Formation Data Scientist OpenClassrooms

Projet 6: Classifiez automatiquement des biens de consommation

Etudiant : Monine Chan

Evaluateur : Patrick Kamnang Wanko

Lundi 29 Août 2022

PLAN

1. Présentation de la problématique et du jeu de données

2. Explication des prétraitements et des résultats du clustering

3. Conclusion sur la faisabilité du moteur de classification et recommandations pour sa création éventuelle

PLAN

1. Présentation de la problématique et du jeu de données

2. Explication des prétraitements et des résultats du clustering

3. Conclusion sur la faisabilité du moteur de classification et recommandations pour sa création éventuelle

1. PRÉSENTATION DE LA PROBLÉMATIQUE

CONTEXTE

- L'entreprise « Place de Marché » souhaite créer une marketplace e-commerce.

- Chaque vendeur poste une description et une photo de leur article mais l'attribution de l'article dans une catégorie est faite manuellement (peu fiable) : le but est d'automatiser cette tâche pour améliorer l'expérience utilisateur des vendeurs.

- Le but de ce projet est de :
  - Réaliser un prétraitement des descriptions (texte) des produits et de leurs images,
  - Appliquer un algorithme de réduction de dimension,
  - Effectuer un clustering et confirmer la similarité entre catégories réelles et clusters.

1. PRÉSENTATION DU JEU DE DONNÉES

VUE D'ENSEMBLE

- Le jeu de données comporte 1050 articles (lignes) qui correspondent aux 1050 images au format .jpg
- Il y a 15 colonnes : la colonne «description» est celle qu'on utilisera pour extraire les features pour la partie NLP.
- Le jeu de données est bien rempli, les items «na» sont absents de la colonne description.

```
Entrée [11]: df_flipkart.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1050 entries, 0 to 1049
Data columns (total 15 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   uniq_id                 1050 non-null   object
 1   crawl_timestamp         1050 non-null   object
 2   product_url             1050 non-null   object
 3   product_name            1050 non-null   object
 4   product_category_tree   1050 non-null   object
 5   pid                     1050 non-null   object
 6   retail_price            1049 non-null   float64
 7   discounted_price        1049 non-null   float64
 8   image                   1050 non-null   object
 9   is_FK_Advantage_product 1050 non-null   bool
 10  description             1050 non-null   object
 11  product_rating          1050 non-null   object
 12  overall_rating          1050 non-null   object
 13  brand                   712 non-null    object
 14  product_specifications  1049 non-null   object
dtypes: bool(1), float64(2), object(12)
memory usage: 116.0+ KB
```

```
Entrée [12]: df_flipkart.isna().sum()

Out[12]: uniq_id                   0
         crawl_timestamp           0
         product_url               0
         product_name              0
         product_category_tree     0
         pid                       0
         retail_price              1
         discounted_price          1
         image                     0
         is_FK_Advantage_product   0
         description               0
         product_rating            0
         overall_rating            0
         brand                   338
         product_specifications    1
         dtype: int64
```

1. PRÉSENTATION DU JEU DE DONNÉES

SÉLECTION DES CATÉGORIES D'ARTICLES

- On va utiliser la colonne «product_category_tree» pour trouver en combien de catégories diviser les articles.
- On observe que le premier niveau de «product_category_tree» permet de diviser les 1050 produits en 7 catégories différentes :

```
Entrée [15]: display(df_cat_level.nunique(), df_cat_level.sample(3))
```

```
cat_level_0      7
cat_level_1     62
cat_level_2    243
cat_level_3    460
cat_level_4    596
cat_level_5    633
dtype: int64
```

	cat_level_0	cat_level_1	cat_level_2	cat_level_3	cat_level_4	cat_level_5
850	Computers	Computers/Laptop Accessories	Computers/Laptop Accessories/USB Gadgets	Computers/Laptop Accessories/USB Gadgets/Techo...	Computers/Laptop Accessories/USB Gadgets/Techo...	Computers/Laptop Accessories/USB Gadgets/Techo...
424	Beauty and Personal Care	Beauty and Personal Care/Hair Care	Beauty and Personal Care/Hair Care/Hair Care A...	Beauty and Personal Care/Hair Care/Hair Care A...	Beauty and Personal Care/Hair Care/Hair Care A...	Beauty and Personal Care/Hair Care/Hair Care A...
96	Home Decor & Festive Needs	Home Decor & Festive Needs/Table Decor & Handi...	Home Decor & Festive Needs/Table Decor & Handi...	Home Decor & Festive Needs/Table Decor & Handi...	Home Decor & Festive Needs/Table Decor & Handi...	Home Decor & Festive Needs/Table Decor & Handi...

PLAN

1. Présentation de la problématique et du jeu de données

2. Explication des prétraitements et des résultats du clustering

3. Conclusion sur la faisabilité du moteur de classification et recommandations pour sa création éventuelle

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

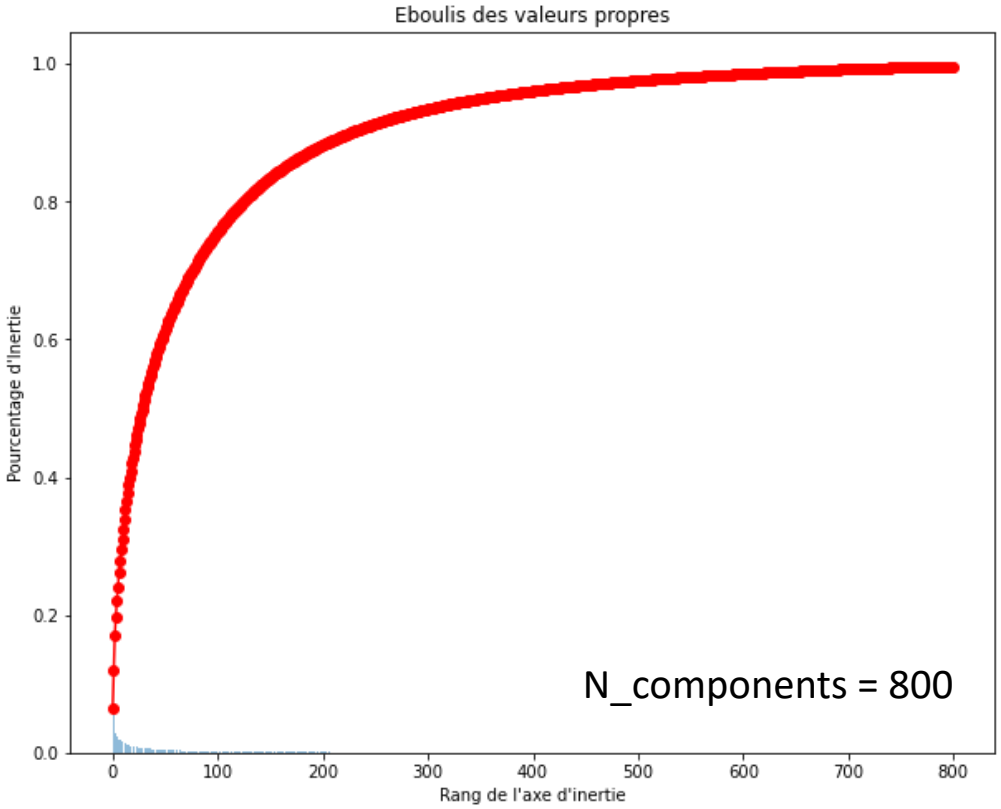
2.1. NLP – COUNTVECTORIZER : PRÉ-TRAITEMENT

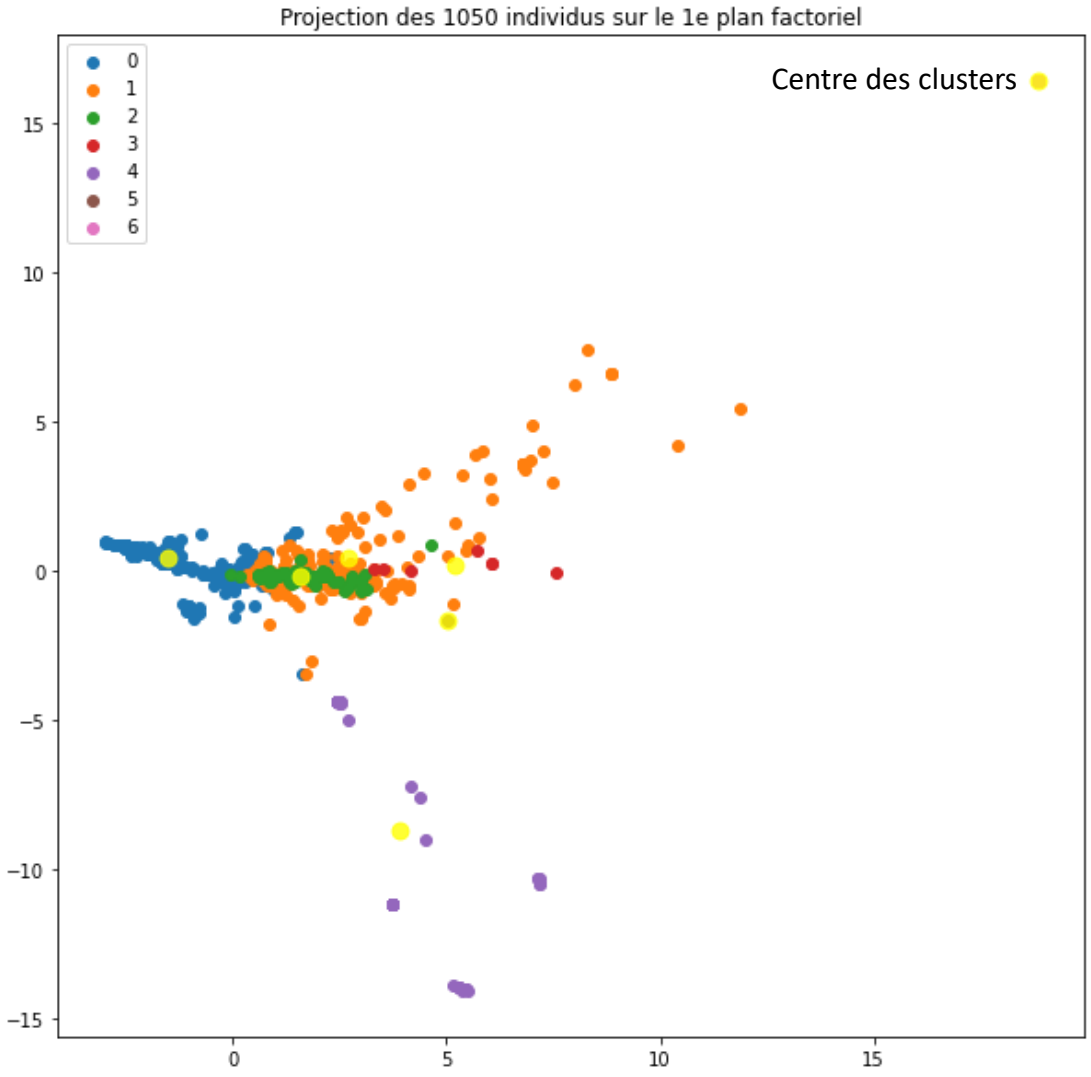
➢ Pour le prétraitement du texte, afin d'appliquer CountVectorizer, on effectue les opérations suivantes :

- ❑ Tokénisation,
- ❑ Suppression des stop-words et de la ponctuation,
- ❑ Lemmatisation,
- ❑ Passage en minuscules.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.1. NLP – COUNTVECTORIZER : PCA ET CLUSTERING





