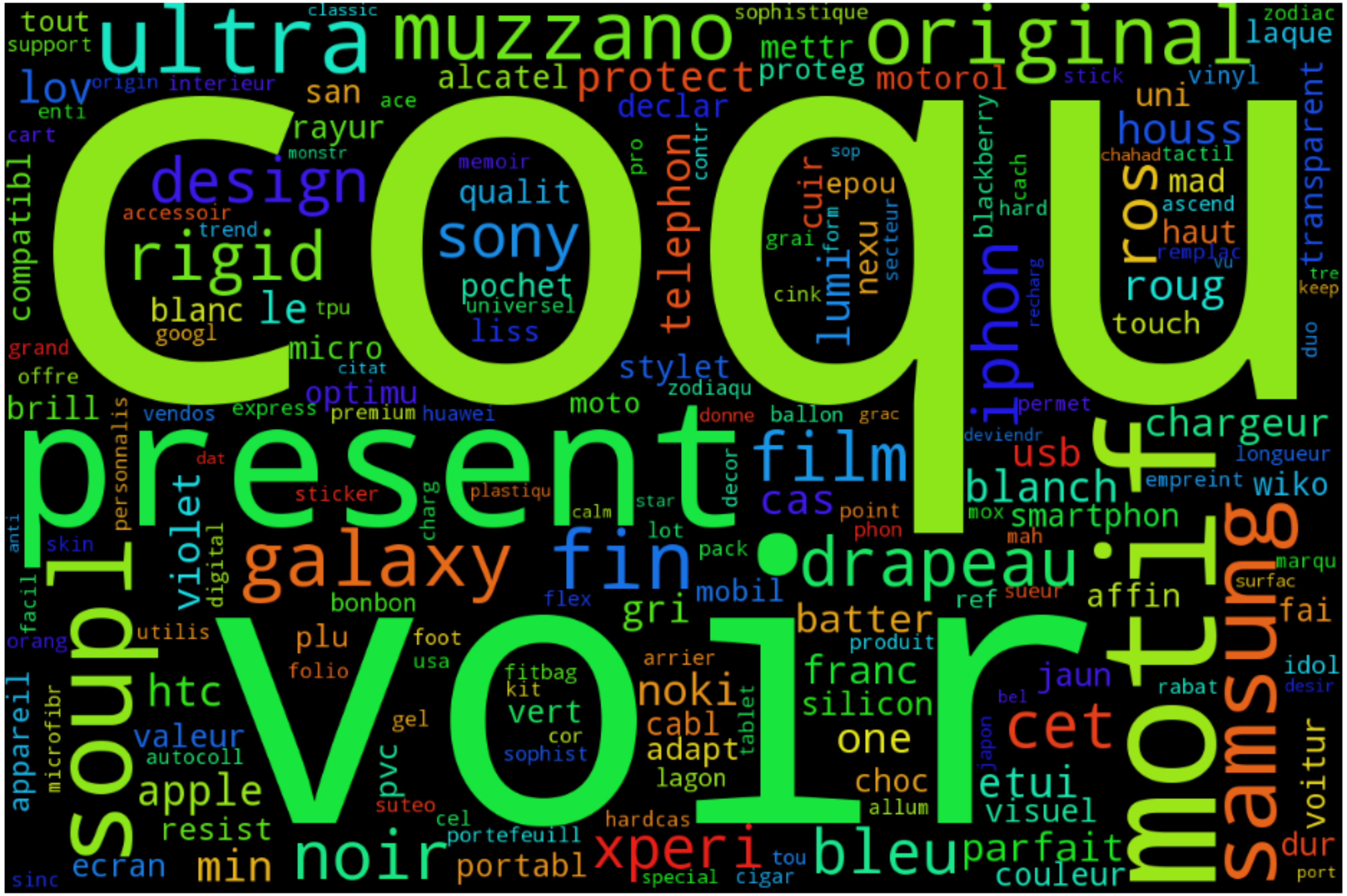
- Upper case can be kept for sentiment analysis.
- Cdiscount: Number can be removed for categorie 1’s level and not categories 2 (xbox 360).
- etc.

