# Projet 6 :
# Classifiez automatiquement des biens de consommation

Raphaël GIRAUDOT

# Sommaire

1. Contexte et objectifs de la mission
2. Données à disposition
3. Extraction de features
    a. Texte
    b. Images
4. Clustering
    a. Texte
    b. Images
5. Alternative extraction features images : CNN
6. Combinaison features texte & image
7. Conclusion & Perspectives

# 1. Contexte et objectifs de la mission


Marketplace, mise en ligne des annonces par les vendeurs :

- Upload photos produit
- Écriture description produit
- Sélection  catégorie du produit

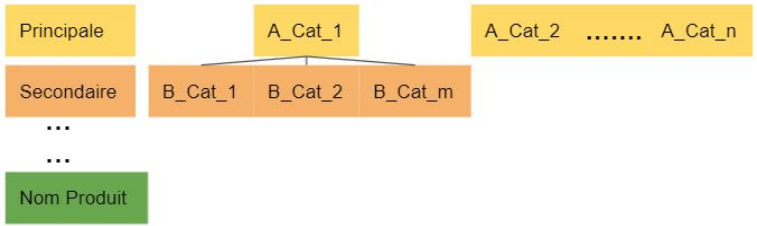
→ Améliorer l'expérience utilisateur

Étudier la faisabilité d'un moteur de classification automatique de produits.

# 2. Données à disposition

Dossier d'images des produits :
- 1050 Produits / Images

Informations textuelles sur les produits (fichier .CSV) :
- 1050 Produits / Descriptions
- 15 Variables, dont:
  ID, Nom produit, Nom fichier image, Description, Catégorie




| | product_category_tree | n | f |
|---|---|---|---|
| 0 | HomeFurnishing | 150 | 14.285714 |
| 1 | BabyCare | 150 | 14.285714 |
| 2 | Kitchen&Dining | 150 | 14.285714 |
| 3 | Computers | 150 | 14.285714 |
| 4 | Watches | 150 | 14.285714 |
| 5 | BeautyandPersonalCare | 150 | 14.285714 |
| 6 | HomeDecor&FestiveNeeds | 150 | 14.285714 |

# 3. Extraction de Features

## a. Texte (Nettoyage)

Description

↓

Uniformisation
(Retrait nombres & Lower)
+
Tokenisation

↓

Retrait Stop-Words

↓

Lemmatisation / Stemming

↓

Bag of Words

```
desc['description'].iloc[100]
```

```
'Buy Goldencollections GC4353 Makeup and Jewellery Vanity Pouch for Rs.783 online. Goldencollections GC4353 Makeup and Jeweller
y Vanity Pouch at best prices with FREE shipping & cash on delivery. Only Genuine Products. 30 Day Replacement Guarantee.'
```

---

```
re.sub(r'\d+', '', desc['description'].iloc[100])
```

```
'Buy Goldencollections GC Makeup and Jewellery Vanity Pouch for Rs. online. Goldencollections GC Makeup and Jewellery Vanity Po
uch at best prices with FREE shipping & cash on delivery. Only Genuine Products.  Day Replacement Guarantee.'
```

['buy', 'goldencollections', 'gc', 'makeup', 'and', 'jewellery', 'vanity', 'pouch', 'for', 'rs', 'online', 'goldencollections', 'gc', 'makeup', 'and', 'jewellery', 'vanity', 'pouch', 'at', 'best', 'prices', 'with', 'free', 'shipping', 'cash', 'on', 'delivery', 'only', 'genuine', 'products', 'day', 'replacement', 'guarantee']

---

['and', 'for', 'and', 'at', 'with', 'on']

---

**base** → **Lemmatisé** / **Racinisé (Stemmed)**
buy → buy / buy, goldencollections → goldencollections / goldencollect, jewellery → jewellery / jewelleri,
vanity → vanity / vaniti, online → online / onlin, shipping → shipping / ship, replacement → replacement / replac, etc...

---

['buy', 'goldencollect', 'gc', 'makeup', 'jewelleri', 'vaniti', 'pouch', 'rs', 'onlin', 'goldencollect', 'gc', 'makeup', 'jewelleri', 'vaniti','pouch', 'best', 'price', 'free', 'ship', 'cash', 'deliveri', 'genuin', 'product', 'day', 'replac', 'guarante']

# 3. Extraction de Features

## a. Texte (Tf-Idf)

Bag of Words

['buy', 'goldencollect', 'gc', 'makeup', 'jewelleri', 'vaniti', 'pouch', 'rs', 'onlin', 'goldencollect', 'gc', 'makeup', 'jewelleri', 'vaniti','pouch', 'best', 'price', 'free', 'ship', 'cash', 'deliveri', 'genuin', 'product', 'day', 'replac', 'guarante']

↓

Tf-Idf

Transformer texte -> valeurs numériques

Tf-Idf = Tf x Idf

Tf : Term Frequency, fréquence d'apparition d'un mot.

ex: "makeup" apparaît 2 fois dans le 1er doc et 50 fois dans l'ensemble des docs, Tf = 2/50

Idf : Inverse Document Frequency, le log de l'inverse de la proportion de document du corpus qui contiennent le mot

ex: "makeup" apparaît dans 20 documents sur 1050, Idf = log(1050/20)

Paramètres :

- max_df : fixe un seuil pour le Document Frequency à ne pas dépasser pour compter le mot (0.7)
- min_df : fixe un seuil minimum pour le Document Frequency pour compter le mot (0.02)

↓

nombre features :
495 (lemmatisé)
500 (stemmed)

exemple du bag of words stemmed :

| | 37 | 55 | 61 | 111 | 114 | 169 | 180 | 198 | 310 | 350 | 354 | 372 | 399 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 100 | 0.407556 | 0.252023 | 0.255334 | 0.269933 | 0.254778 | 0.246919 | 0.255334 | 0.283669 | 0.313734 | 0.261592 | 0.228951 | 0.276492 | 0.255056 |

