Projet 6 : Classifiez automatiquement des biens de consommation

Problématique

Problématique : Réaliser une première étude de faisabilité d'un moteur de classification d'articles, basé sur une image et une description, pour l'automatisation de l'attribution de la catégorie de l'article

Problématique

- L'entreprise « **Place du marché »** souhaite lancer une plate-forme de e-commerce.
- La classification des articles sur la plate-forme est faite à la main par les vendeurs, et est donc peu fiable.
- On souhaite vérifier la faisabilité d'un moteur de classification qui automatiserait cette tâche à partir de la photo et de la description des articles.

Présentation des données

- Les données de 1050 articles vendus sur la plate-forme de « **Place du marché »**
- Ces données contiennent entre autres : le nom de l'article, la description entrée par le vendeur et le nom du fichier contenant la photo de chaque article.
- Il y a également une colonne indiquant l'arbre de catégories de chaque produit.

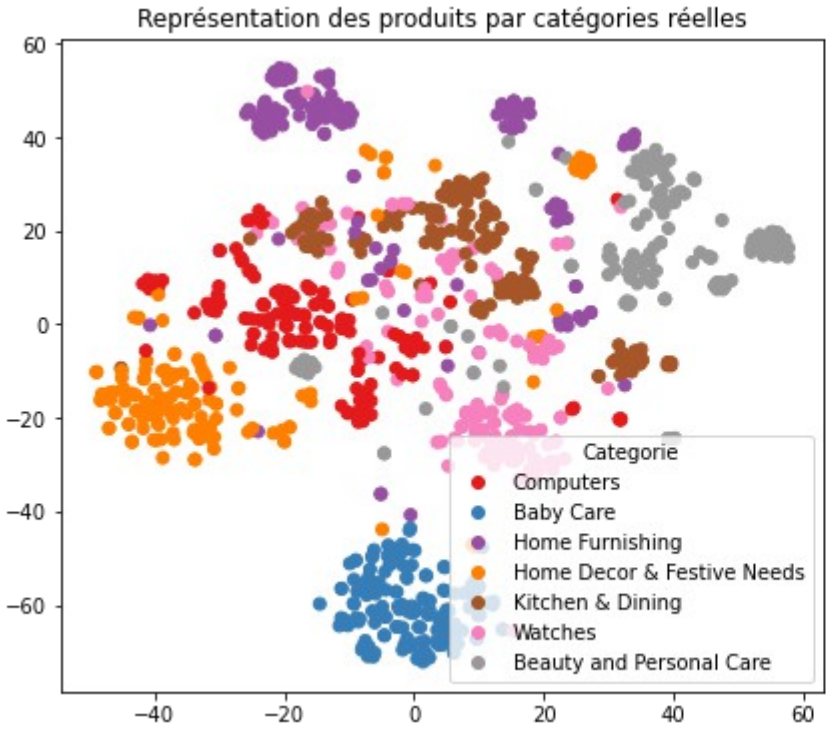
Prétraitement du texte

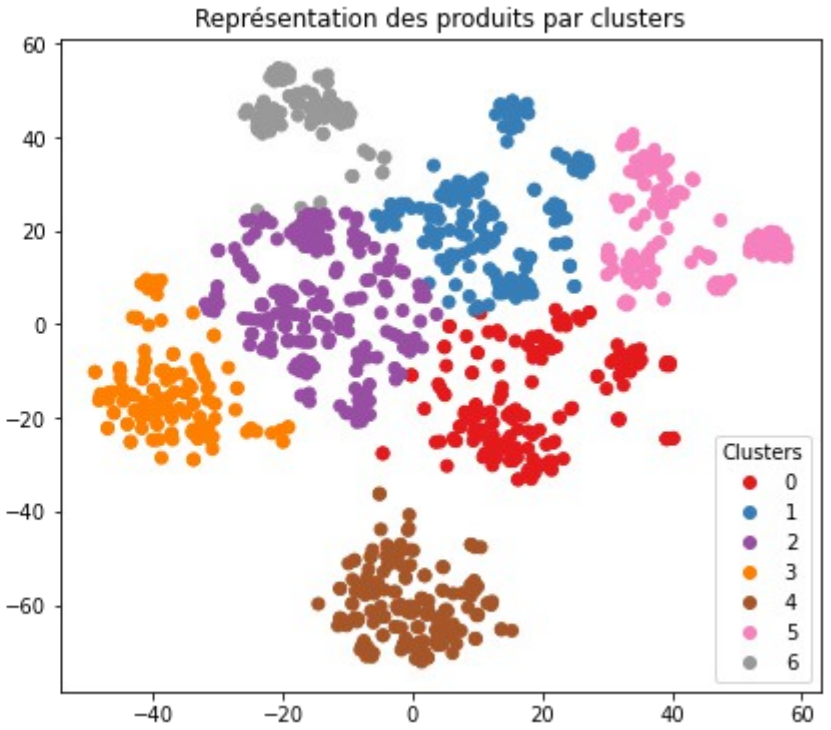
- On fusionne les noms du produit et sa description pour maximiser les informations qui seront obtenues lors de l'extraction des features texte.
- Les catégories considérées pour chaque produit seront celles situés à la racine de l'arbre des catégories. Cela donne 7 catégories possibles pour chaque produit.
- Le texte sera préparé avec un tokenizer, un retrait de ponctuation, une mise en minuscule, et un lemmatizer, pour pouvoir être utilisé par nos modèles.
- Les catégories seront extraites à l'aide d'un Kmeans et représentées à l'aide d'un Tsne.

Bag of word - Tf-idf

- On teste l'extraction de features avec Bag of word et Tf-idf.
- Les deux algorithmes construisent un vocabulaire par rapport aux mots utilisés dans l'ensemble des phrases.
- Bag of word transforme une phrase en un vecteur comptant le nombre d’apparition de chaque mot par rapport au vocabulaire construit.
- Tf-idf calcule l'importance de chaque mot dans la phrase par rapport à l'ensemble des phrases.
- Au final, le Tf-idf donne le meilleur ARI.