Regular expression

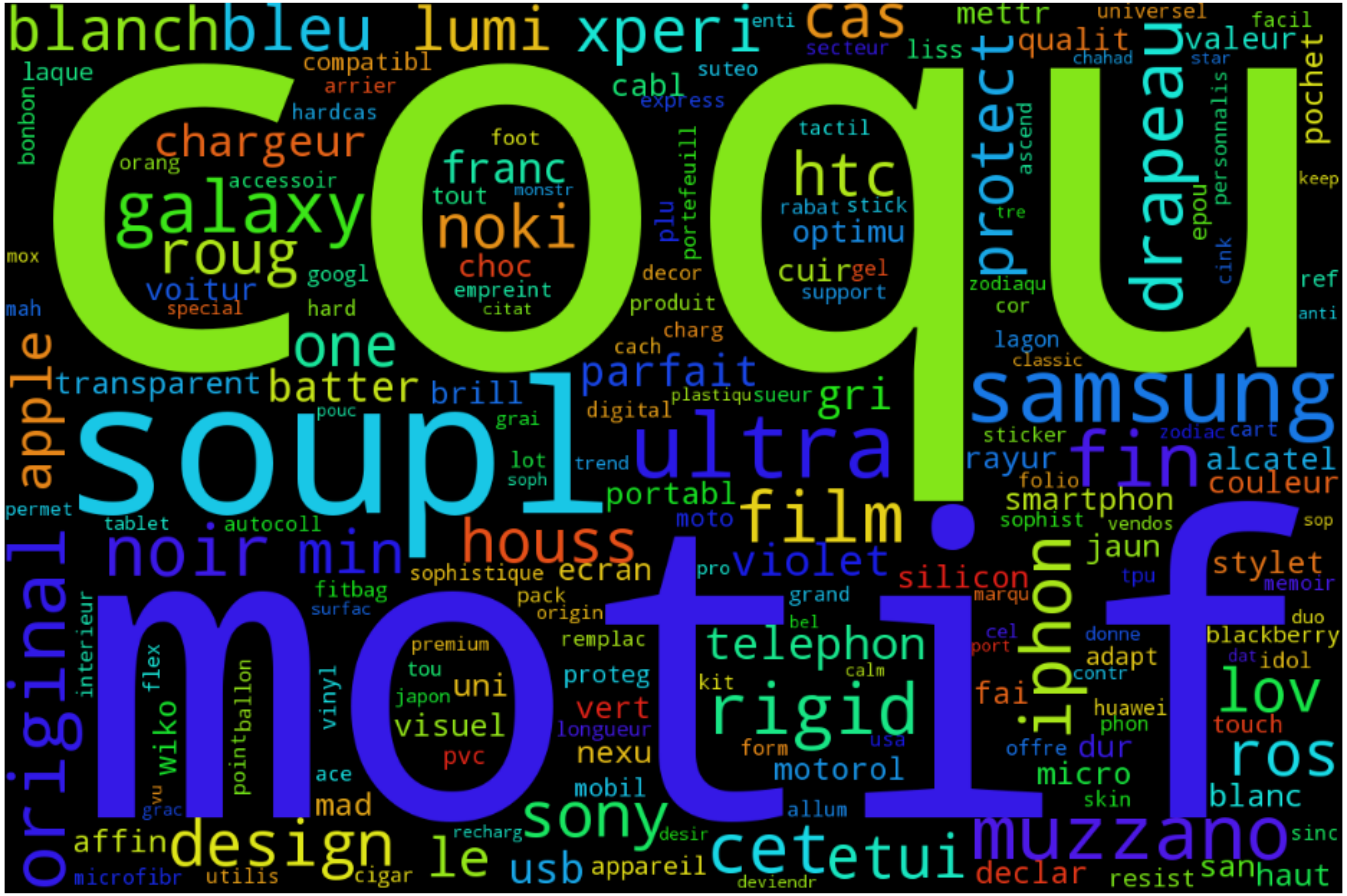
Stopword - Category - “Téléphonie - GPS”



Stopword - Category - “Téléphonie - GPS”



Stopword - Category - “Téléphonie - GPS”



Libraries for text processing

- **NLTK** *(python)*: language processing (steming, stop words)
- **Lucene** *(java)*: text indexation and information retrieval
- **BeautifulSoup**: clean html text.