# 3. Extraction de Features

## b. Images (Nettoyage)

1. Chargement en niveau de gris
2. Débruitage : Flou gaussien pour lisser

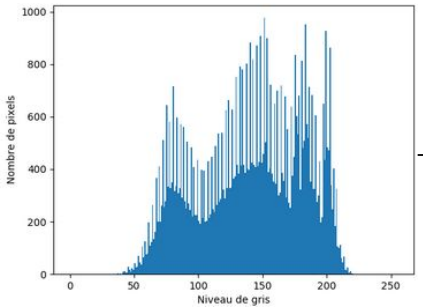
Image bruitée :

Flou Gaussien

Image dé-bruitée :

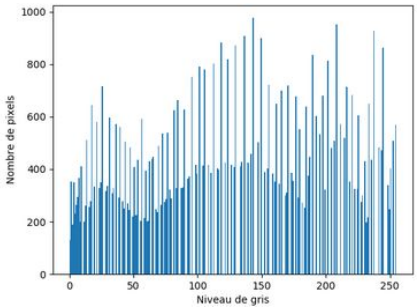
3. Egalisation : Corriger / Uniformiser le contraste

Image sous-exposée :




CLAHE

Image égalisée :



# 3. Extraction de Features

## b. Images

Descripteurs:
ORB / SIFT

↓

Clustering des Descripteurs
(KMeans)

↓

Bag of Visual Words

↓

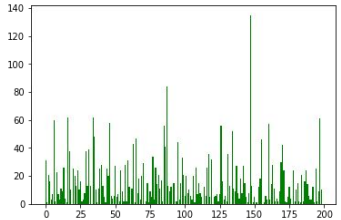
209 features

Extraction de descripteurs :
Chercher des point d'intérêt dans l'image (différence de gradients), orientation et localisation

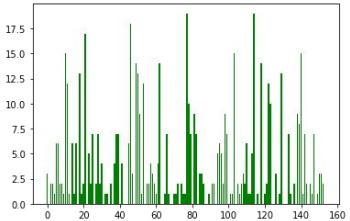
Regroupement des descripteurs proches dans des classes

Histogramme des classes des descripteurs = Empreinte digitale des images
Compter le nombre de descripteurs dans chaques classes

ORB :




SIFT :




| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ... | 200 | 201 | 202 | 203 | 204 | 205 | 206 | 207 | 208 | 209 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 69.0 | 44.0 | 44.0 | 59.0 | 28.0 | 62.0 | 72.0 | 32.0 | 53.0 | 21.0 | ... | 25.0 | 60.0 | 37.0 | 32.0 | 44.0 | 61.0 | 57.0 | 29.0 | 50.0 | 25.0 |
| 1 | 45.0 | 41.0 | 24.0 | 59.0 | 42.0 | 51.0 | 27.0 | 21.0 | 44.0 | 10.0 | ... | 104.0 | 151.0 | 10.0 | 26.0 | 28.0 | 47.0 | 42.0 | 34.0 | 89.0 | 39.0 |
| 2 | 40.0 | 38.0 | 27.0 | 26.0 | 53.0 | 36.0 | 55.0 | 149.0 | 26.0 | 81.0 | ... | 36.0 | 204.0 | 106.0 | 66.0 | 21.0 | 22.0 | 33.0 | 35.0 | 37.0 | 25.0 |

# 4. Clustering

## a. Features Texte

1ère Méthode: BoW → ACP → k-means

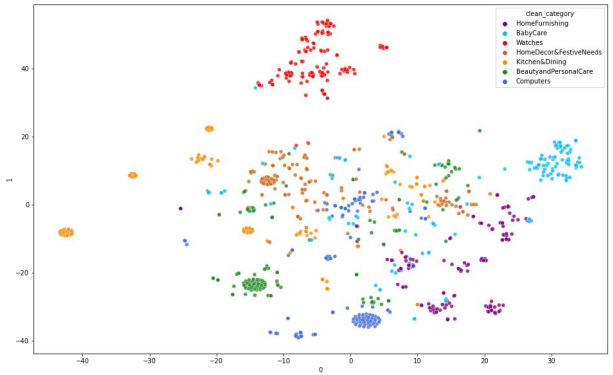
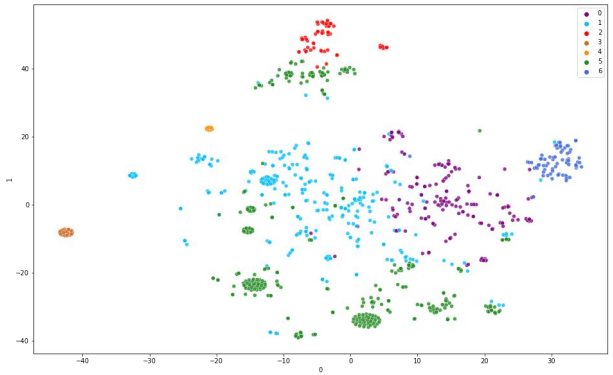
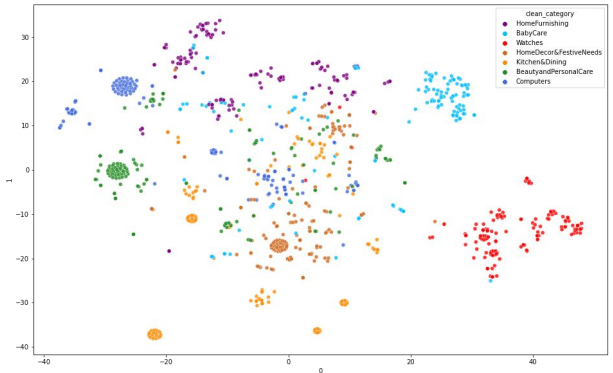
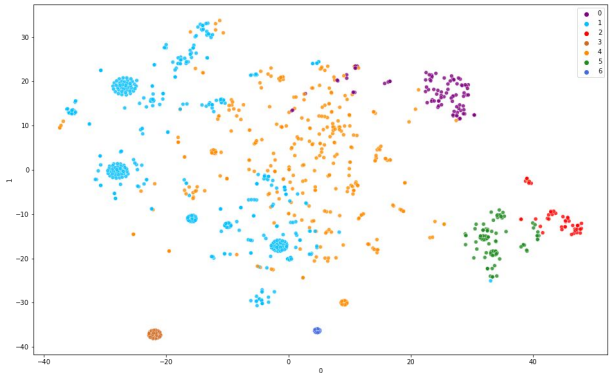
2ème Méthode: BoW → LDA