Bag of word - Tf-idf



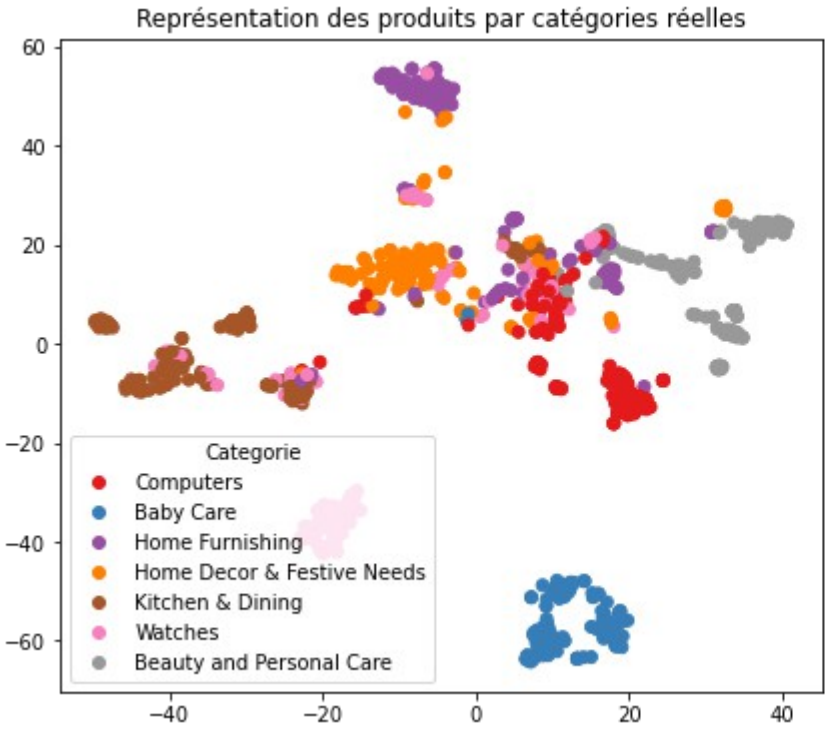


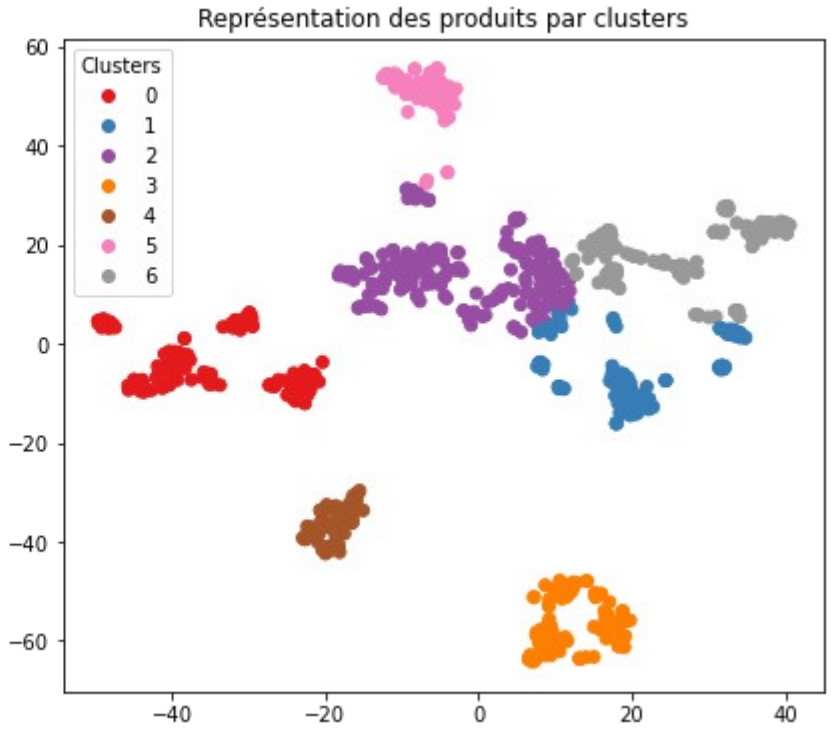
ARI : 0.5041

Word2Vec

- Le Word2Vec transforme chaque mot en un vecteur.
- Plus les mots ont un sens proche, plus leurs vecteurs sont proches.
- On peut faire des opérations entre ces vecteurs pour extraire le sens d'une combinaison de mot.
- Il prend également en compte la position des mots dans chaque phrase par rapport aux autres pour la création du vecteur.
- Les features extraites de chaque phrase correspondent à la moyenne des vecteurs de la phrase.