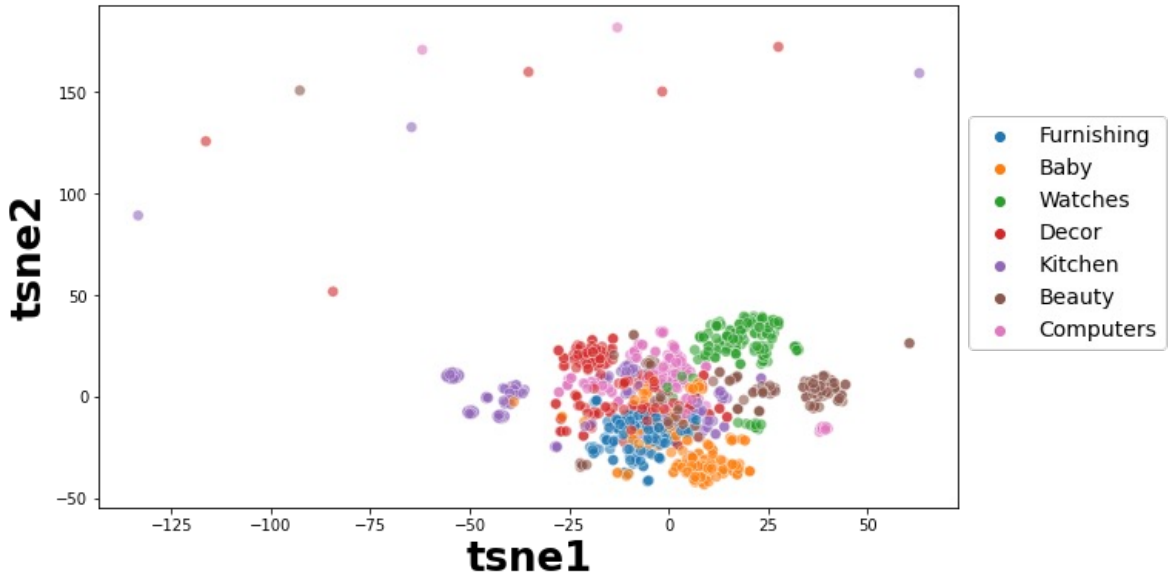
- Pour la réduction de dimension, on utilise une ACP.
- Le clustering est effectué avec un algorithme Kmeans. Le nombre de cluster est défini à l'avance : 7 car on sait qu'on doit trouver 7 catégories.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

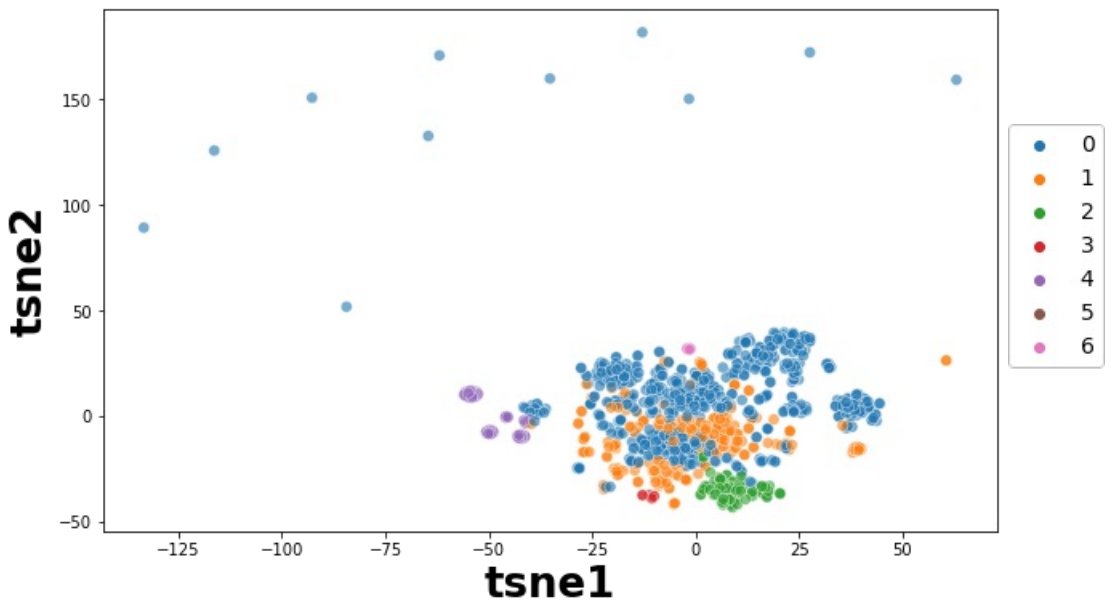
2.1. NLP – COUNTVECTORIZER : ARI ET T-SNE

ARI = 0.0548

TSNE selon les vraies classes - CountVectorizer



TSNE selon les clusters - CountVectorizer



- L'ARI est faible ce qui dénote un manque trop important de similarité entre vraies classes et clusters.
- CountVectorizer ne permettra pas de réaliser un moteur de classification.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

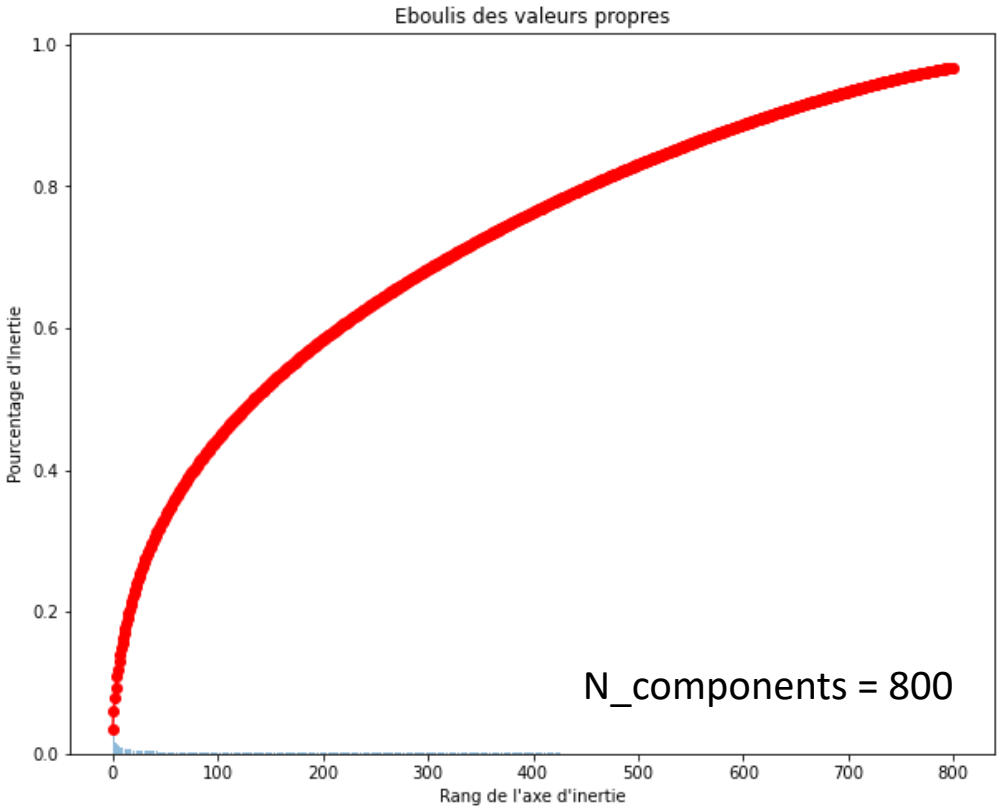
2.2. NLP – TF-IDF : PRÉ-TRAITEMENT

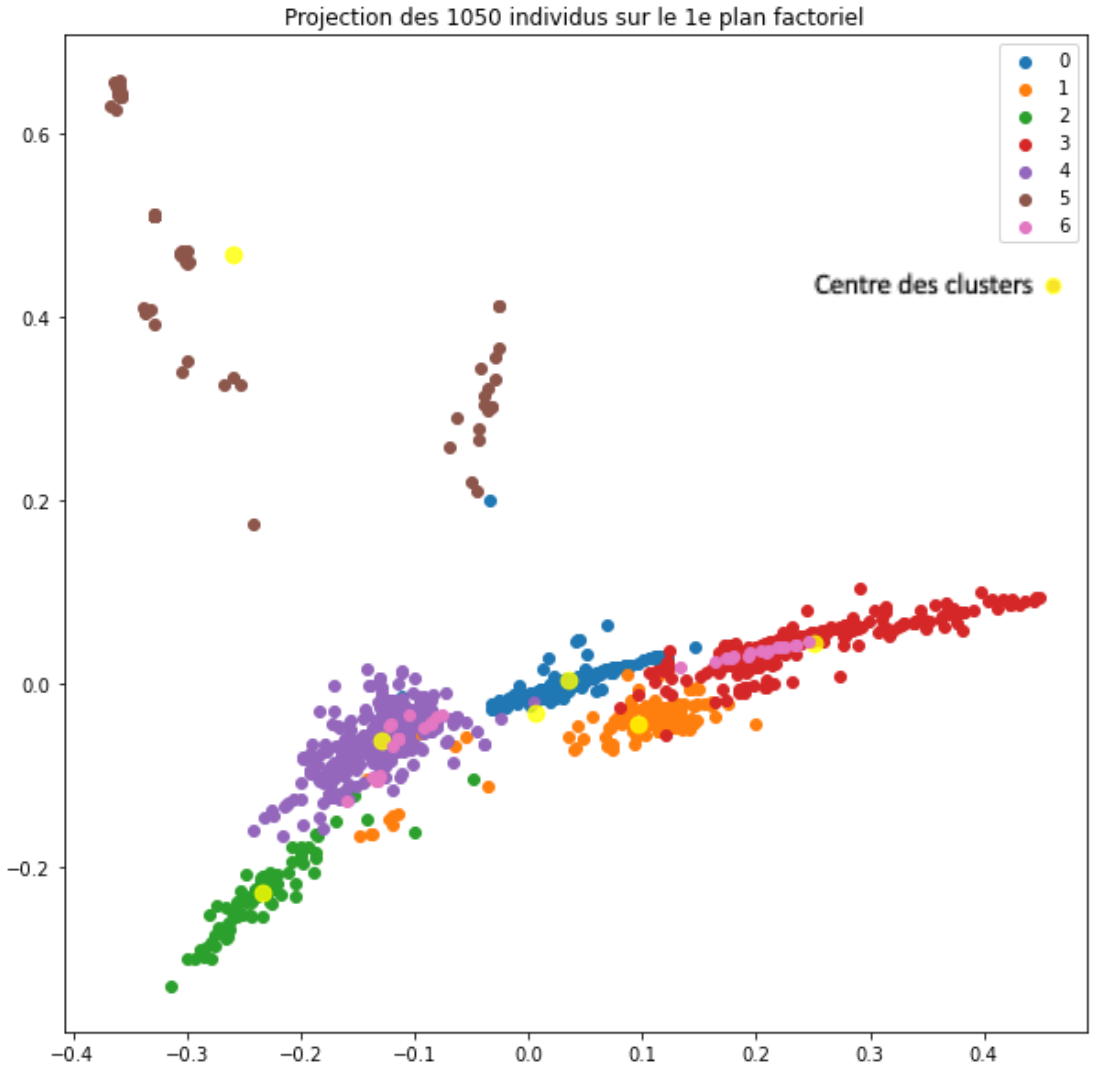
➢ Pour le prétraitement du texte, afin d'appliquer Tf-Idf, on effectue les opérations suivantes:

- ❑ Tokénisation,
- ❑ Suppression des stop-words et de la ponctuation,
- ❑ Lemmatisation,
- ❑ Passage en minuscules.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.2. NLP – TF-IDF : PCA ET CLUSTERING