Comparaison & Evaluation : t-SNE (Hue: Catégories “cibles”) ↔ t-SNE (Hue: Clusters)

# 4. Clustering

## a. Features Texte

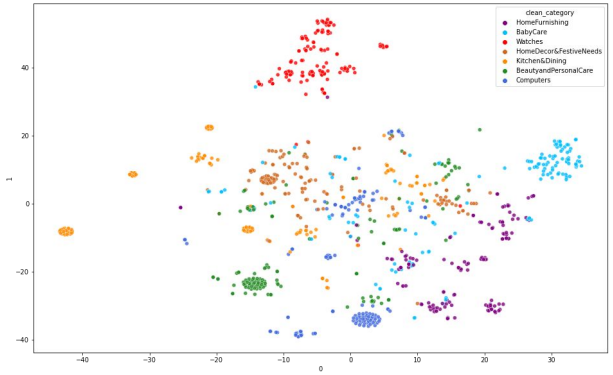
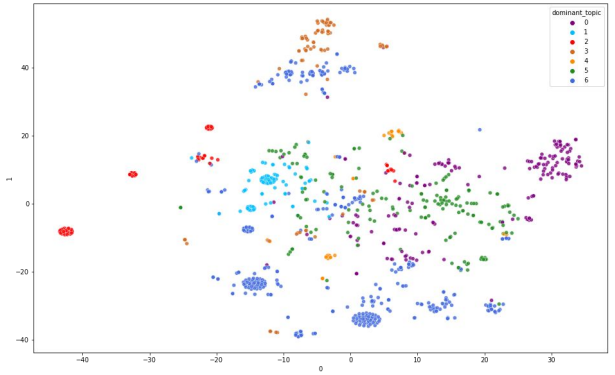
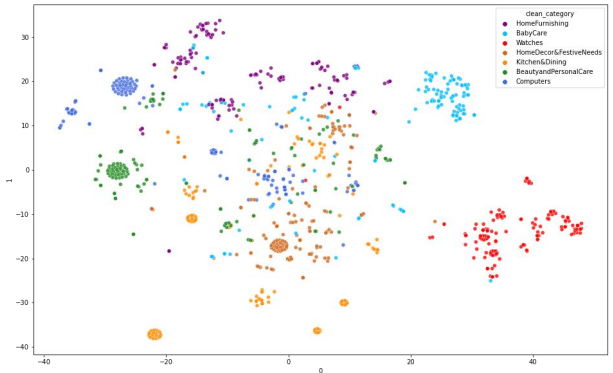
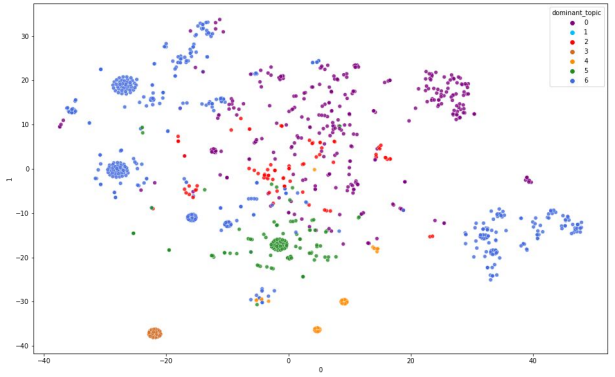
1ère Méthode: Comparaison & Évaluation des projections t-SNE

| | Hue: Catégories “cibles” | Hue: Clusters k-means | |
|---|---|---|---|
| Lemmatized |  |  | Adjusted Rand Score : 0.13251767473185916 |
| Stemmed |  |  | Adjusted Rand Score : 0.129795390970757 |

# 4. Clustering

## a. Features Texte

2ème Méthode: Comparaison & Évaluation des projections t-SNE

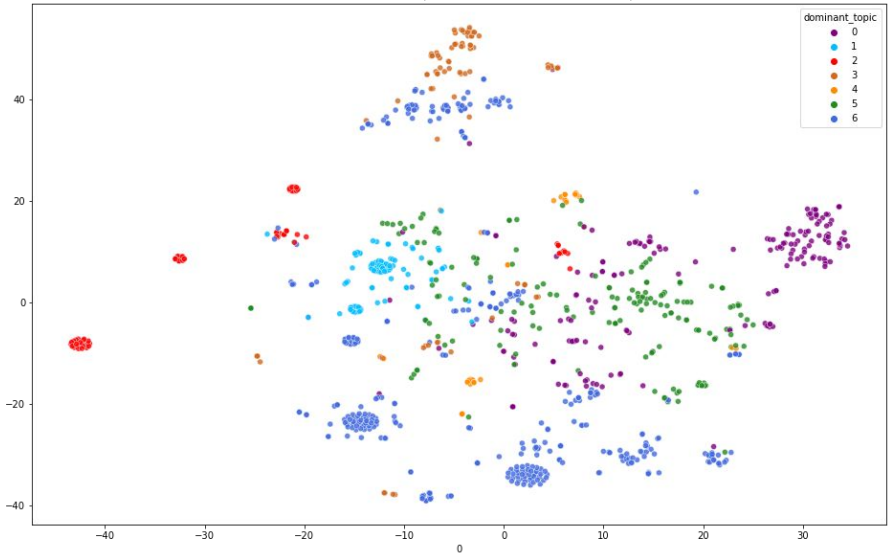
| | Hue: Catégories “cibles” | Hue: Clusters LDA | |
|---|---|---|---|
| Lemmatized |  |  | Adjusted Rand Score : 0.17563542416610572 |
| Stemmed |  |  | Adjusted Rand Score : 0.12958369304144618 |