Word2Vec



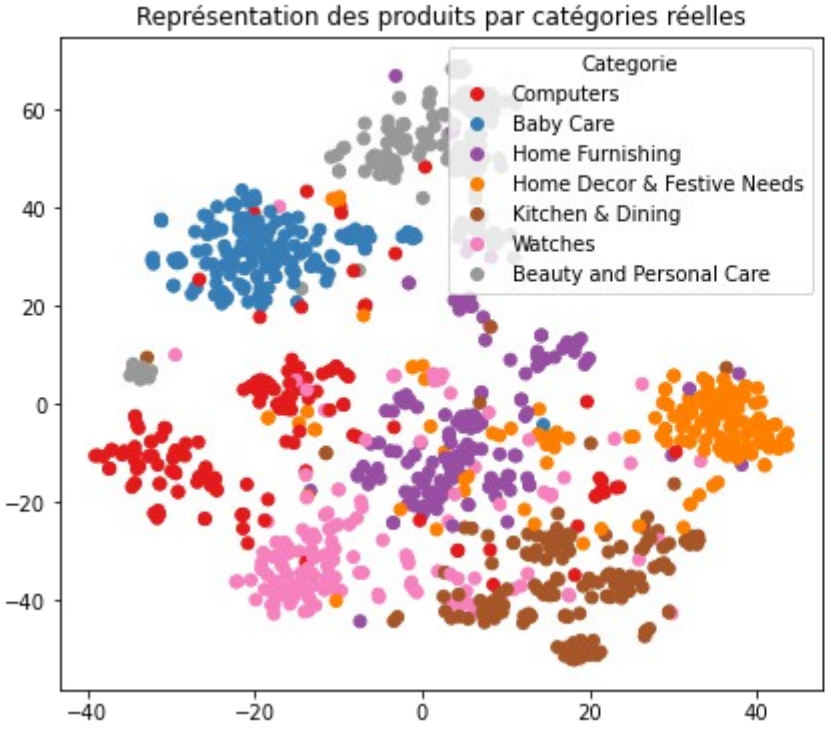


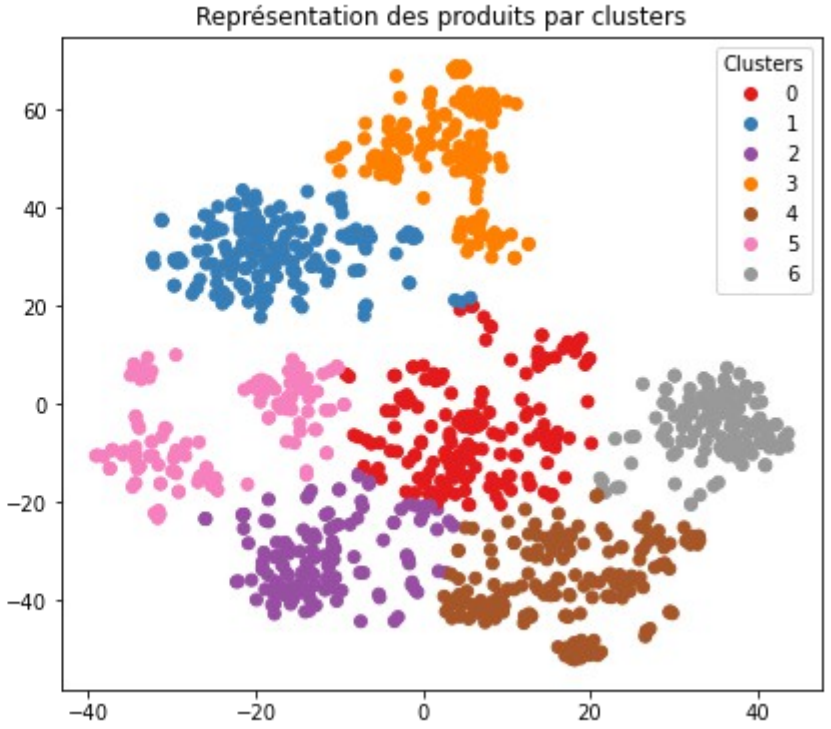
ARI : 0.5071

BERT HuggingFace

- BERT est une abréviation de : BERT : Bidirectional Encoder Representations from Transformers
- Comme Word2Vec, il transforme les mots en vecteur.
- A la différence de Word2Vec, BERT a déjà été pré-entraîné.