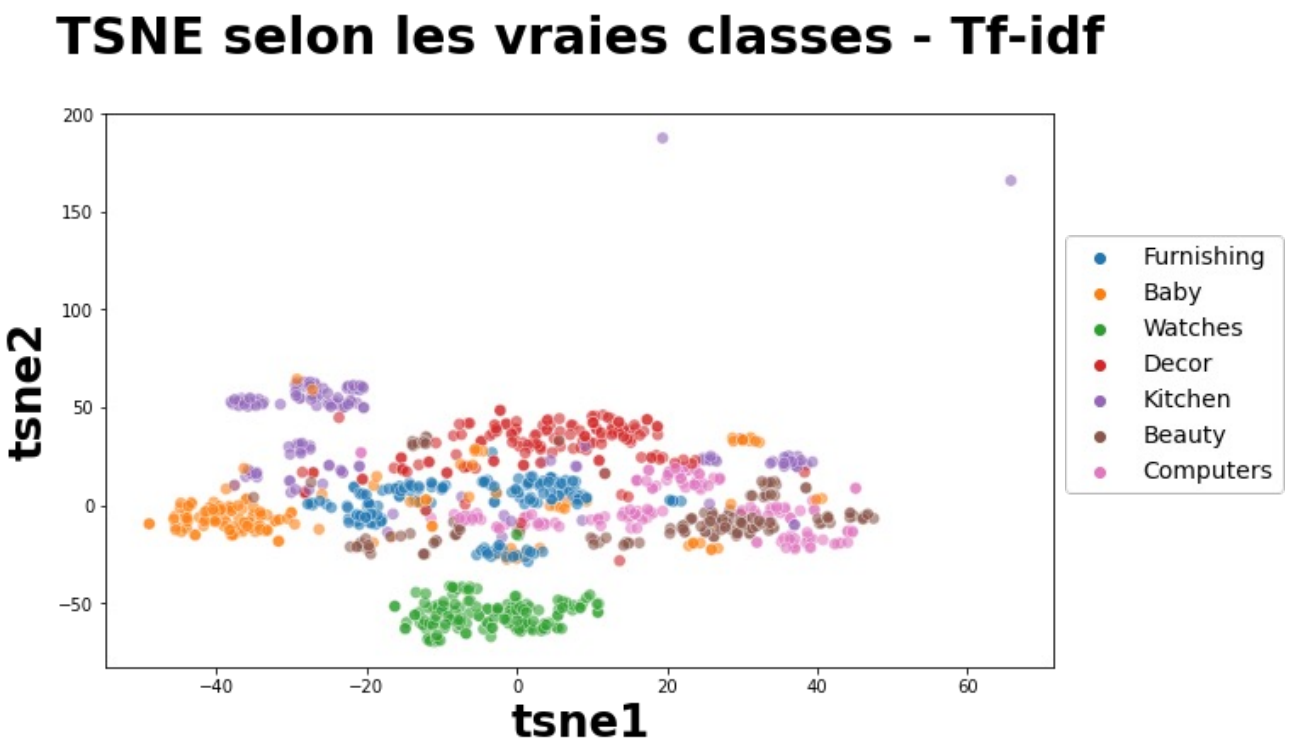


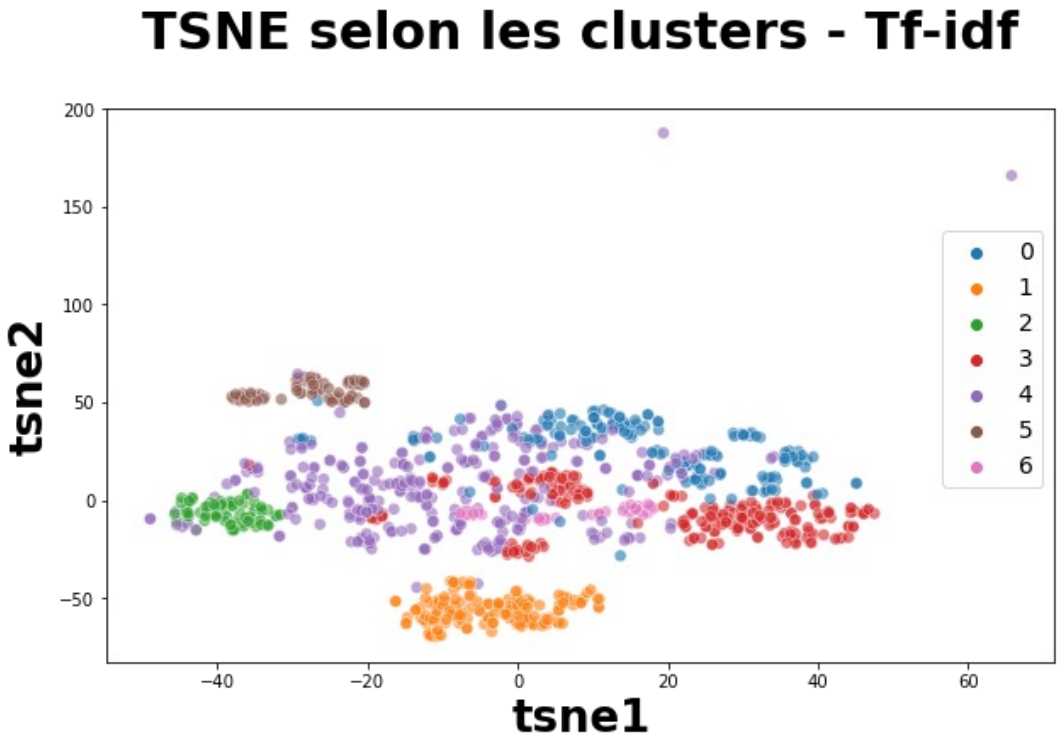
- Pour la réduction de dimension, on utilise une ACP.
- Le clustering est effectué avec un algorithme Kmeans. Le nombre de cluster est défini à l'avance : 7 car on sait qu'on doit trouver 7 catégories.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.2. NLP – TF-IDF : ARI ET T-SNE

ARI = 0.287





- L'ARI est bien meilleur que CountVectorizer.
- Néanmoins, TF-IDF n'est pas suffisamment performant pour réaliser un moteur de classification efficace.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

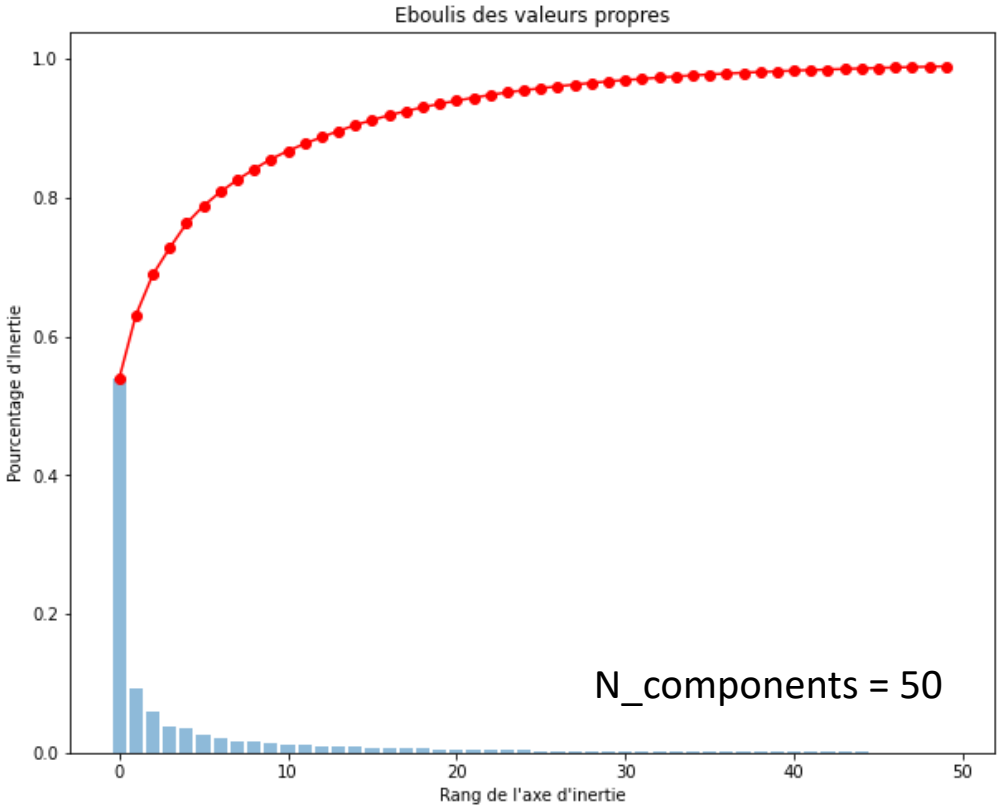
2.3. NLP – WORD2VEC : PRÉ-TRAITEMENT

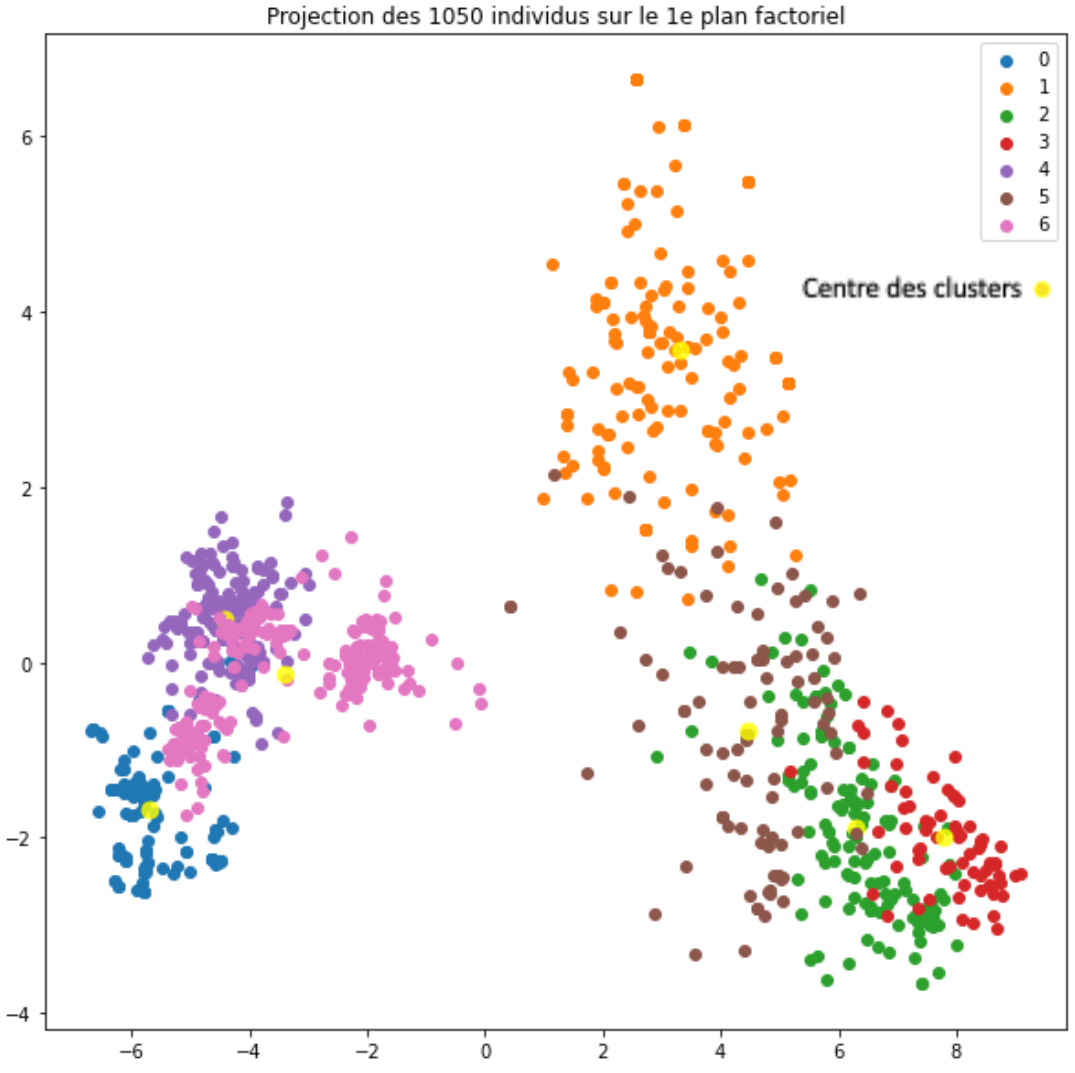
➢ Pour le prétraitement du texte, afin d'appliquer Word2Vec, on effectue les opérations suivantes:

- ❑ Tokénisation,
- ❑ Suppression des stop-words et de la ponctuation,
- ❑ Lemmatisation,
- ❑ Passage en minuscules.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.3. NLP – WORD2VEC : PCA ET CLUSTERING