Vectorisation

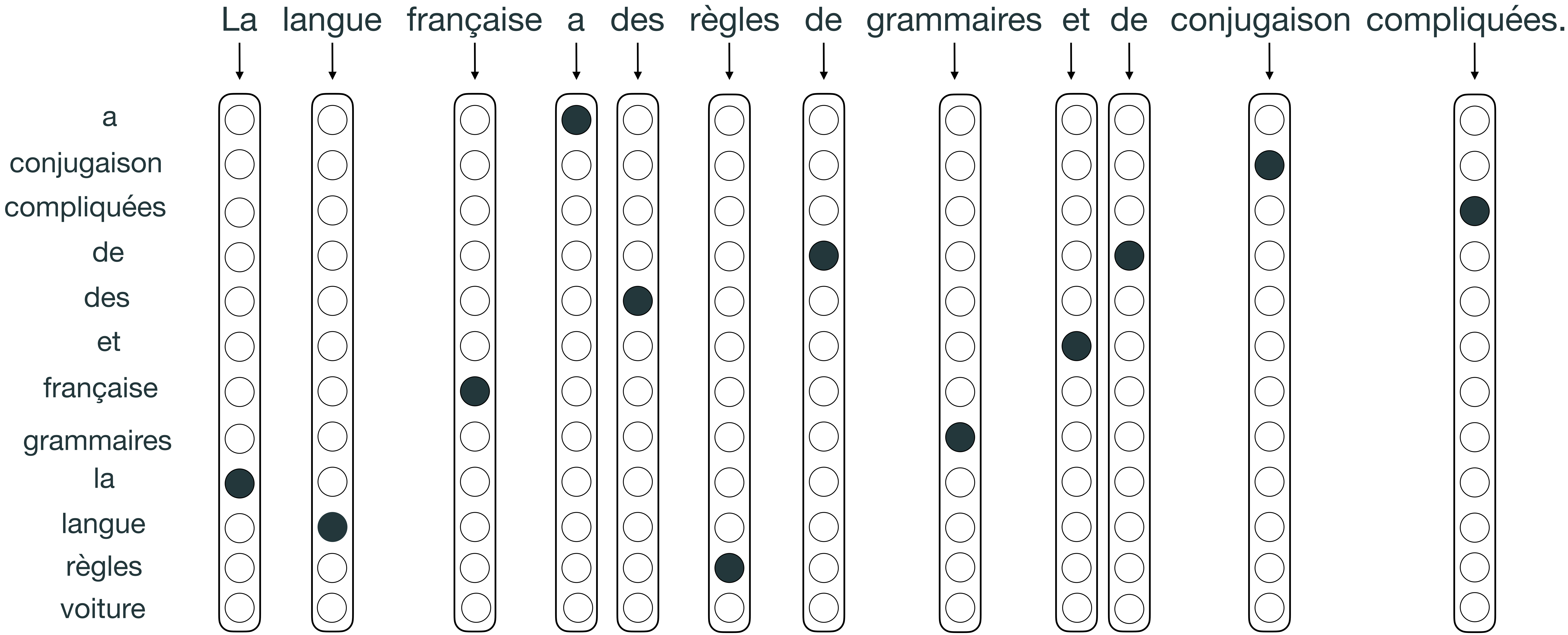
Objectives and difficulties

- Transform text to numerical data to be used in AI algorithms.
- Manage high number of features. Example:
  - 21.543 lines on category “TELEPHONIE - GPS”
  - 24.486 unique words -> 8384 after cleaning.
- Choose significatif words

Two types of solutions:

- Frequency based : Vectorizer
- Learning based : Word Embedding (See corresponding course).