BERT HuggingFace



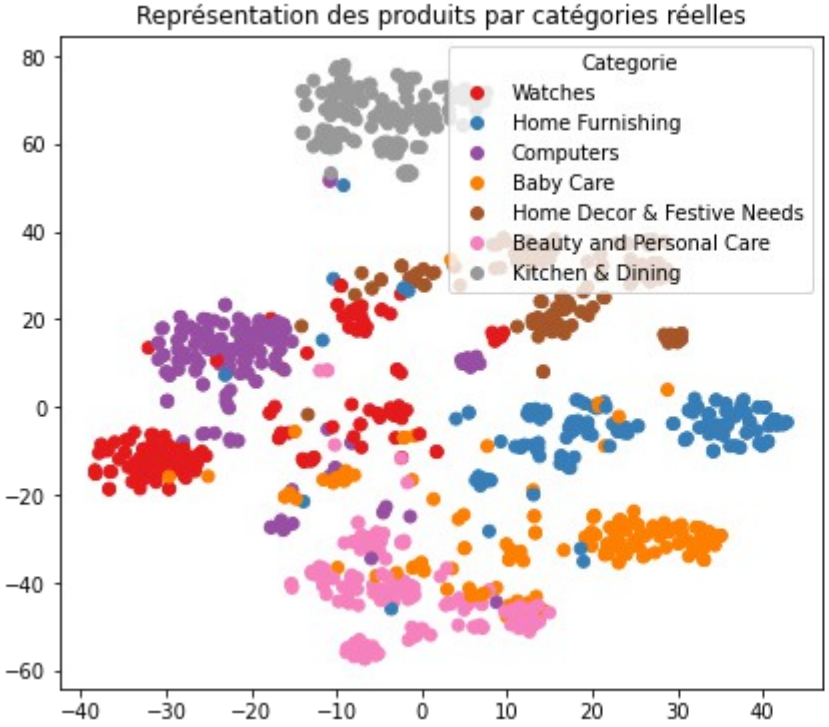


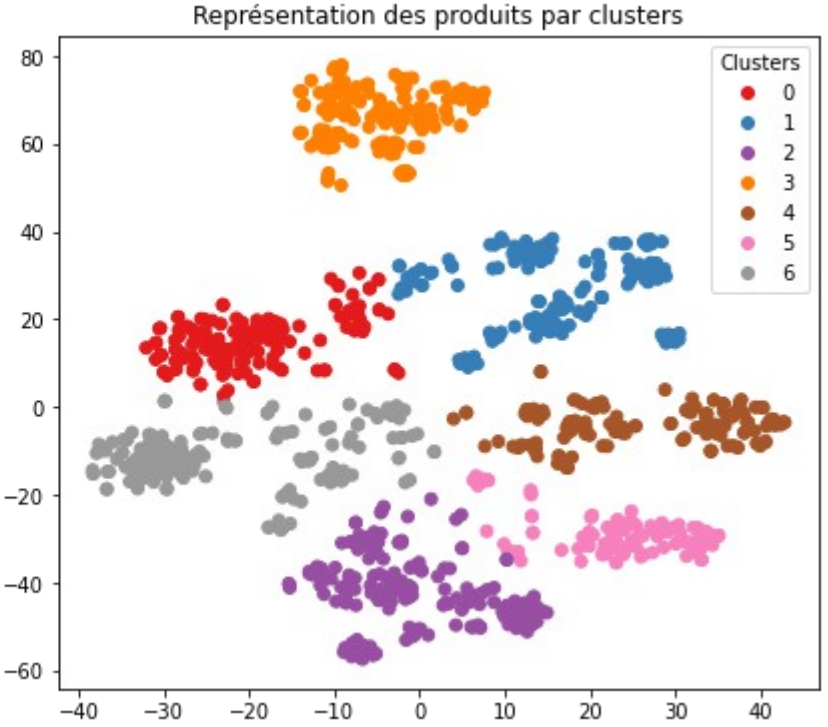
ARI : 0.6283

USE - Universal Sentence Encoder

- USE est un modèle développé par google et ouvert au grand public.
- Transforme les phrases en vecteur.
- Ce modèle est une combinaison de plusieurs modèles entraînés sur un grand nombre de phrases différentes.
- Il est adapté à tout type de tâche demandant à un programme d'interpréter du texte.