# 4. Clustering

## a. Features Texte

2ème Méthode: Comparaison & Évaluation des projections t-SNE

Hue: Clusters LDA (Lemmatized)




**Topic 0:** → BabyCare ?
age bleach genuine dimension kit fit head box brown ceramic

**Topic 1:** → HomeDecor&FestiveNeeds ?
aroma dial help durability comfort gentle adorable capacity happy grey

**Topic 2:** → Kitchen&Dining ?
keep distinctive essential attractive item discount id general kadhai gentle

**Topic 3:** → Part of Watches?
graphic dual bedsheet limited express button huge clasp grey collection

**Topic 4:** → Melange
express bedsheet body geometric cotton floral content dual essential coffee

**Topic 5:** → Melange
handle ideal bleach apply gentle find kid get cupcake cash

**Topic 6:** → Melange
brand come beauty contemporary handcrafted ce bath comfortable get cupcake

# 4. Clustering

## a. Features Images

BoW (SIFT-ORB) → ACP → k-means

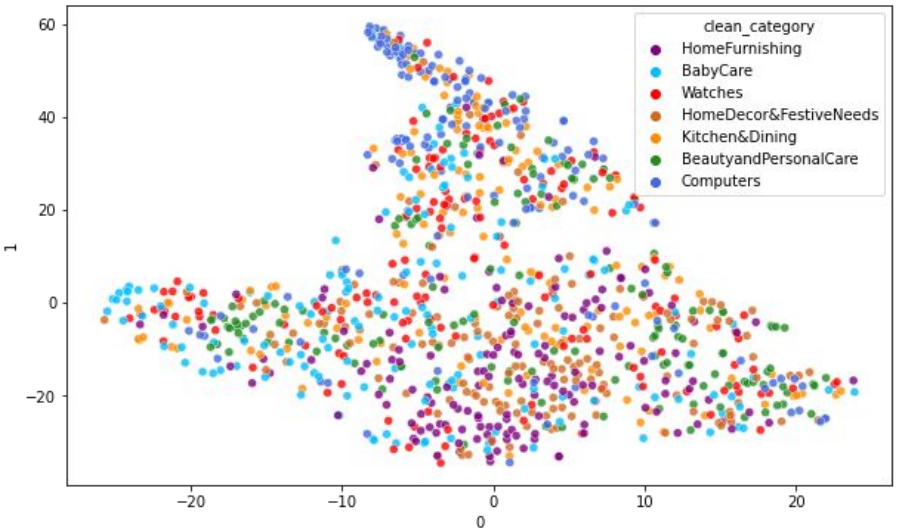
Comparaison & Evaluation :

t-SNE (Hue: Catégories “cibles”) ⟷ t-SNE (Hue: Clusters)

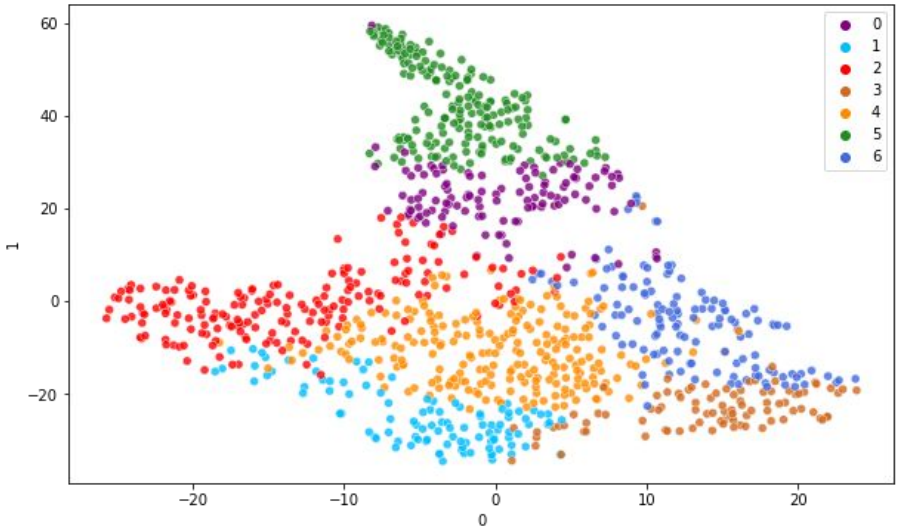
# 4. Clustering

## b. Features Images : ORB

Hue: Catégories “cibles”