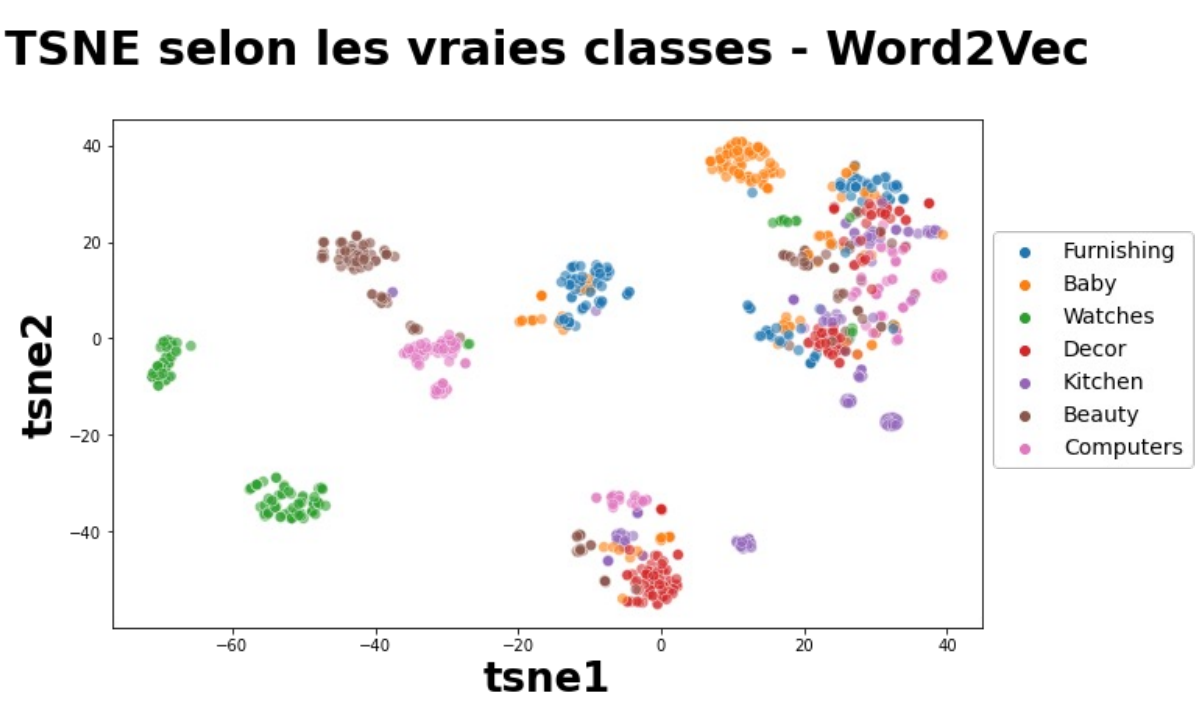


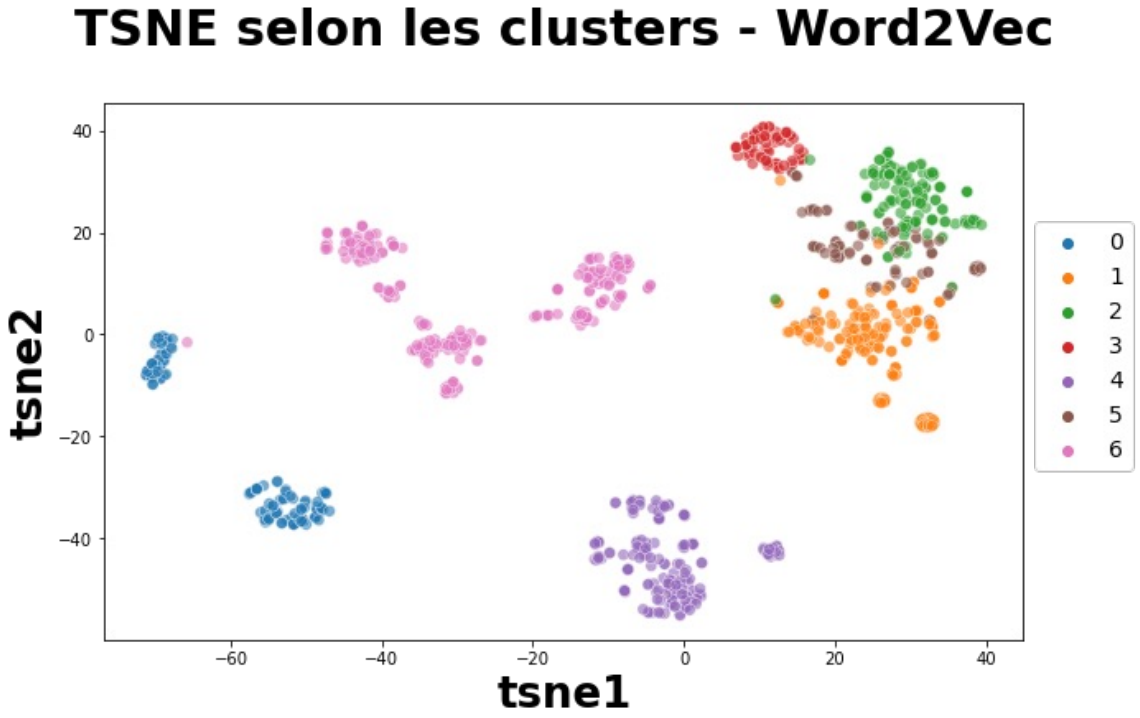
- Pour la réduction de dimension, on utilise une ACP.
- Le clustering est effectué avec un algorithme Kmeans. Le nombre de cluster est défini à l'avance : 7 car on sait qu'on doit trouver 7 catégories.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.3. NLP – WORD2VEC : ARI ET T-SNE

ARI = 0.251





- L'ARI de Word2Vec (0.251) est un moins bon que Td-Idf (0.287).
- Word2Vec n'est pas suffisamment performant pour réaliser un moteur de classification efficace.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

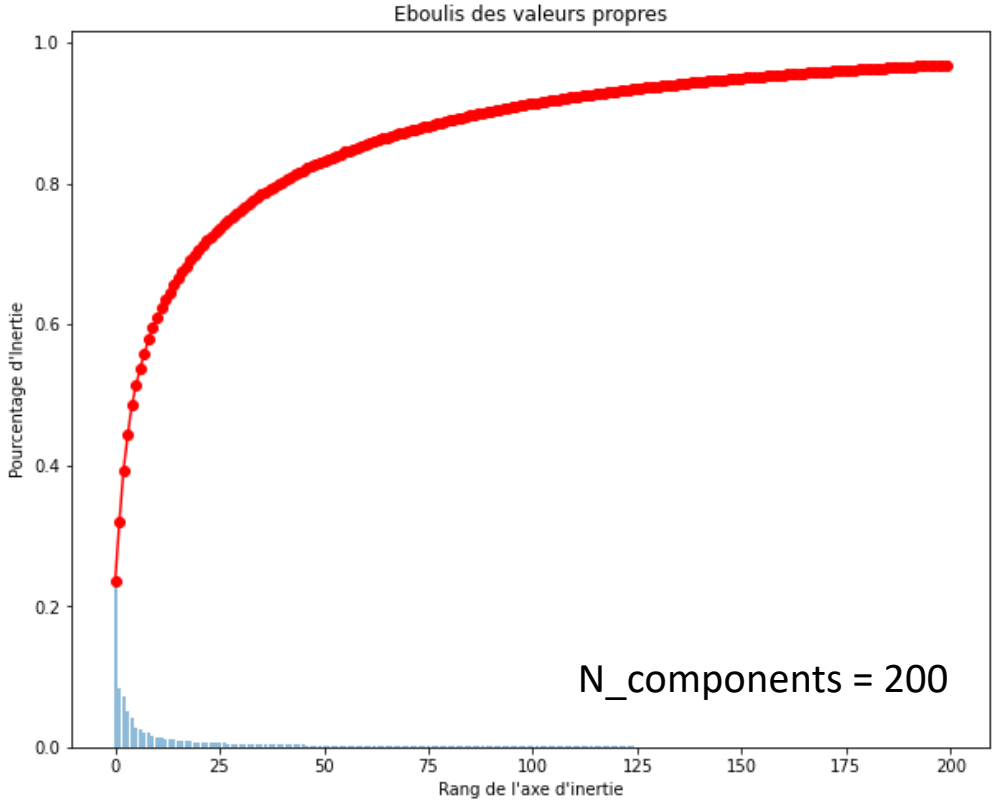
2.4. NLP – BERT : PRÉ-TRAITEMENT

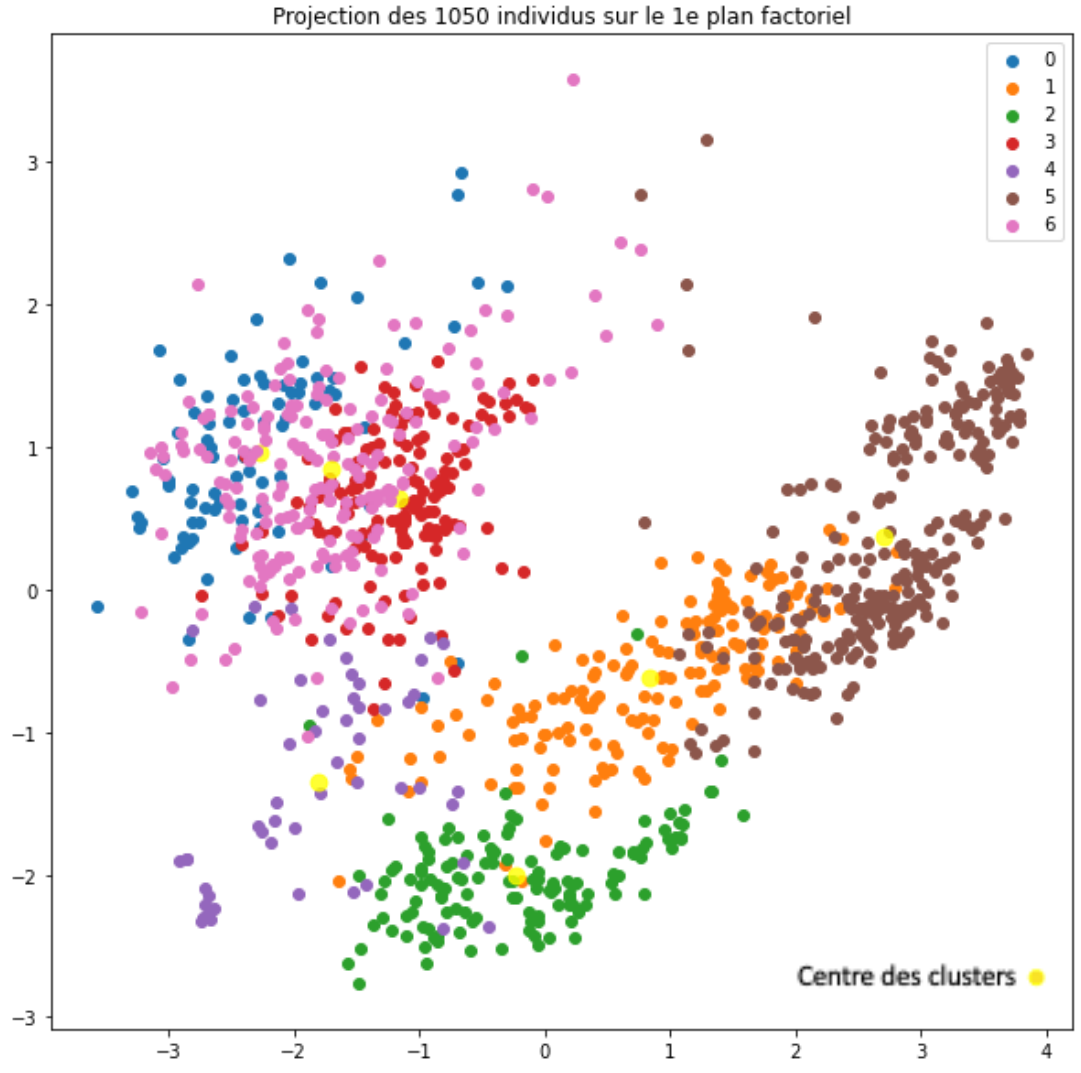
➢ Pour le prétraitement du texte, afin d'appliquer BERT, on effectue les opérations suivantes:

- ❑ Tokénisation,
- ❑ Passage des mots en minuscules.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.4. NLP – BERT : PCA ET CLUSTERING



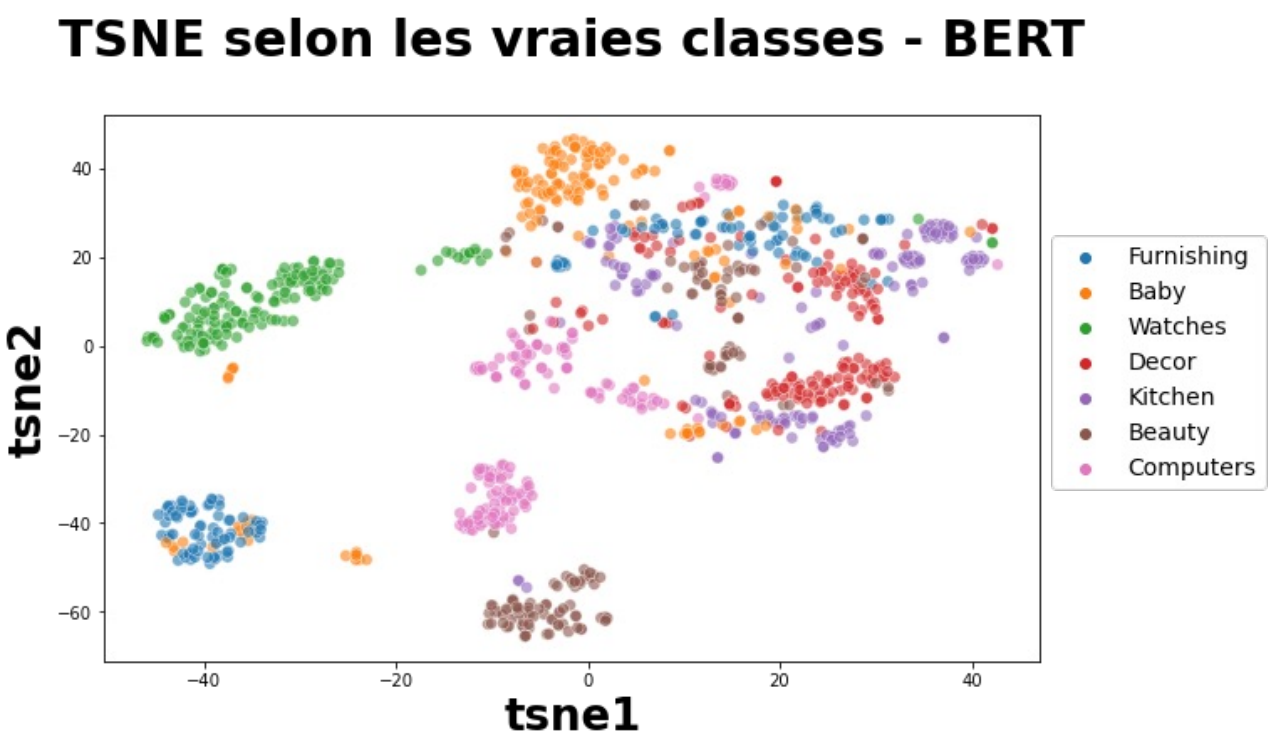


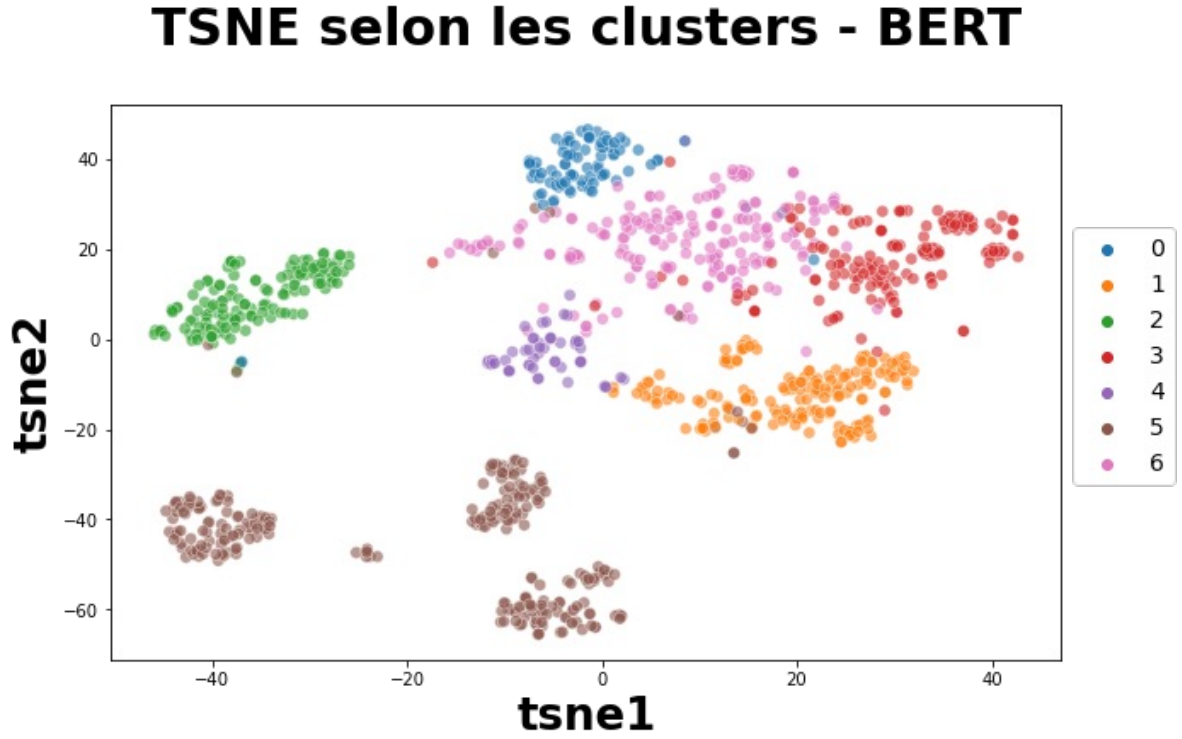
- Pour la réduction de dimension, on utilise une ACP.
- Le clustering est effectué avec un algorithme Kmeans. Le nombre de cluster est défini à l'avance : 7 car on sait qu'on doit trouver 7 catégories.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.4. NLP – BERT : ARI ET T-SNE

ARI = 0.267





- L'ARI de BERT (0.267) est moins bon que Td-Idf (0.287).
- BERT n'est pas suffisamment performant pour réaliser un moteur de classification efficace.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

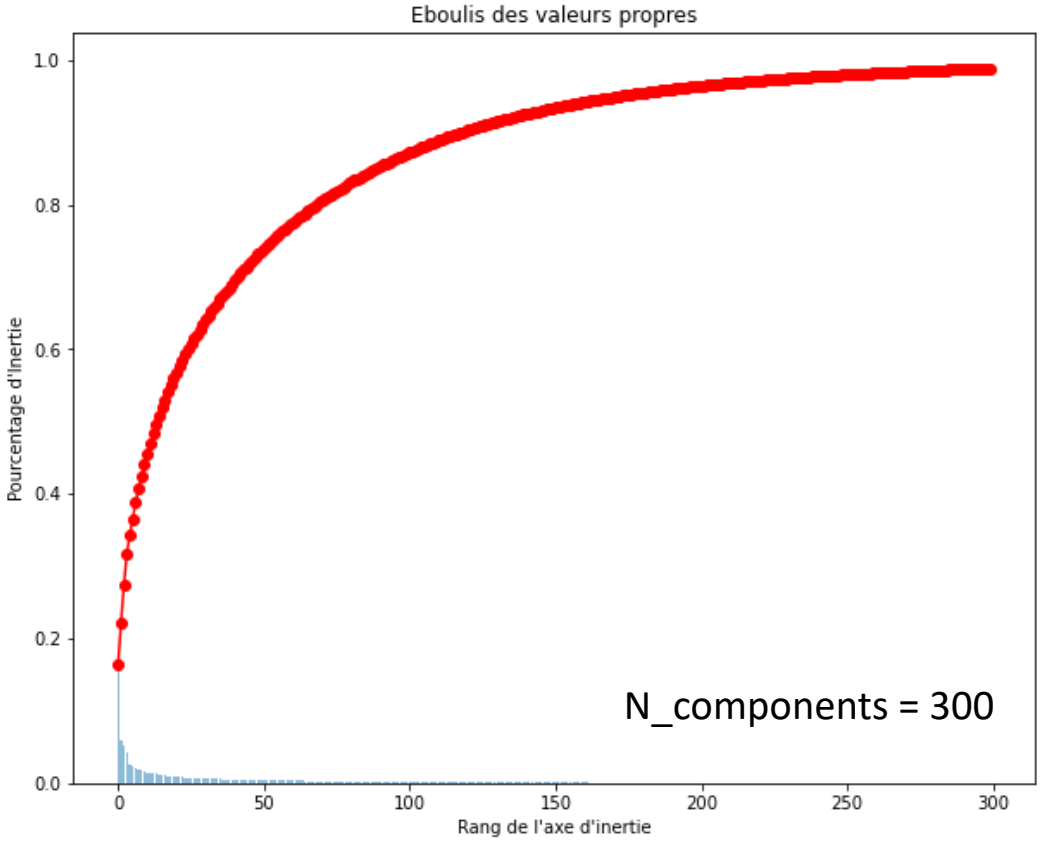
2.5. NLP – USE : PRÉ-TRAITEMENT

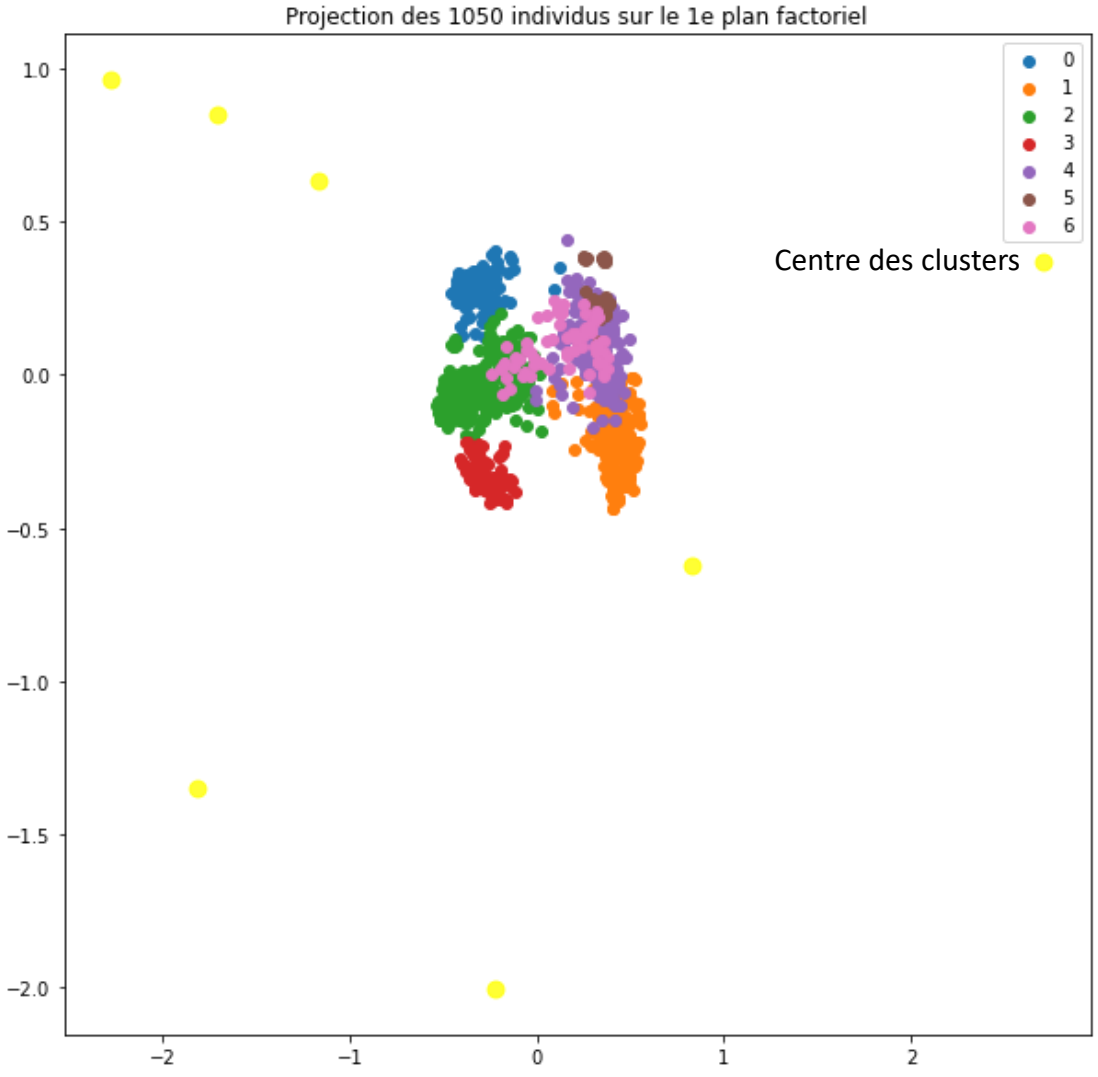
➢ Pour le prétraitement du texte, afin d'appliquer USE, on effectue les opérations suivantes:

- ❑ Tokénisation,
- ❑ Passage des mots en minuscules.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.5. NLP – USE : PCA ET CLUSTERING



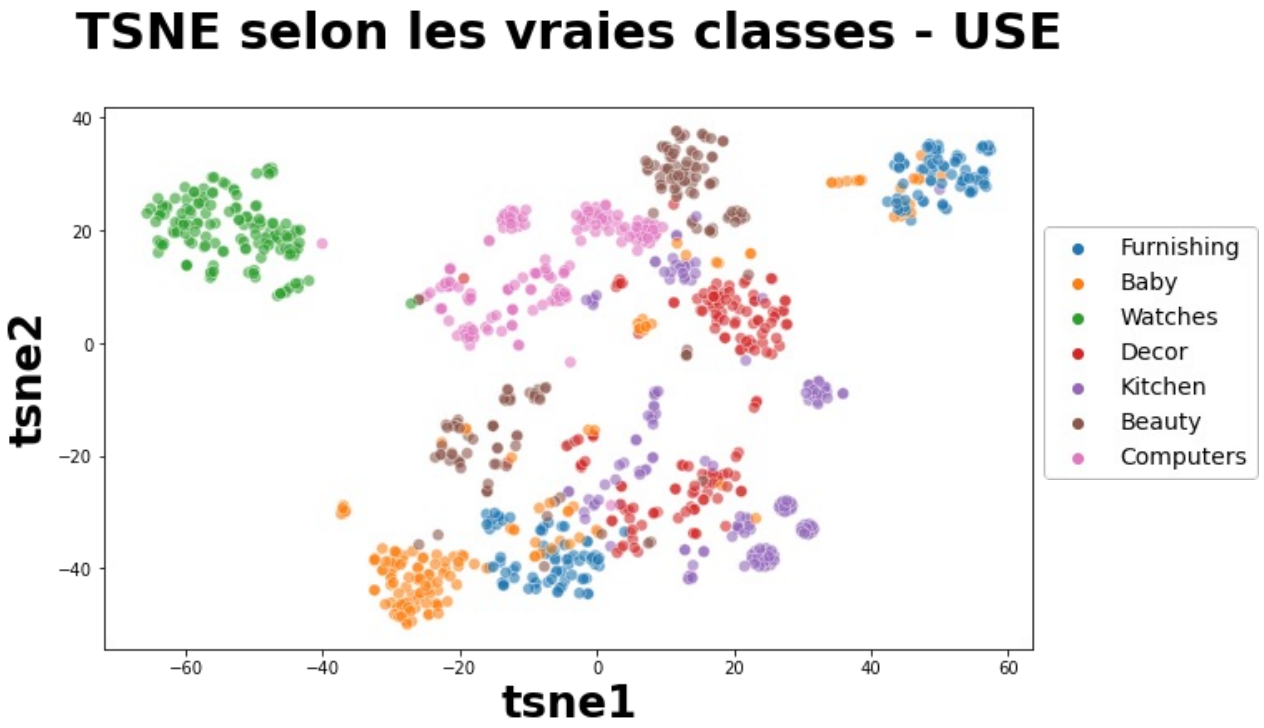


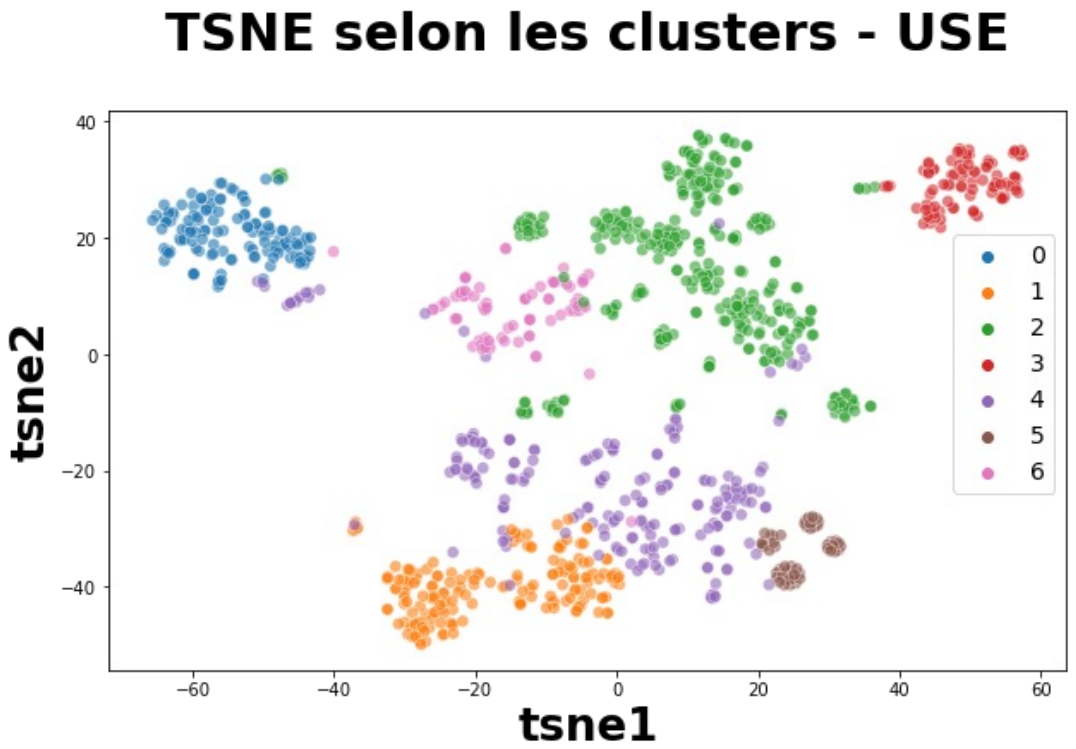
- Pour la réduction de dimension, on utilise une ACP.
- Le clustering est effectué avec un algorithme Kmeans. Le nombre de cluster est défini à l'avance : 7 car on sait qu'on doit trouver 7 catégories.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.5. NLP – USE : ARI ET T-SNE

ARI = 0.332





- L'ARI de USE (0.332) est le meilleur des algorithmes de NLP testés jusqu'à présent.
- Néanmoins, USE n'est pas suffisamment performant pour réaliser un moteur de classification efficace.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