USE - Universal Sentence Encoder





ARI : 0.662

Préparation des images

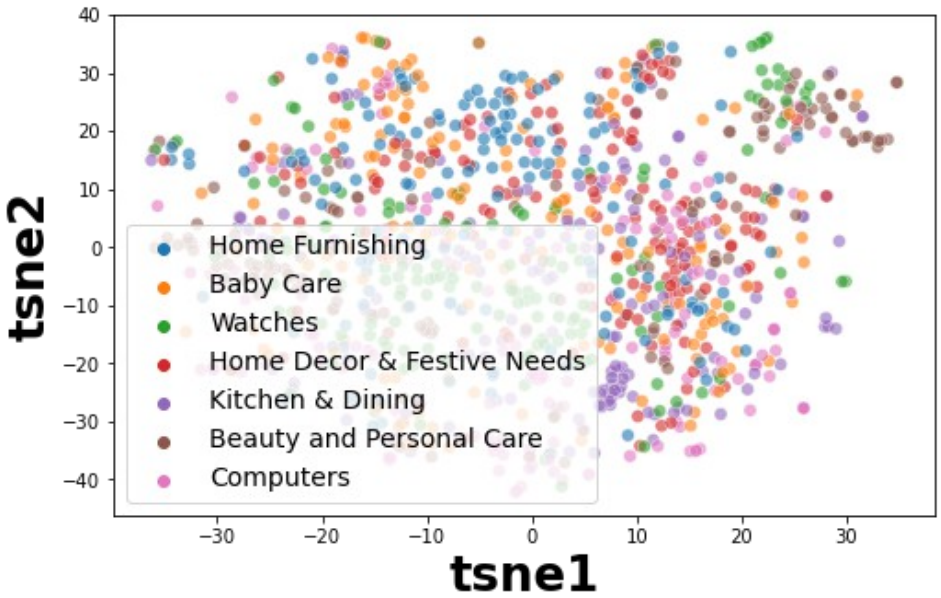
- On récupère les images situées dans le dossier annexe sous la forme d'un numpy array.
- Selon le modèle utilisé, on récupère soit une version noir et blanc des images (SIFT), soit toutes les images dans le même format de 224 sur 224 pixels (CNN).

