PROJET 6 – « CLASSIFIEZ AUTOMATIQUEMENT DES BIENS DE CONSOMMATION »

Soutenance de projet

19 Février 2020

Sommaire

I. Rappel de la problématique et présentation du jeu de données

II. API, Prétraitements et clustering

III. Conclusion sur la faisabilité du moteur de classifications et recommandations

I - PROBLÉMATIQUE

Rappel de la problématique

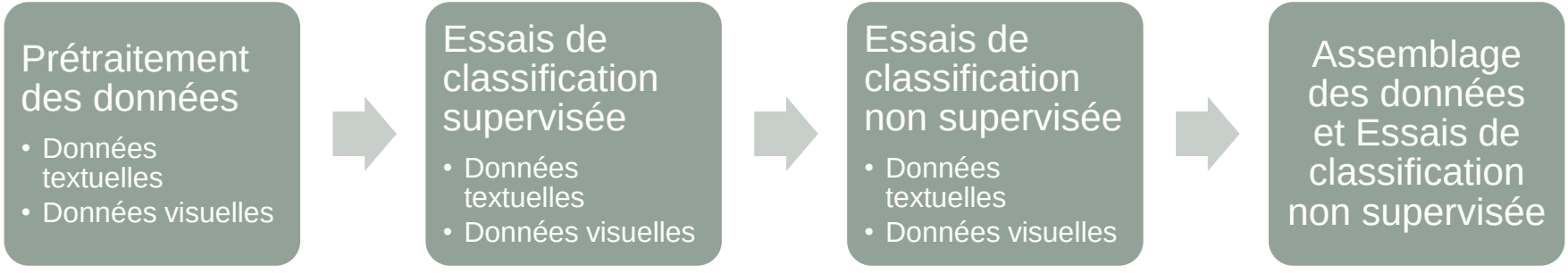
Présentation du jeu de données

Classer automatiquement des articles

- **Contexte** : « Place de marché » : plateforme e-commerce.
- **Moyen** : Automatisation de l'attribution des catégories aux articles
- **Objectif** : améliorer l'expérience utilisateur et fiabiliser la catégorisation

- **But du projet : étudier la faisabilité** de cette catégorisation :
  - Extraction de données depuis une API
  - Analyse et prétraitement du jeu de données : visuelles / textuelles
  - Clustering

Etude de faisabilité : processus



Jeu de données

- 1050 articles
- 15 colonnes par article:
  - Identifiant : id, nom, catégorie de produit, marque, description
  - Prix / prix soldé
  - Note du produit,
  - Image
  - etc.

- Exemples d’articles:
  - Bracelets de montre
  - Vases
  - Linge de lit
  - Batteries d’ordinateur
  - Etc.

II – API, PRETRAITEMENTS ET CLUSTERING

Données textuelles

Données Visuelles

Modélisations effectuées

Données complémentaires : API Amazon

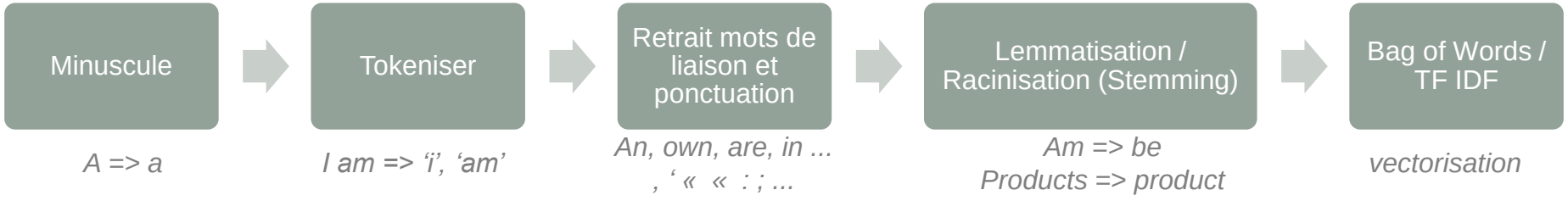
- Exemple : extraction de données pour un type d'article absent de la base de données : **sacs à main**