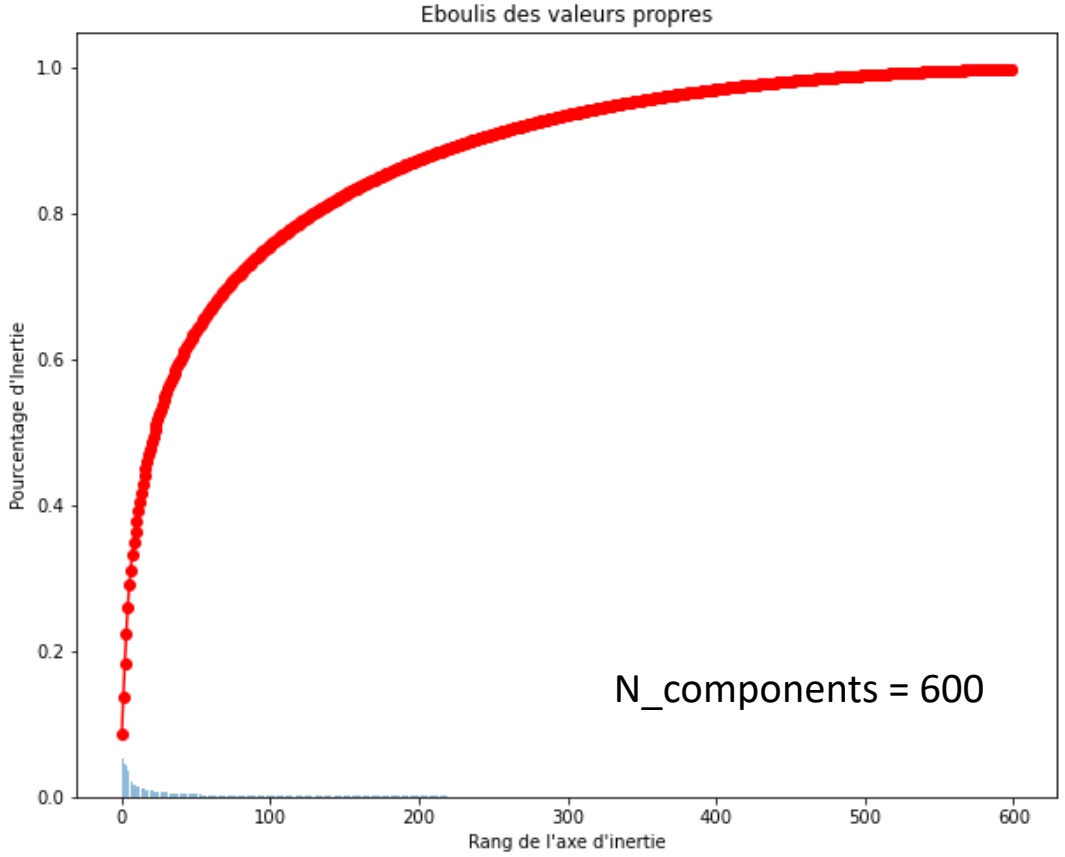
2.6. IMAGES – SIFT : PRÉ-TRAITEMENT

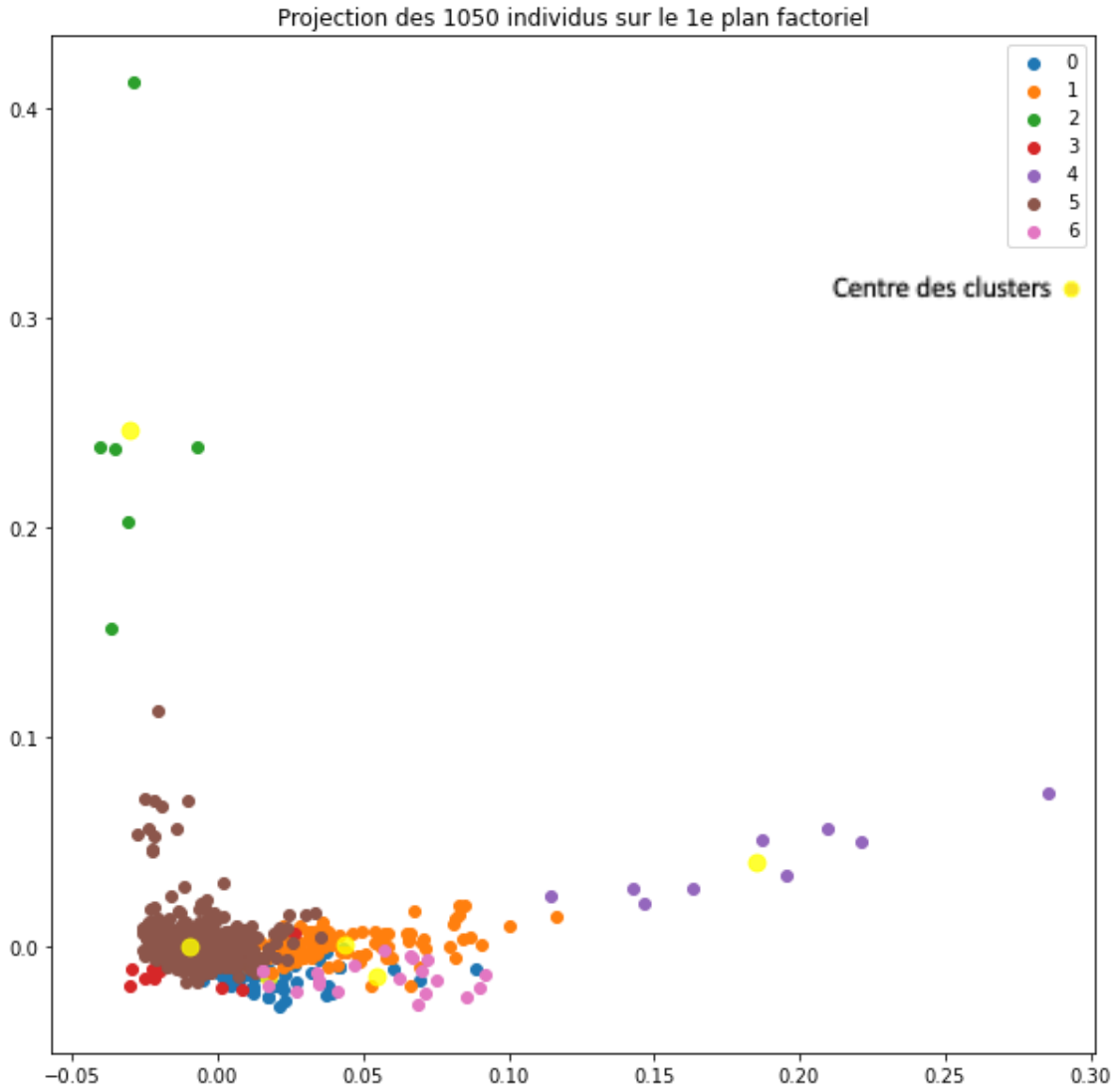
➢ Pour le prétraitement des images, afin d'appliquer SIFT, on effectue les opérations suivantes:

- ❑ Conversion des images en gris,
- ❑ Redimensionnement en 224x224,
- ❑ Egalisation de l'histogramme,
- ❑ Utilisation d'un filtre GaussianBlur.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.5. IMAGES – SIFT : PCA ET CLUSTERING



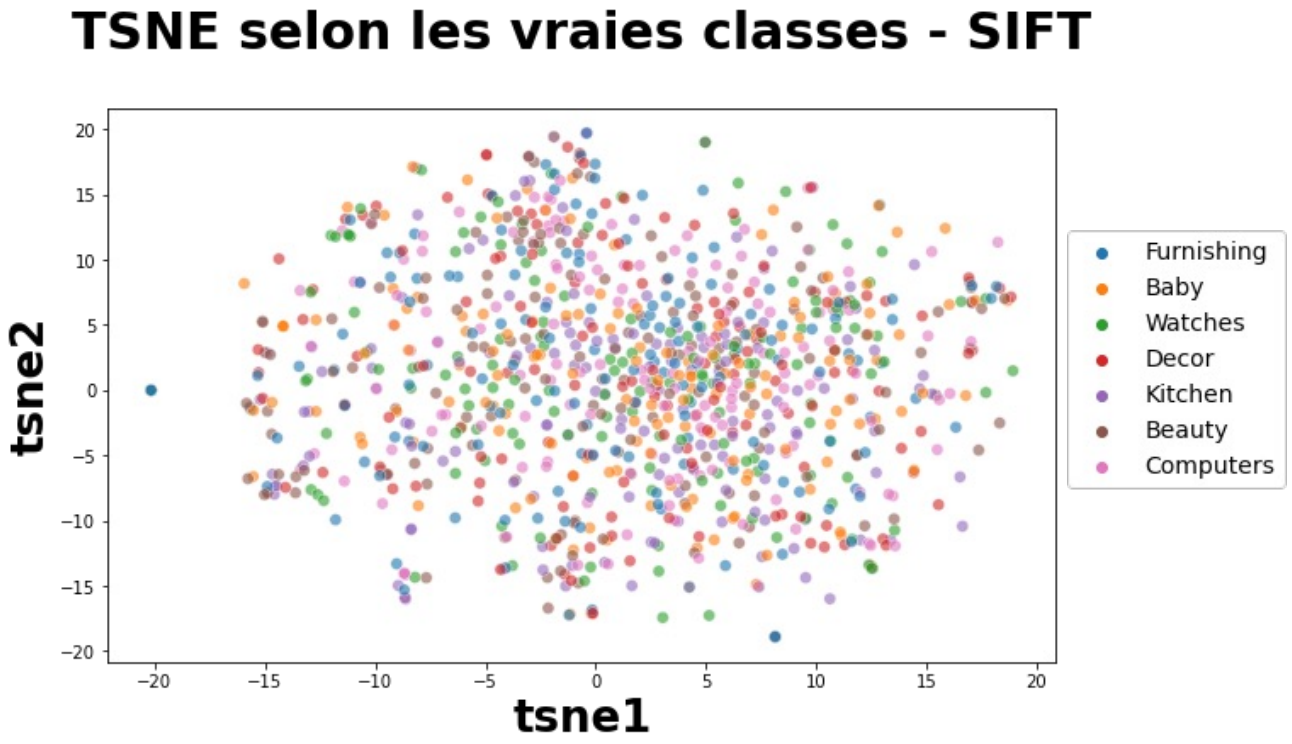


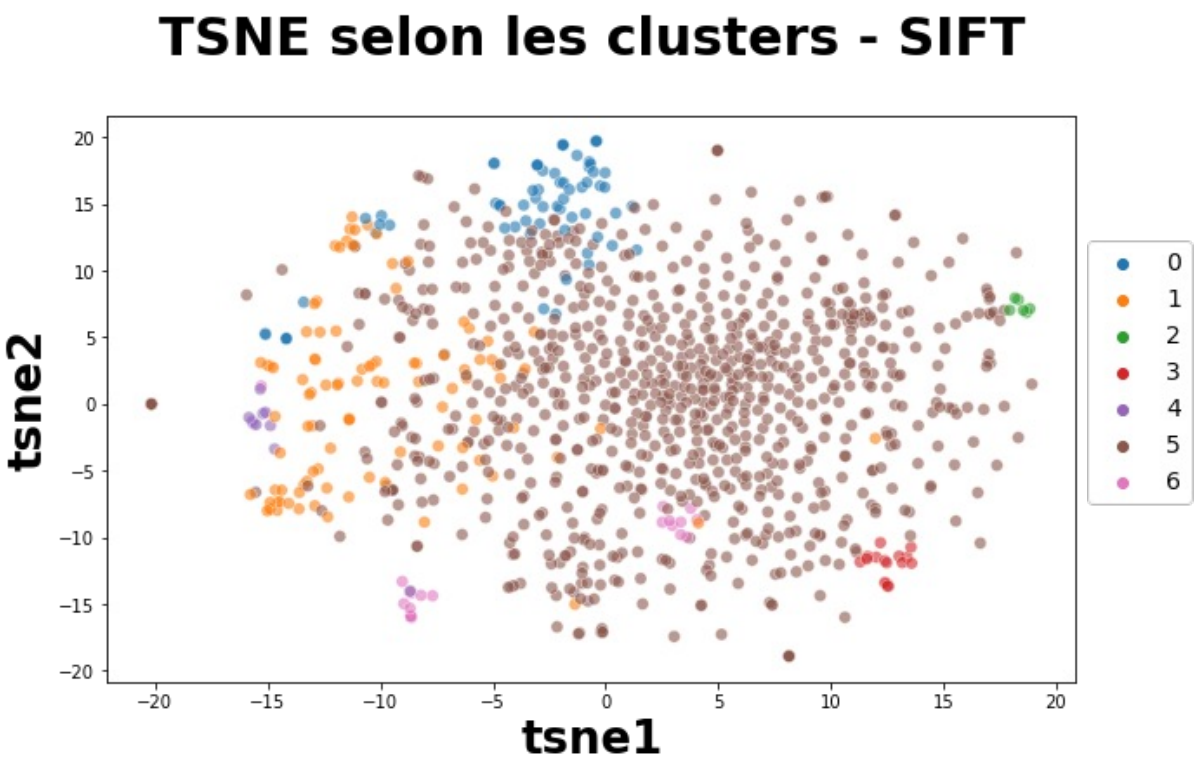
- Pour la réduction de dimension, on utilise une ACP.
- Le clustering est effectué avec un algorithme Kmeans. Le nombre de cluster est défini à l'avance : 7 car on sait qu'on doit trouver 7 catégories.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.5. IMAGES – SIFT : ARI ET T-SNE

ARI = 4.15^{e-05}





- L'ARI de SIFT est très faible.
- SIFT ne permettra pas de réaliser un moteur de classification.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

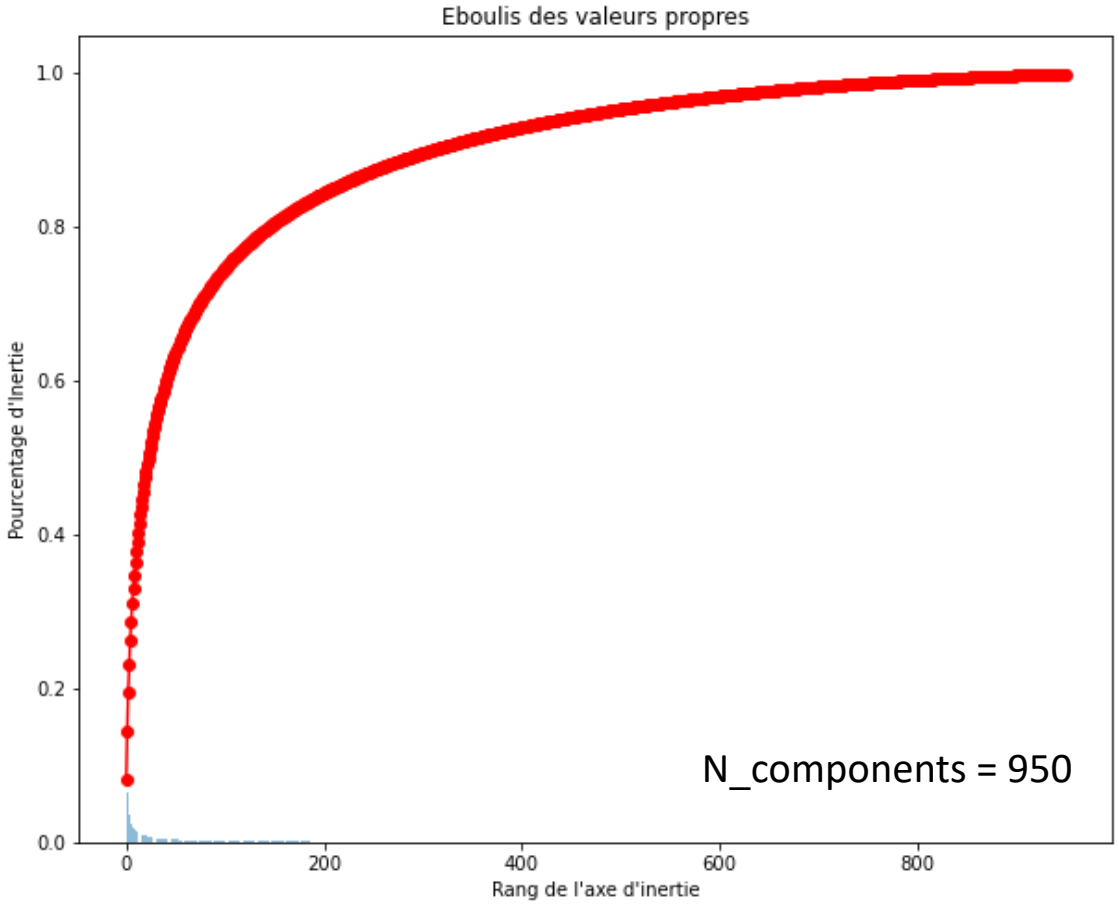
2.7. IMAGES – CNN VGG16 : PRÉ-TRAITEMENT