SIFT

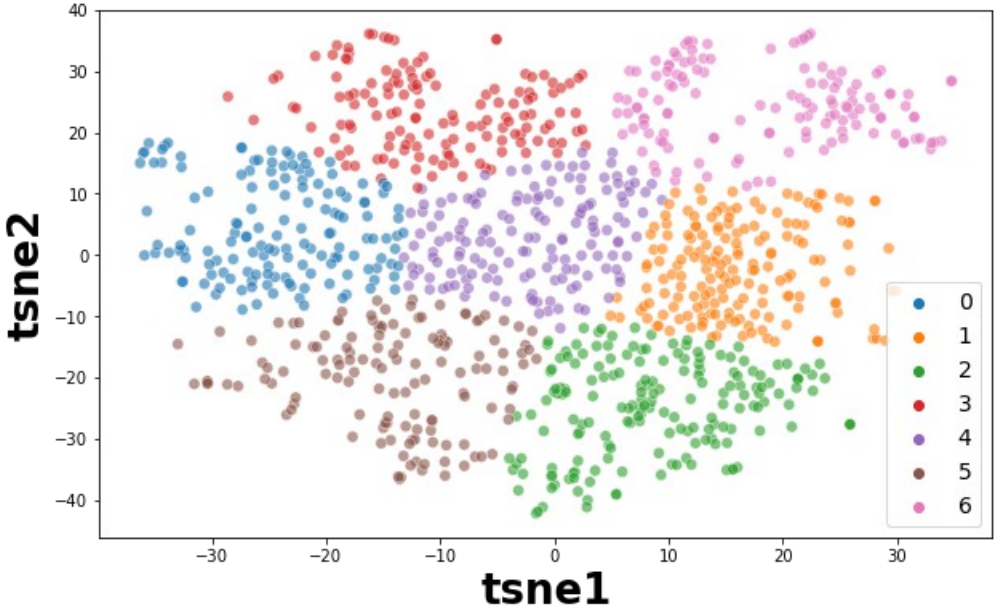
- SIFT : Scale invariant feature transform
- SIFT est utilisé pour réduire une image en un ensemble de points et leur vecteur descriptors.
- On va faire des clusters de ces descriptors.
- Nos features correspondra pour une image au nombre de descripteurs par cluster.