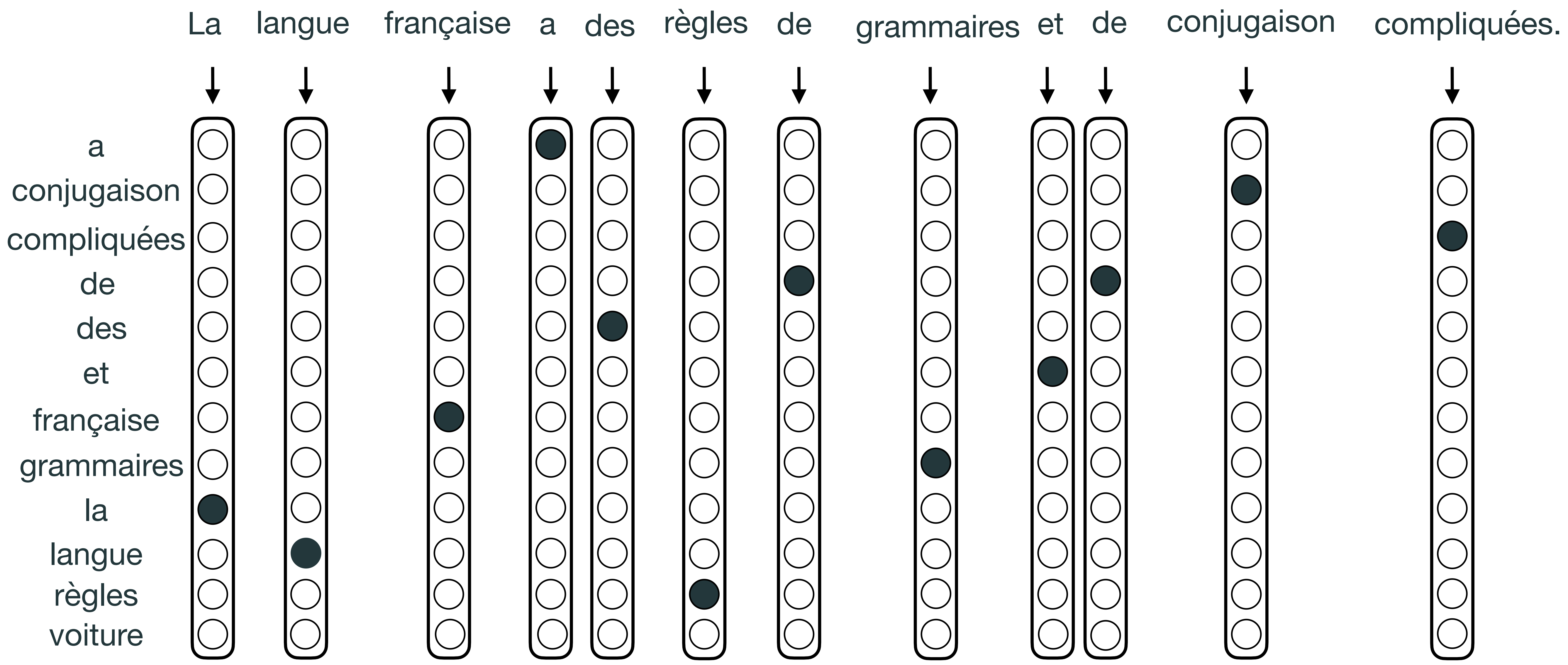
One-Hot-Encoder



Dictionary size: V = 12

Count & Binary Vectoriser

How to convert OHE encoding to ‘sentences’ encoding ?



Count Vectoriser

a	1
conjugaison	1
compliquées	1
de	2
des	1
et	1
française	1
grammaires	1
la	1
langue	1
règles	1
voiture	0

Binary Vectoriser

a	1
conjugaison	1
compliquées	1
de	1
des	1
et	1
française	1
grammaires	1
la	1
langue	1
règles	1
voiture	0

Limitation: all words have the same weights

TF-IDF

Assign a weight to word, a token or an association of words in a document regarding to a corpus of document.