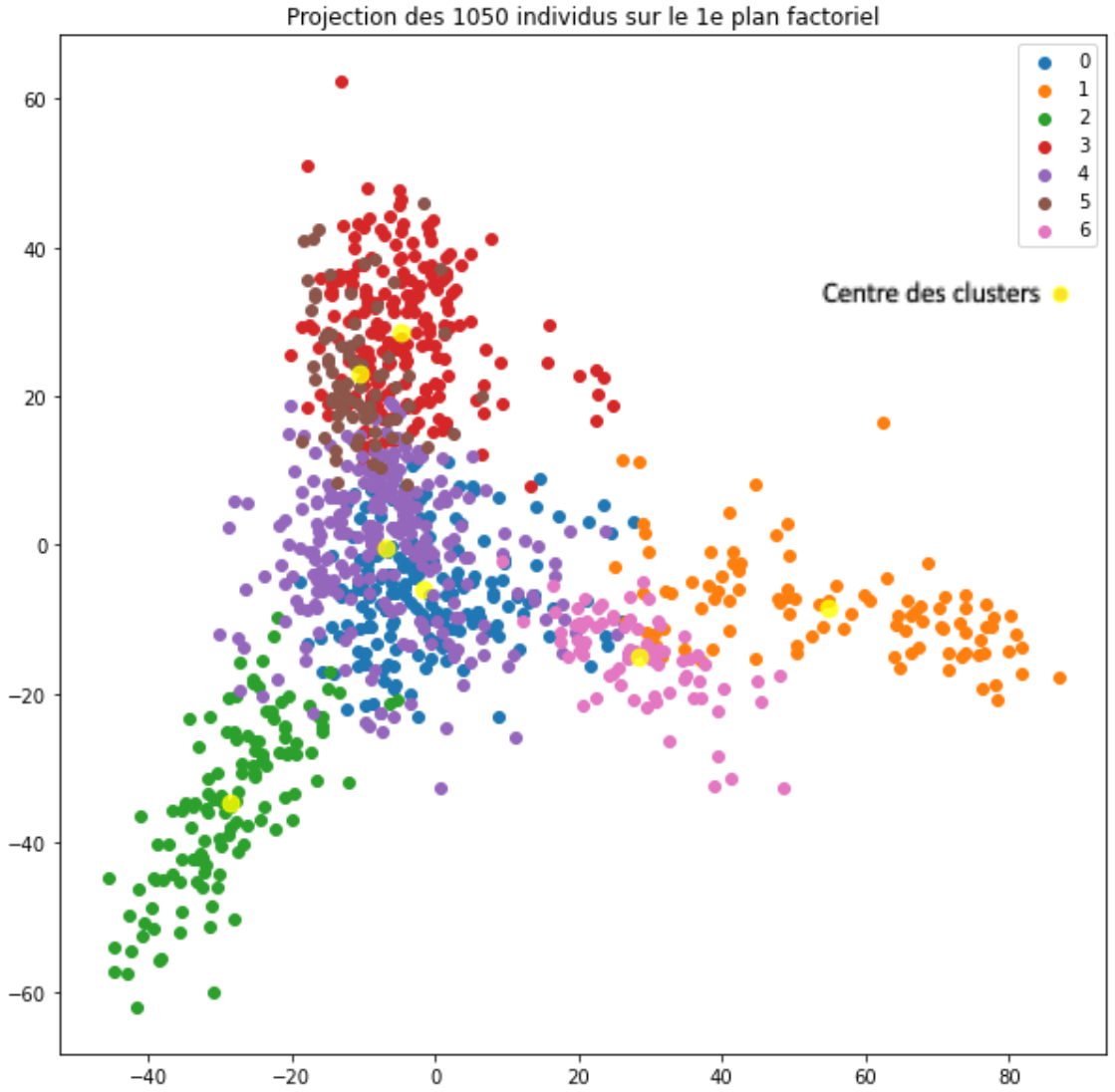
- Pour le prétraitement des images, afin d'appliquer CNN VGG16, on effectue les opérations suivantes:

  - ❑ Redimensionnement en 224x224.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.5. IMAGES – CNN VGG16 : PCA ET CLUSTERING



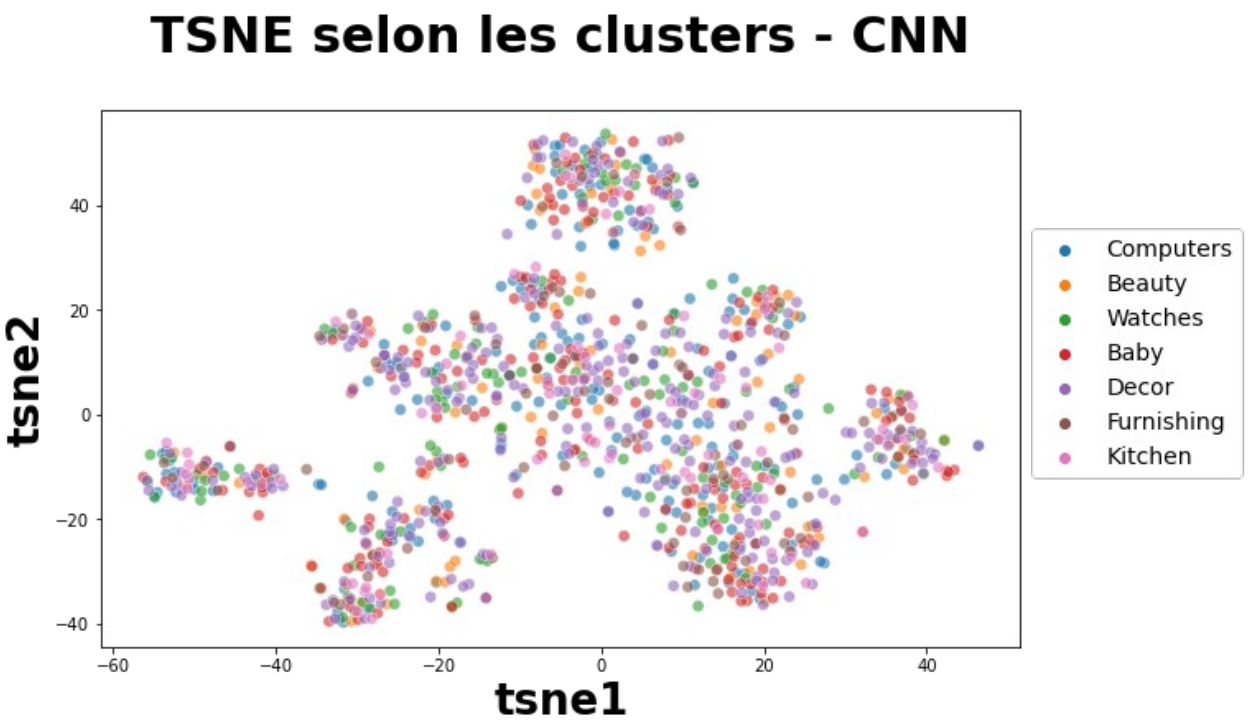


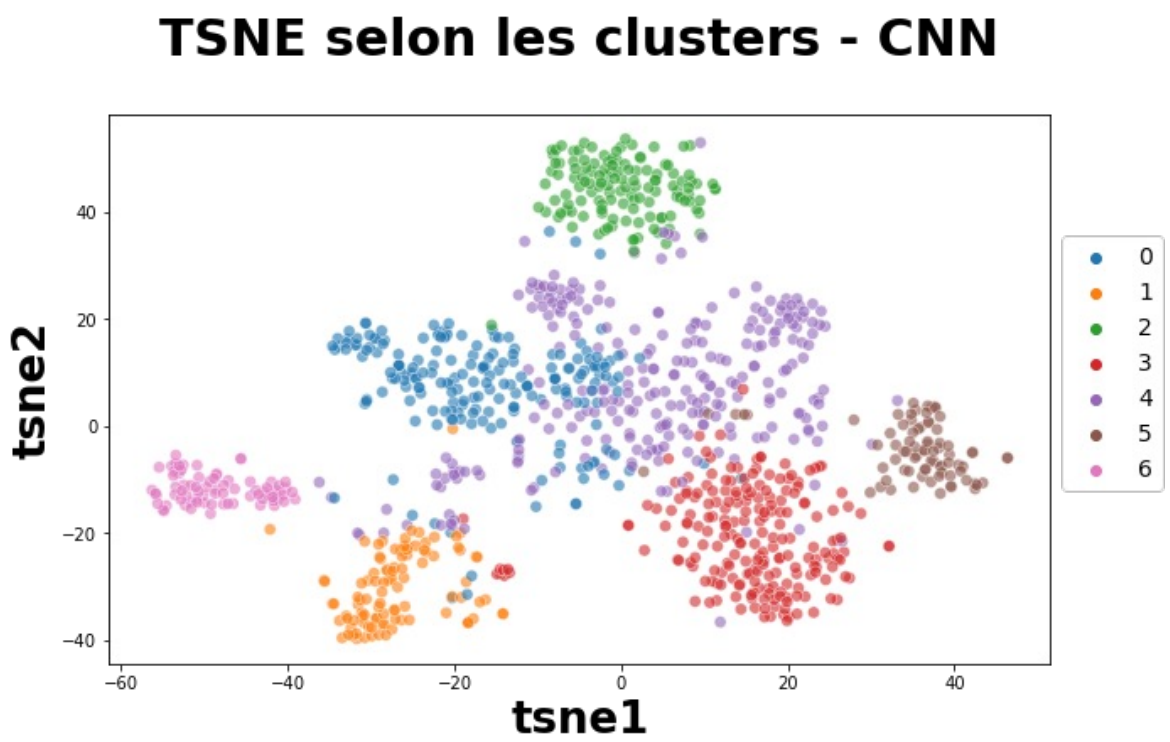
- Pour la réduction de dimension, on utilise une ACP.
- Le clustering est effectué avec un algorithme Kmeans. Le nombre de cluster est défini à l'avance : 7 car on sait qu'on doit trouver 7 catégories.

2. PRÉTRAITEMENTS ET RÉSULTATS DU CLUSTERING

2.5. IMAGES – CNN VGG16 : ARI ET T-SNE

ARI = 0.483





- L'ARI de CNN VGG16 est intéressant.
- CNN VGG16 semble être un bon candidat pour réaliser un moteur de classification.

PLAN

1. Présentation de la problématique et du jeu de données

2. Explication des prétraitements et des résultats du clustering

3. Conclusion sur la faisabilité du moteur de classification et recommandations pour sa création éventuelle

3. FRÉQUENCE DE MISE À JOUR

FAISABILITÉ DU MOTEUR DE CLASSIFICATION ET RECOMMENDATIONS

	Résumé de l'ARI pour les différents modèles						
Features	Texte					Image	
Algorithme	Word Vectorizer	Tf-Idf	Word2Vec	BERT	USE	SIFT	CNN VGG16
ARI	0.0548	0.287	0.251	0.267	0.332	**0.0000415**	**0.483**

- Le meilleur modèle pour les features texte est USE avec un ARI de 0.332.
- Le meilleur modèle pour les features image est CNN VGG16 un ARI de 0.483.
- **Conclusion** : on propose donc de créer un moteur de classification sur la base du modèle réseau de neurones **CNN VGG16**.
- **Recommendations pour la création du moteur de classification**:
  - Augmenter le nombre des images pour améliorer l'apprentissage.
  - Utiliser d'autre modèles pré-entrainés plus gros (exemple : Efficient Net B4).
  - Exploration du fine-tuning.
  - Image augmentation.

Merci de votre attention!