SIFT

TSNE selon les vraies classes



TSNE selon les clusters



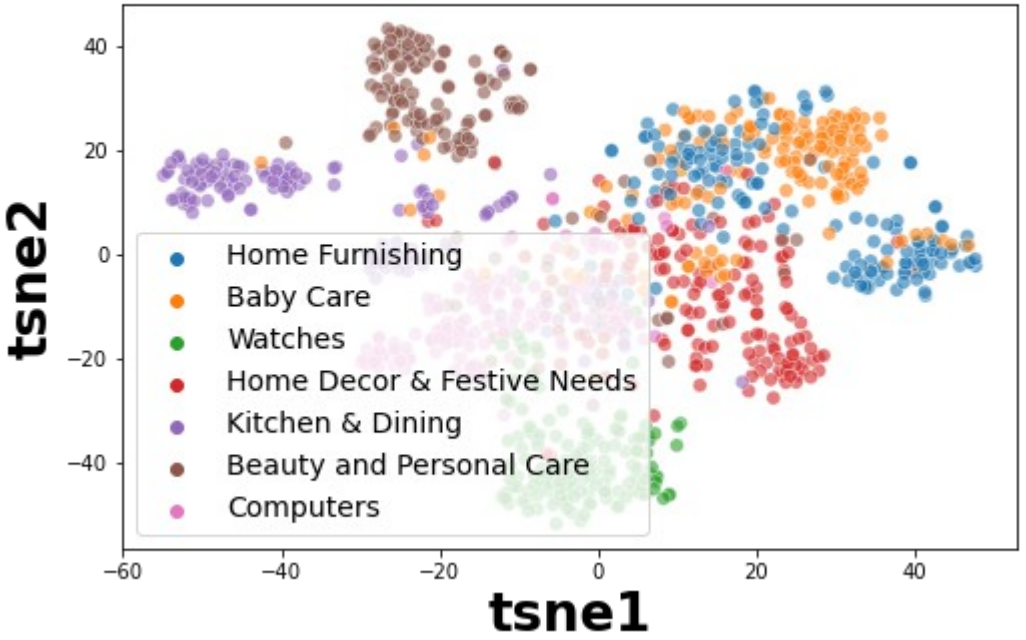
ARI : 0.05

CNN – Transfert learning

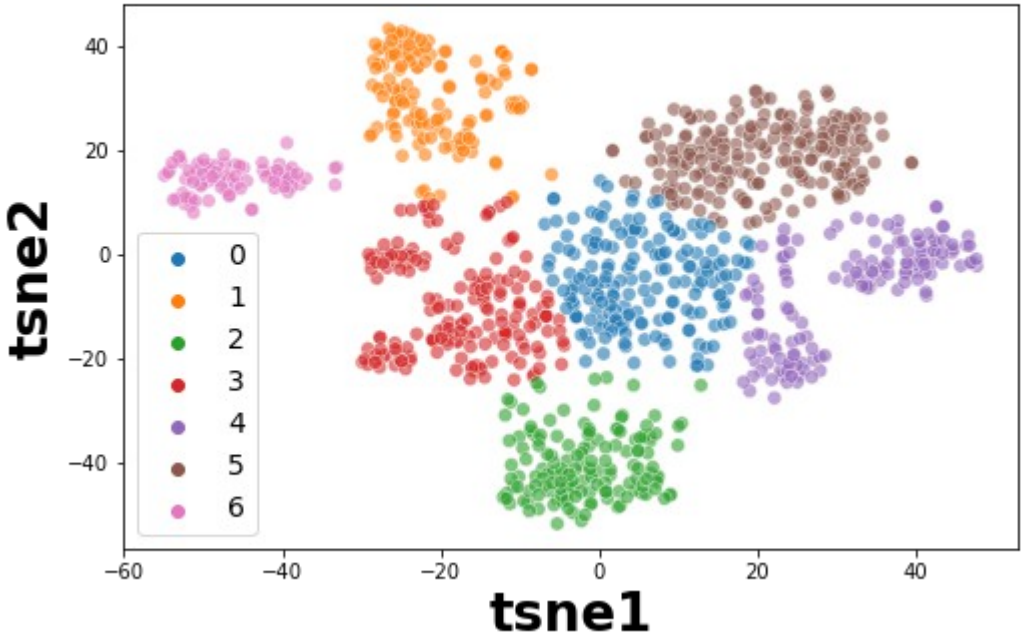
- On importe le modèle pré-entrainé VGG16.
- On modifie le modèle pour récupérer uniquement les features.