- *t* : a word or and association of words.
- *d* : a document.
- *D* : a corpus of document.

DEFINITION: TF-IDF general formula.

$$\text{tfidf}(t, d) = \text{tf}(t, d) \times \text{idf}(t, D)$$

- *tf(t, d)* : *Term-Frequency*. Number of occurence of token *t* in document *d.*
- idf*(t, D)* : *Inverse-Document-Frequency*. Importance of token *t* in the corpus *D.*

TF Formula

The *tf(t, d)* general formula is defined as the number of occurence *t* in document *d*.

$$tf(t,d) = f_{t,d}$$

This definition is used in **scikit-learn** python library and **MlLib** spark library.

However there exist some variations:

Binary	0,1
Logarithmique normalisation	$1 + log(f_{t,d})$
max normalisation	$K + K \times \frac{f_{t,d}}{max_{t' \in d} f_{t',d}}$
max normalisation (0.5)	$0.5 + 0.5 \times \frac{f_{t,d}}{max_{t' \in d} f_{t',d}}$

IDF Formula

The *idf(t, D)* change from an implementation to another.

$log(\frac{N_D}{DF(t,D)})$	
$log(\frac{N_D + 1}{DF(t,D) + 1})$	MlLib (Spark)
$log(\frac{N_D + 1}{DF(t,D) + 1}) + 1$	Scikit-learn (Python)

- N_D: Number of documents.
- $DF(t, D)$: Number of documents in which terms t appears.

Dimension issue

Binary Vectoriser

a	1
conjugaison	1
compliquées	1
de	1
des	1
et	1
française	1
grammaires	1
la	1
langue	1
règles	1
voiture	

V=11

Binary Vectoriser

a	1
	⋮
conjugaison	1
	⋮
compliquées	1
	⋮
de	1
	⋮
des	1
	⋮
et	1
	⋮
française	1
	⋮
grammaires	1
	⋮
la	1
	⋮
langue	1
	⋮
règles	1
	⋮
voiture	0

V=10.000

- Vectors are very big
- Dimensions gros quickly

Hashing [Weinberger and al, 2009]

Vectorise descriptions while reduce features space

X $\Longrightarrow$ ϕ

Vector of size V, unknown until computing all vocabulary.

Vector of size n_{hash} fixed.

- Determinist function.
- Only one pass on data to build the vector.
- Unbiased cross product: $\mathbb{E}[\langle \phi(x), \phi(x') \rangle] = \langle x, x' \rangle$

Hashing [Weinberger and al, 2009]