- *Requete d'extraction :*

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=ItemSearch& #recherche d'article
Keywords=handbag& #titre du produit
VariationPage=1& #première page uniquement
Sort=salesrank #tri en fonction des produits qui se vendent le plus en premier
```

Données textuelles : prétraitement

- Traitements successifs *(librairie NLTK)*



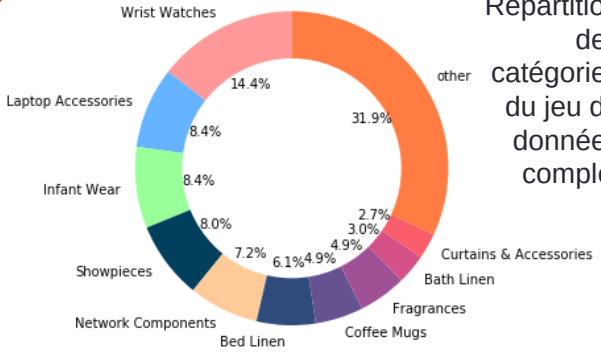
- Exemple

```
Buy Epresent Mfan 1 Fan USB USB Fan for Rs.219 online. Epresent Mfan 1 Fan USB USB Fan at best prices with FREE sh
ipping & cash on delivery. Only Genuine Products. 30 Day Replacement Guarantee.

======== LOWERCASE ========
buy epresent mfan 1 fan usb usb fan for rs.219 online. epresent mfan 1 fan usb usb fan at best prices with free sh
ipping & cash on delivery. only genuine products. 30 day replacement guarantee.

======== TOKENIZER ========
['buy', 'epresent', 'mfan', '1', 'fan', 'usb', 'usb', 'fan', 'for', 'rs.219', 'online', '.', 'epresent', 'mfan', '
1', 'fan', 'usb', 'usb', 'fan', 'at', 'best', 'prices', 'with', 'free', 'shipping', '&', 'cash', 'on', 'delivery',
'.', 'only', 'genuine', 'products', '.', '30', 'day', 'replacement', 'guarantee', '.']

======== STOPWORDS ========
['buy', 'epresent', 'mfan', '1', 'fan', 'usb', 'usb', 'fan', 'rs.219', 'online', 'epresent', 'mfan', '1', 'fan', '
usb', 'usb', 'fan', 'prices', 'free', 'shipping', 'cash', 'delivery', 'genuine', 'products', '30', 'day', 'replace
ment', 'guarantee']

======== LEMMATISATION ========
['buy', 'epresent', 'mfan', '1', 'fan', 'usb', 'usb', 'fan', 'rs.219', 'online', 'epresent', 'mfan', '1', 'fan', '
usb', 'usb', 'fan', 'price', 'free', 'shipping', 'cash', 'delivery', 'genuine', 'product', '30', 'day', 'replaceme
nt', 'guarantee']
```

==== TF-IDF ====

	mot	tfidf
7	fan	0.636882
18	usb	0.454660
6	epresent	0.388140
11	mfan	0.388140
16	rs.219	0.194070
0	1	0.143306
12	online	0.054498
10	guarantee	0.049844
1	30	0.049227
15	replacement	0.048849

Données textuelles : catégorisation



Répartition des catégories du jeu de données complet

Classifieur supervisé (Essai avec SVC) :

- Création d'une nouvelle feature « catégorie niveau 2 » à partir des données (62 catégories)
- Accuracy sur jeu test : 79 %

Classifieurs non supervisés

- Réduction de dimension : Latent Dirichlet Allocation avec 62 catégories

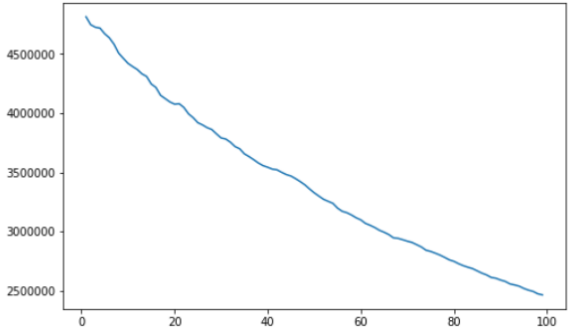
Affichage des 10 mots les plus importants de chaque topic –