Hue: Clusters k-means

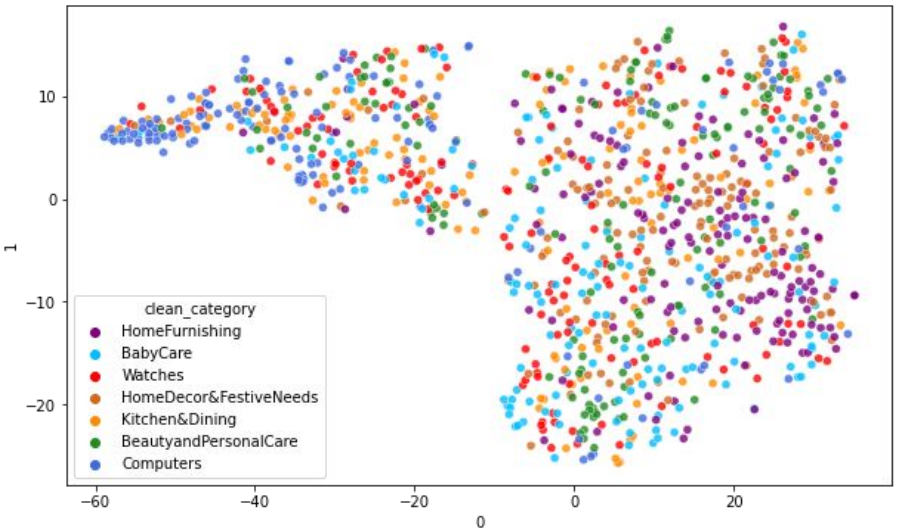



Adjusted Rand Score : 0.07736418170614169

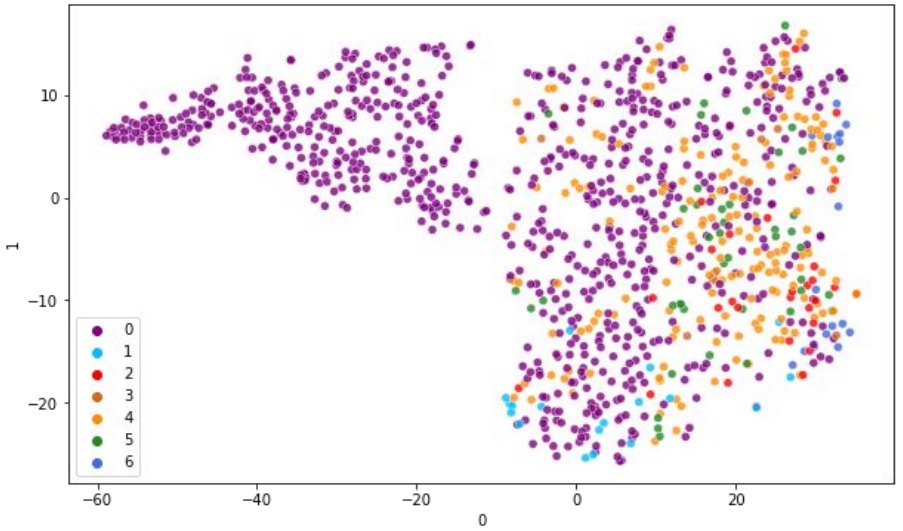
# 4. Clustering

## b. Features Images : SIFT

Hue: Catégories “cibles”