Hashed Feature Map

$$\phi_j^{\xi,h}(x) = \sum_{i \ s.t \ h(i)=j} \xi(i) x_i$$

Where

$$h : \mathbb{N} \to \{1,\ldots,nhash\}$$
$$i \mapsto j = h(i)$$

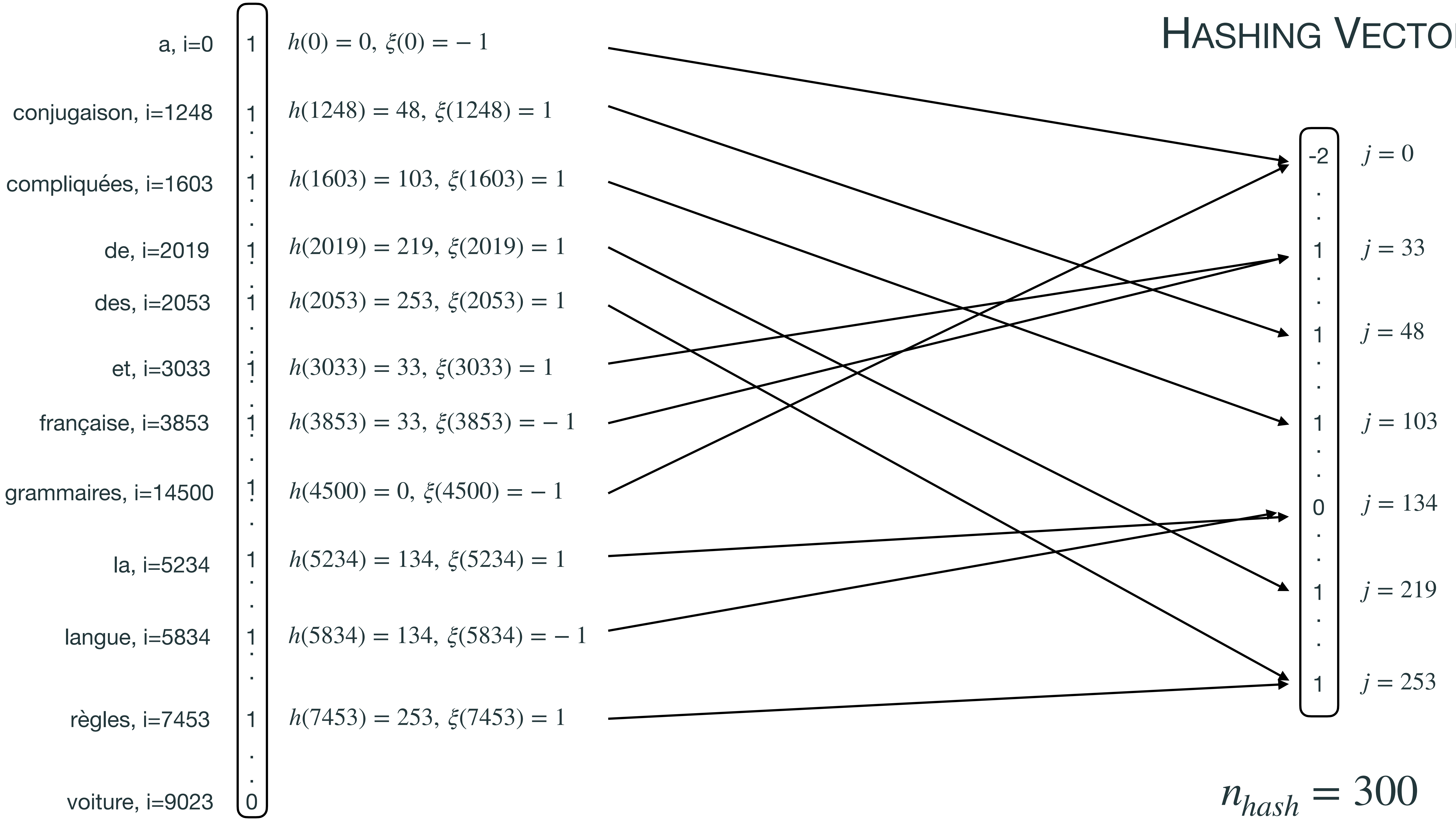
And

$$\xi : \mathbb{N} \to \{1,\ldots, -1\}$$
$$i \mapsto j = \xi(i)$$

Hashing [Weinberger and al, 2009]

Binary Vectoriser

Hashing Vector



$n_{hash} = 300$

$V = 10.000$

Application of vectorisation for learning

- Hashing and then TF-IDF are applied on training dataset.
- Same hashing function are used on test dataset.
- TF value between a word *t* and a document *d* are recomputed for the test dataset.
- IDF terms computed during training are re-used.

N-Grams

Problems

Some words does not have the same sense according to the context where it used.

- *Short de bain* $\neq$ *short* $\neq$ *bain*

Solution

We consider not only the word (*unigram*) but also succession of two (*bigram*) or more words (*n-gram*).

- Solve language ambiguity.
- Explosion of vectors size. Example:
  - For 21.543 lines of categorise “TELEPHONIE - GPS"
  - 8.384 unigrams, 50.012 bigram, 90.854 trigram..

TP

- Clean Cdiscount text’s dataset.
- Vectorize and hash text dataset using scikit-learn library.
- Apply product classification on vectorised data.