```
Topic #0: perucci decker resistant wine easy stylish watche comfortable beautifull island
Topic #1: watch offer cushion cover black home taste strap 5 geometric
Topic #2: double sheet bedsheet cm warranty cotton apple adapter macbook laptop
Topic #3: ceramic prithish mug delivery shipping genuine cash product free buy
Topic #4: wooden 299 17 mediterranean sea lucky ship ii part handcrafted
Topic #5: skin laptop print shape set combo pad mouse warranty multicolor
```

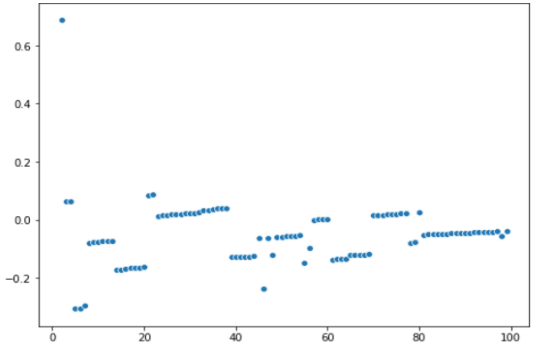
- Kmeans (après ACP) :

Non concluante

Evolution de la somme des inerties en fonction du nombre de clusters

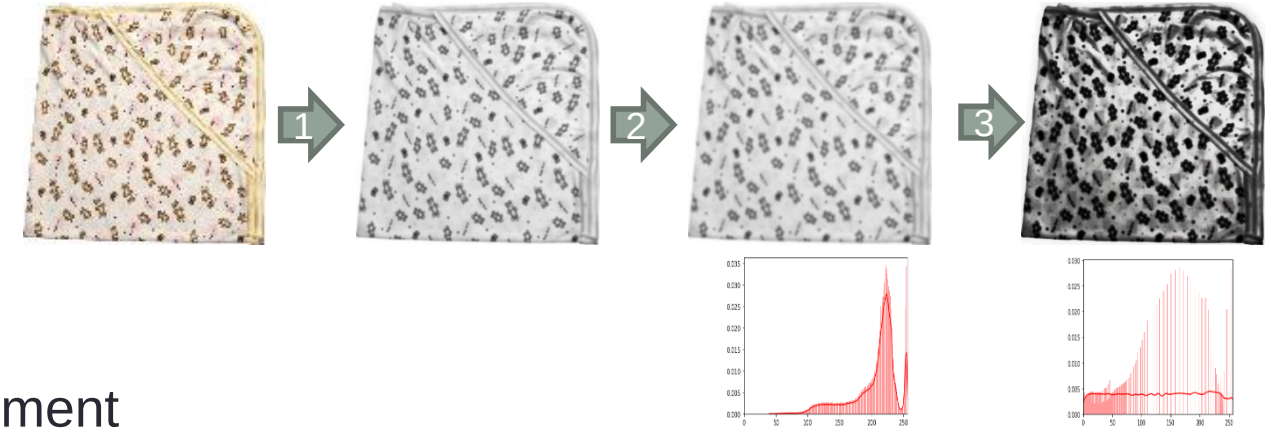


Coefficient de silhouette moyen en fonction du nombre de clusters

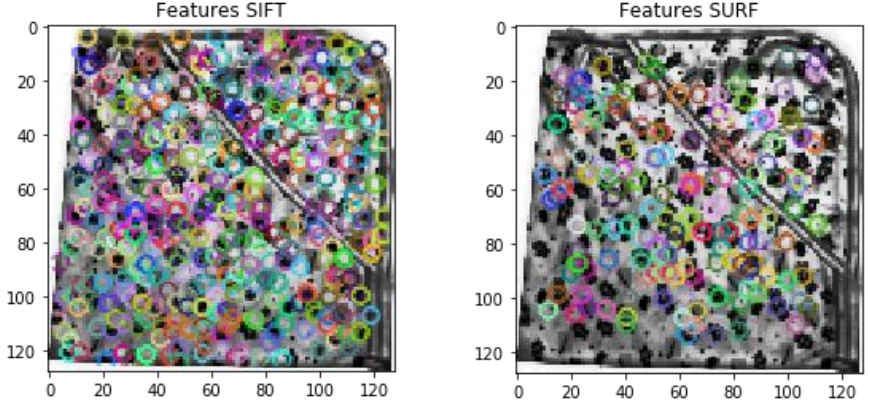


Données visuelles : extraction de features

- Pré-traitement
  - (1) Noir et Blanc
  - (2) Réduction bruit (flou gaussien)
  - (3) Egaliseur
  - (4) Redimensionnement



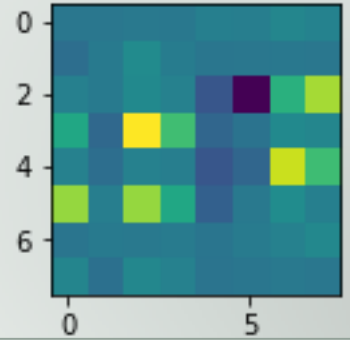