Hue: Clusters k-means

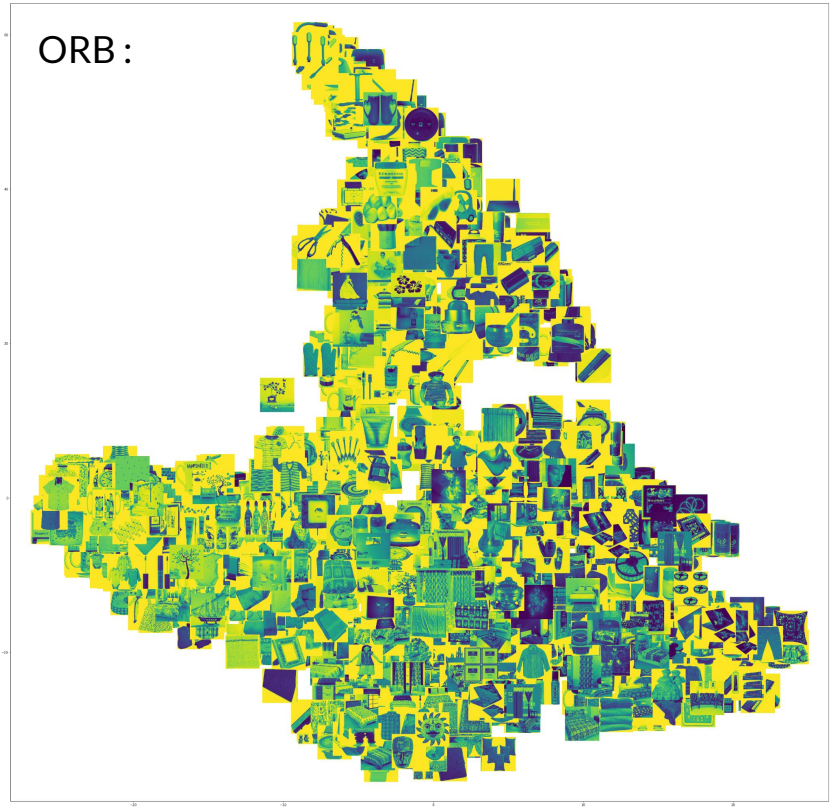



Adjusted Rand Score : 0.037254157784403 47

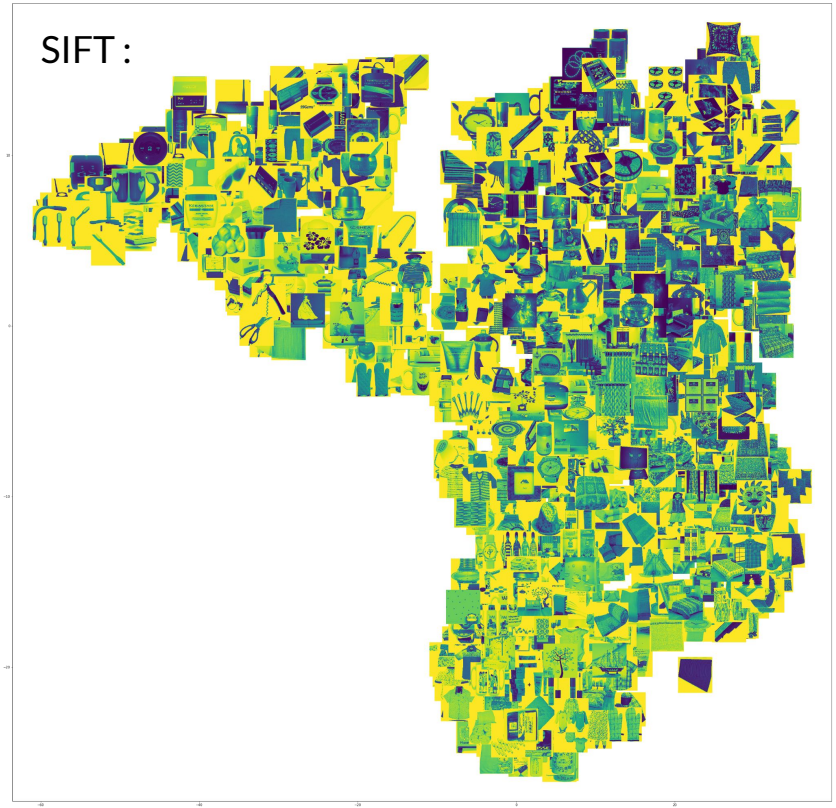
# 4. Clustering

## c. Projections Images

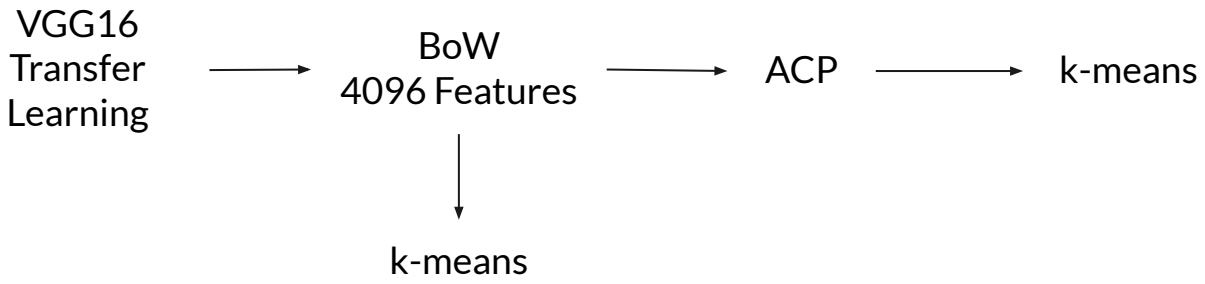
ORB :



SIFT :



# 5. Alternative Images Feature Extraction : CNN




Comparaison & Evaluation :

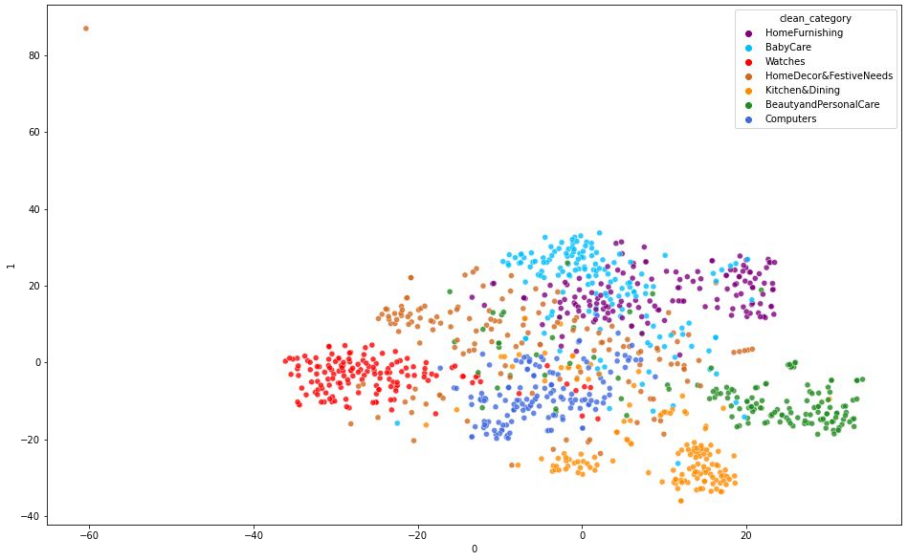
t-SNE
(Hue: Catégories “cibles”)

t-SNE
(Hue: Clusters)

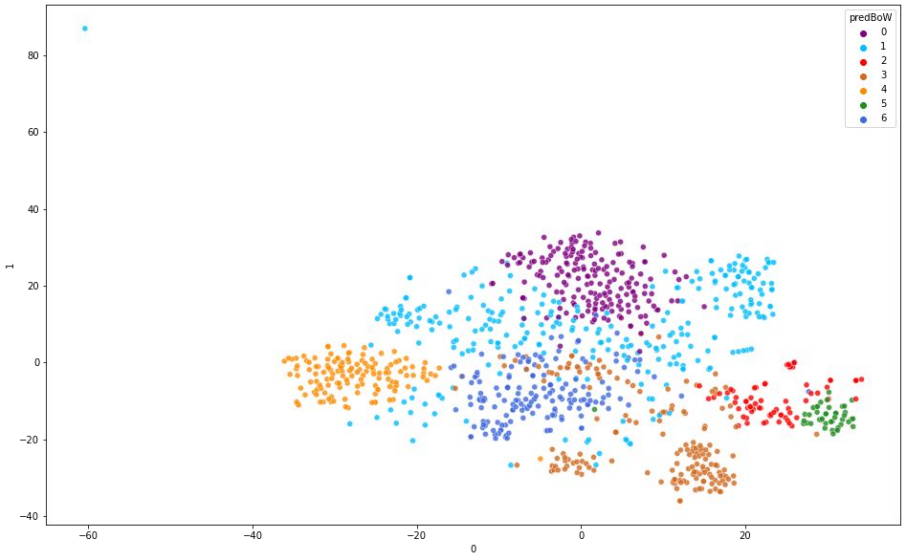
# 5. Alternative Images Feature Extraction : CNN

## a. t-SNE & K-means avant réduction de dimension

Hue: Catégories “cibles”