CNN – Transfert learning

TSNE selon les vraies classes



TSNE selon les clusters

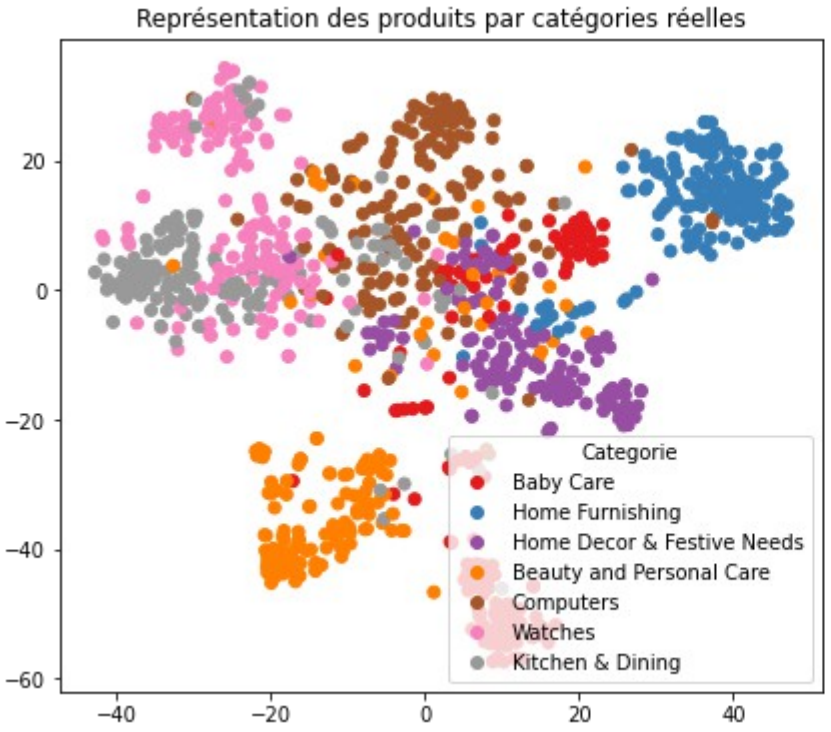


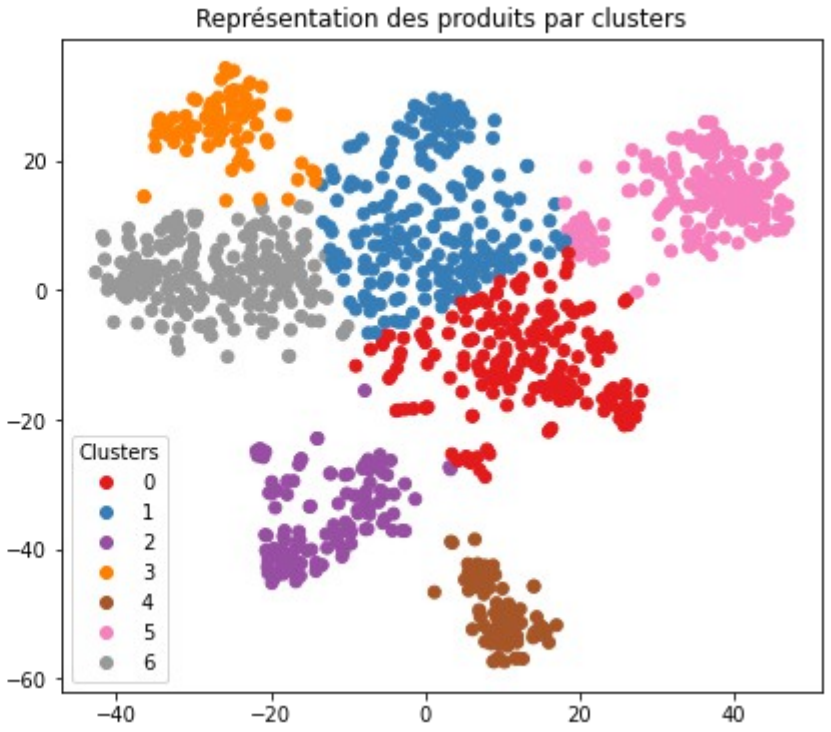
ARI : 0.44

Fusion des features

- On a étudié l'extraction des features du texte et des images individuellement
- Peut-on obtenir de meilleur cluster en combinant les features extraites du texte et les features extraites des images ?

Fusion des features





ARI : 0.4874

Conclusion

- On peut prédire avec un bon ARI la catégorie d'un article à partir de son nom et de sa description via USE.
- Il est possible d'automatiser l'attribution de catégories à un article.
- Il faudra néanmoins proposer au client la possibilité de rectifier.