- Extraction de features (Exemple avec SIFT/SURF)



Données visuelles : extraction de features

- Création de features à partir des informations

N descripteurs SURF par image (64px / 1 dim)

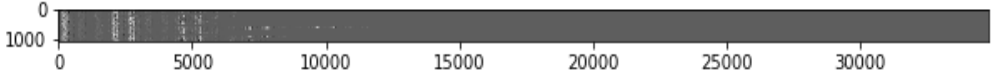


Concaténation des features d'une image sur une ligne de tableau

Remplissage des features « absentes » par des 0

Concaténation des features de toutes les images dans un même tableau

- Obtention d'un array « creux » :

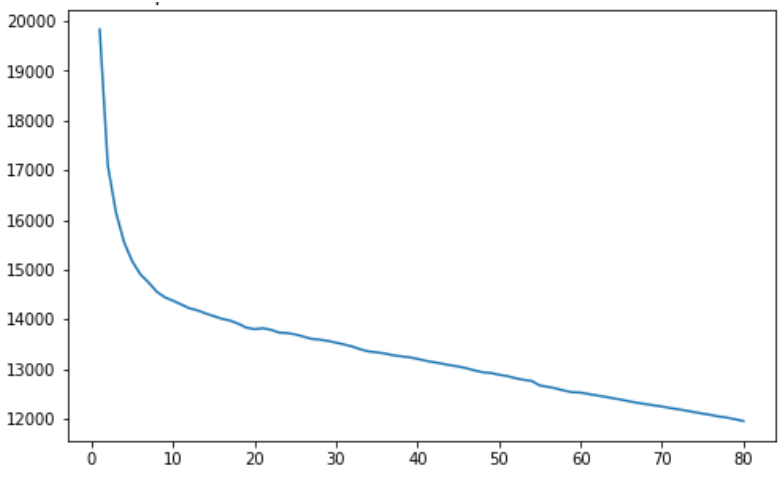


- Création de nouvelles colonnes à partir des descripteurs:
  - Min, max, médiane, variance, moments d'ordre 3 et 4

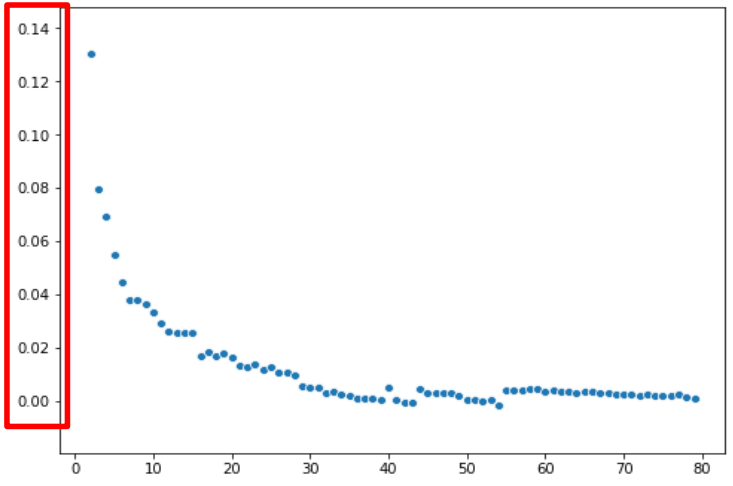
Données visuelles : classification

- Classification après extraction des features
  - Kmeans après ACP : *Non concluant*

Evolution de la somme des inerties en fonction du nombre de clusters



Coefficient de silhouette moyen en fonction du nombre de clusters



Données visuelles : Réseaux de neurones

- Construction d'un réseau de neurone convolutif simple