Hue: Clusters k-means



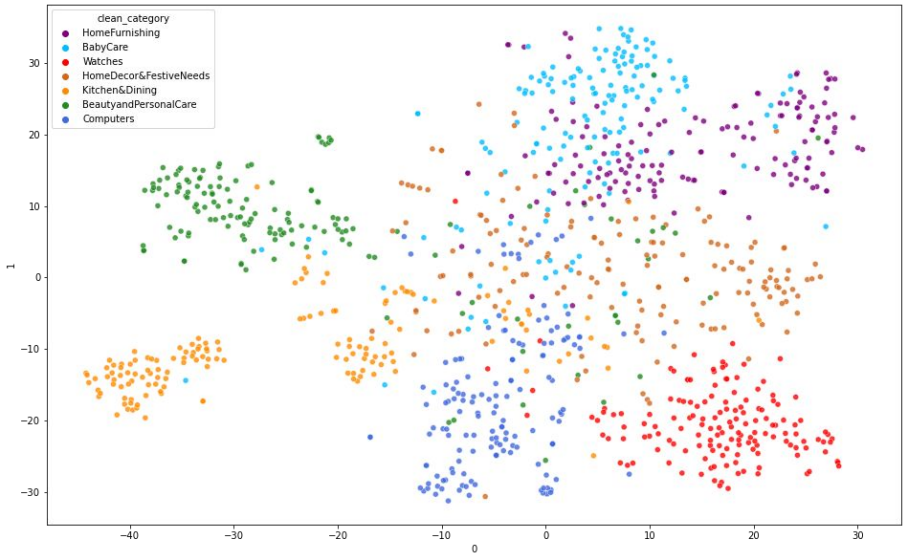

Adjusted Rand Score : 0.47777881544908984

# 5. Alternative Images Feature Extraction : CNN

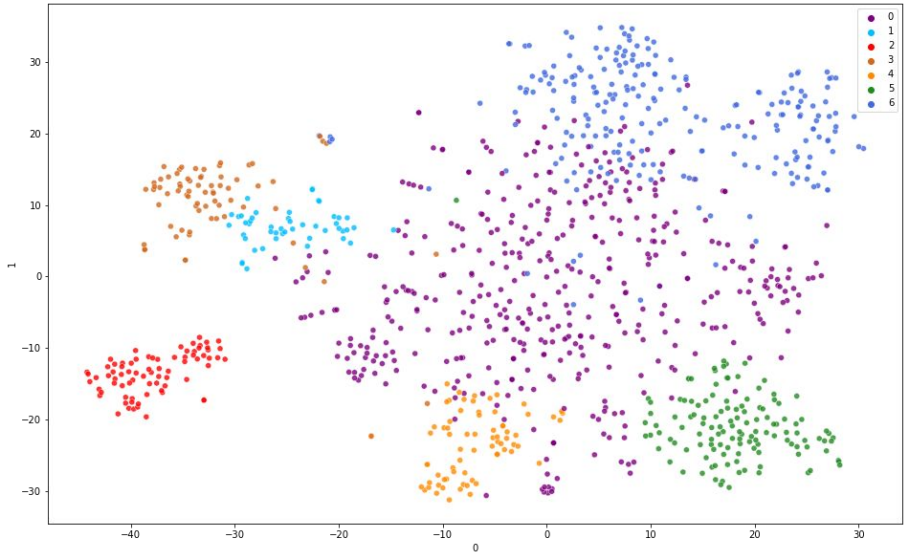
## b. t-SNE & K-means après réduction de dimension

4096 Features → 235 Features (5,7%)

Hue: Catégories “cibles”