```
model = Sequential()
model.add(Conv2D(32, kernel_size=(3,3), padding='same', activation='relu', input_shape=(128,128,3,)))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Conv2D(32, kernel_size=(3,3), padding='same', activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(ohe.categories_[0].shape[0], activation='softmax'))
model.compile(loss='mean_squared_error', optimizer='sgd')
```

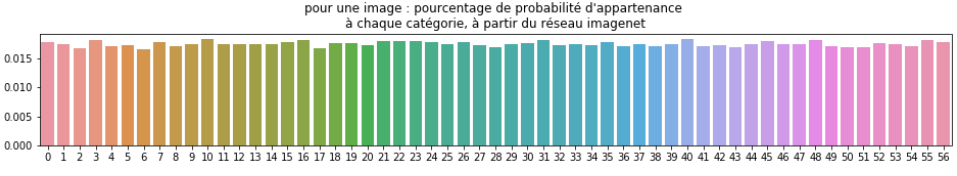
- Entraînement supervisé

```
model.fit(train_array_cnn, train_array_cats, epochs=3, batch_size=40, verbose=2)
```

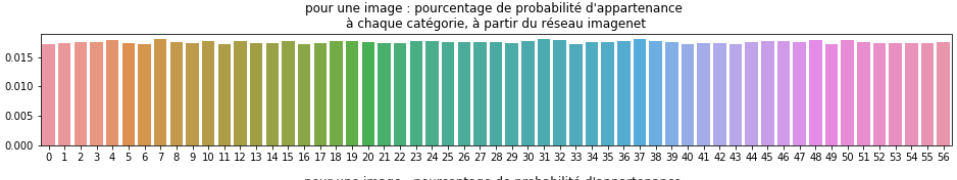
  - Accuracy score : 8 %
  - Classification de tout le jeu de test dans la catégorie la plus représentée : non concluant

Données visuelles : Réseaux de neurones

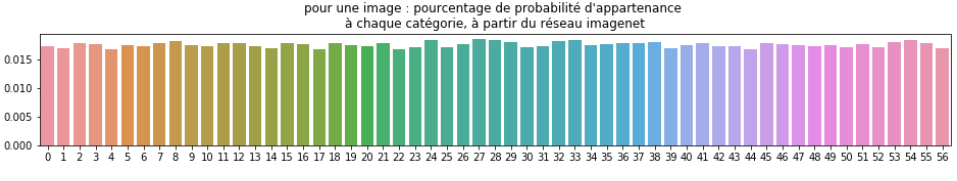
- Transfer Learning (ImageNet VGG16):
  - Substitition dernières couches par couche Dense
  - Préparation des données:
    - Redimensionnement 224 x 224
    - Onehotencoding catégories
  - Entraînement du réseau : probabilité d’appartenance à chaque catégorie : exemple pour 3 images



pour une image : pourcentage de probabilité d'appartenance à chaque catégorie, à partir du réseau imagenet



pour une image : pourcentage de probabilité d'appartenance à chaque catégorie, à partir du réseau imagenet



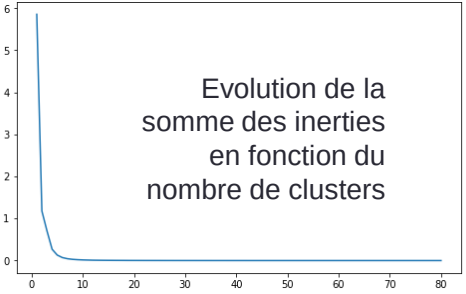
pour une image : pourcentage de probabilité d'appartenance à chaque catégorie, à partir du réseau imagenet

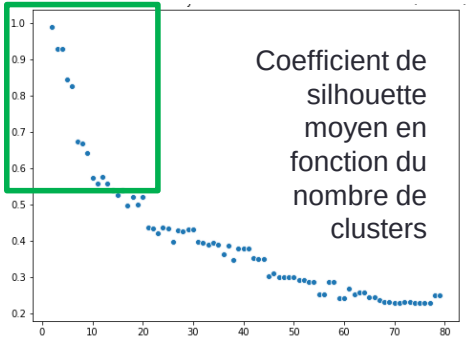
Assemblage données visuelles et textuelles

- Export / import des données
- Dimensions des données à assembler
  - Tableau données textuelles: *(1050 lignes, 651 colonnes)*
  - Nouvelles features Descripteurs : *(1050 lignes, 6 colonnes)*
  - Réseau de neurone Imagenet : *(1050 lignes, 57 colonnes)*
- Fort déséquilibre entre dimensions des données textuelles et visuelles:
  ⇒ *Réduction de dimension des données textuelles par ACP : 341 colonnes avec 80 % de variance*

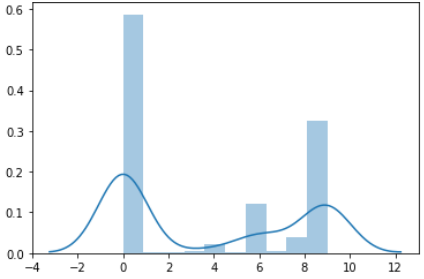
- Assemblage: obtention d'un array de 1050 lignes x 404 colonnes

- Classification non supervisée





Distribution du nombre d'éléments par catégorie (exemple avec 10 clusters)

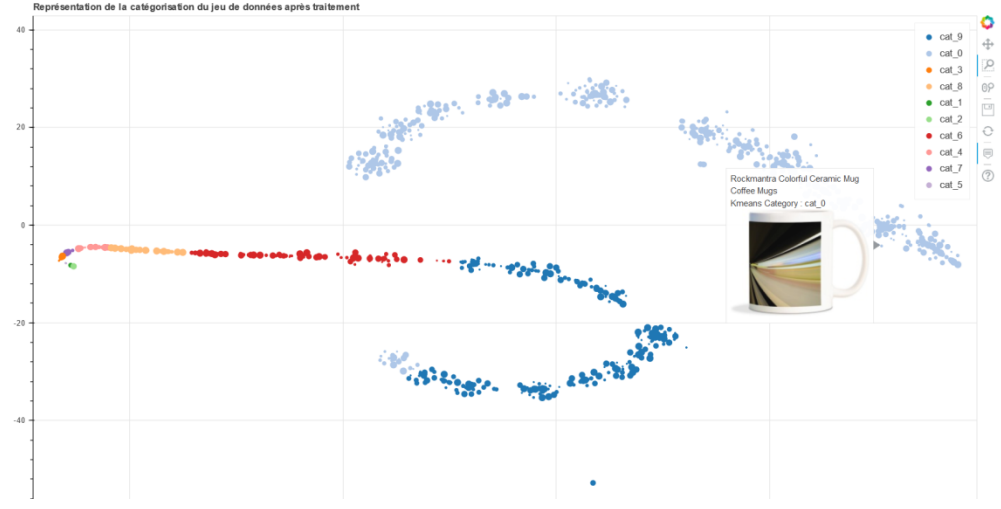


III – CONCLUSIONS ET RECOMMANDATIONS

Résultats obtenus

- Résultats du clustering (Kmeans + T-SNE pour visualisation)

Voir graphe dynamique



- Clustering non supervisé : résultat non concluant
- Alternative envisageable : apprentissage supervisé

Aller plus loin

- **Taille du jeu de données** : appels à l'API
- **Stopwords NLP** : vocabulaire du e commerce
- **Transfer learning sur données textuelles** (e.g. BERT)
- **Obtention d'un jeu de données labelisé** :
  - Collecte de donnée sur le site ou obtention externe
  - Déploiement dans un second temps

MERCI DE VOTRE ATTENTION