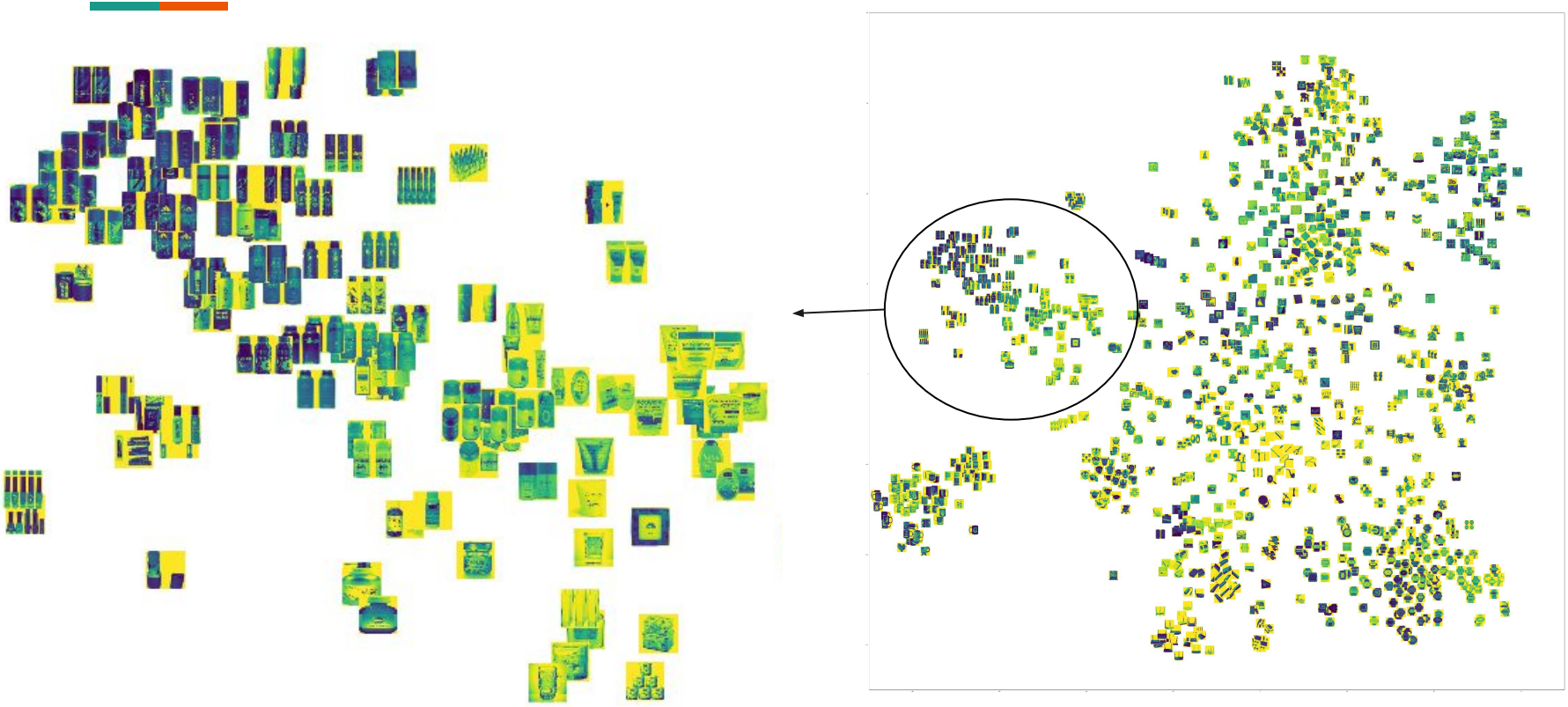

Hue: Clusters k-means




Adjusted Rand Score : 0.2895916607889346

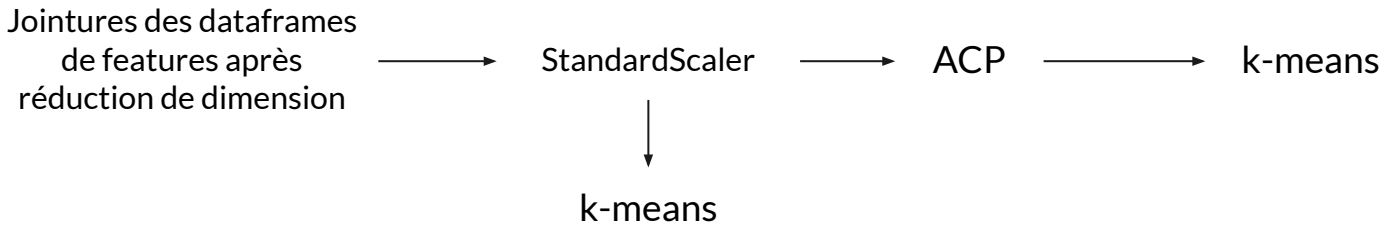
# 5. Alternative Images Feature Extraction : CNN

## c. Projection t-SNE avec images



# 6. Combinaison features Texte & Images

## a. Méthodologie




Comparaison & Evaluation :

t-SNE (Hue: Catégories “cibles”) ⟷ t-SNE (Hue: Clusters)

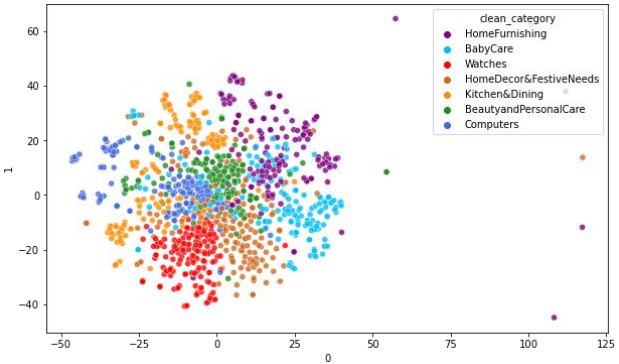
# 6. Combinaison features Texte & Images

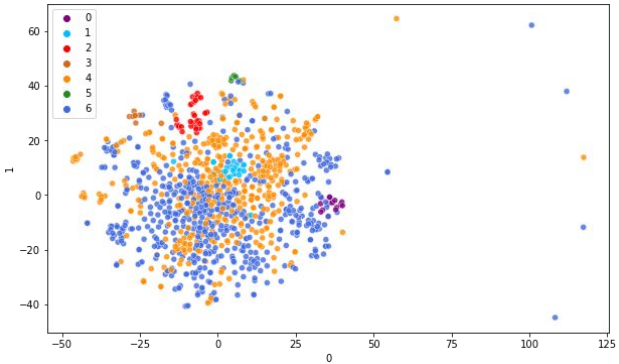
## b. Comparaison des performances

Hue: Catégories “cibles”

Hue: Clusters k-means

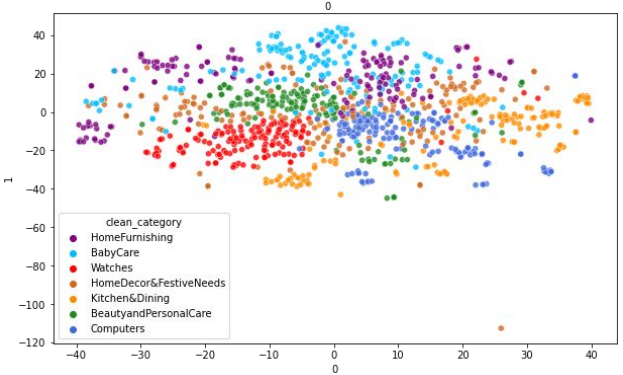
Avant ACP
308 Features

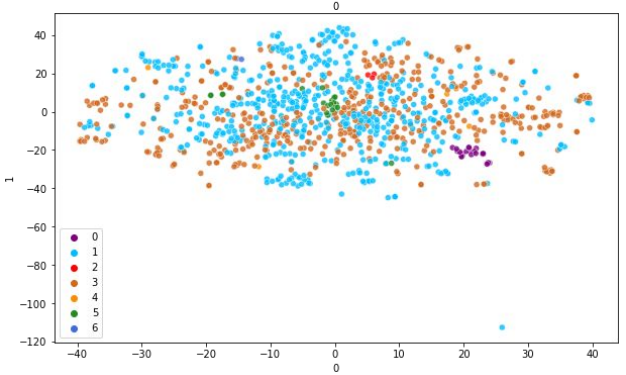



Adjusted Rand Score :
0.031698616150308764

Après ACP
235 Features




Adjusted Rand Score :
0.017041886632577986

# 7. Conclusion & Perspectives

Bons résultats :

- Etude de faisabilité validée
- Bon résultats sur les features prises séparément :
  - Texte : LDA
  - Images : CNN

Perspectives :

- Algorithme qui combine les résultats des 2 approches
- Algorithme supervisé quand plus de données à disposition