Classifiez Automatiquement des Biens de Consommation

Synthèse

Contexte

PROJET

étudier la possibilité d'affecter automatiquement des catégories à des biens de consommation vendus sur une marketplace e-commerce en utilisant

- leurs descriptions textuelles (en anglais)
- leurs photos

➢ OBJECTIF : automatiser la mise en place de la classification de ces articles, pour remplacer l'étiquetage manuel actuel (chronophage & sujet à erreurs)

DÉMARCHE

- étude de faisabilité analysant, prétraitant et extrayant les caractéristiques des données texte et image
- évaluation la capacité à regrouper automatiquement les produits par catégorie grâce à des techniques de réduction de dimension, de clustering & de mesure de similarité, et visualisation des résultats préliminaires
- évaluation des résultats de la classification supervisée

Sommaire

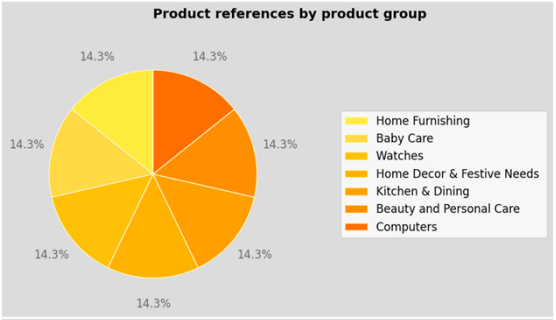
1 – Évaluation de Faisabilité

1.1 – Analyse Exploratoire des Données

- Données csv – 1 fichier 1,050 lignes x 15 colonnes
  - 2.17% de NaNs
  - Pas de doublons
  - Colonnes utiles : 'product_name', 'product_category_tree', 'image', 'description'
  - Répartition égale entre les 7 catégories

- Données images – 1 dossier
  - 1,050 photos .jpg listées dans la colonne 'image' du .csv

Exemples de photos d'articles dans la catégorie "Home Decor & Festive Needs"



Wordcloud avant retraitements, toutes categories incluses

1 – Évaluation de Faisabilité

1.2.1 – Étude de Faisabilité - Retraitements & Pré-Classification du Texte

- Étude de faisabilité => 3 questions préalables à la classification supervisée:
  - Peut-on extraire du texte des features caractéristiques des catégories de produits?
  - Ces features sont-elles suffisamment discriminantes pour assurer la séparabilité des catégories en classes distinctes?
  - Peut-on quantifier la similarité entre les classes de features extraites et les catégories réelles de produits?

- Démarche:

Pré-processing	Extraction des features	Réduction de dimension	Clustering & similarité
Tokenization Stop words Lemmatization Stemming Custom stop words	Bag-of-Words: Count Vectorizer Tf-idf Word Embedding: Word2Vec BERT USE	t-SNE	K-Means (7 clusters) Analyse graphique ARI

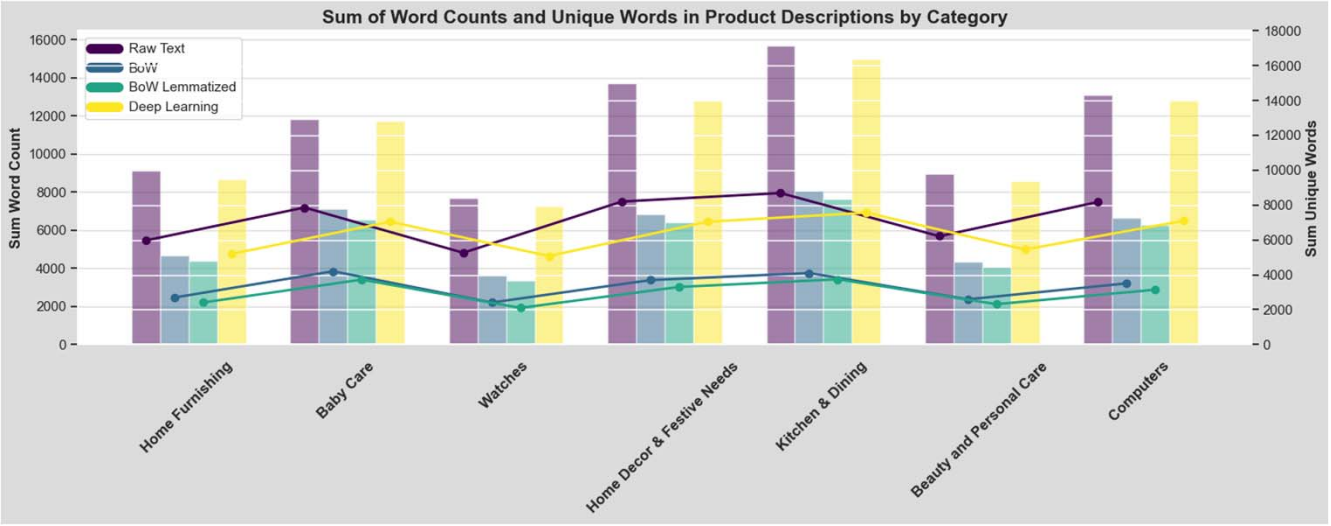
1 – Évaluation de Faisabilité

1.2.1 – Étude de Faisabilité - Retraitements & Pré-Classification du Texte

❖ Exemples de retraitements:

Corpus initial	Corpus Deep Learning	Corpus Bag-of-Words
Buy Exotic India Blessing Buddha Showpiece - 36.83 cm for Rs.15000 online. Exotic India Blessing Buddha Showpiece - 36.83 cm at best prices with FREE shipping & cash on delivery. Only Genuine Products. 30 Day Replacement Guarantee.	buy exotic india blessing buddha showpiece cm for rs online exotic india blessing buddha showpiece cm at best prices with free shipping cash on delivery only genuine products day replacement guarantee	exotic india blessing buddha showpiece online exotic india blessing buddha showpiece
37 mots	31 mots	11 mots
29 mots uniques	25 mots uniques	6 mots uniques

*texte de la ligne 97 (image f4d4c2eec77732f56e47722d7a355f2b.jpg), groupe 3 – Home Decor & Festive Needs

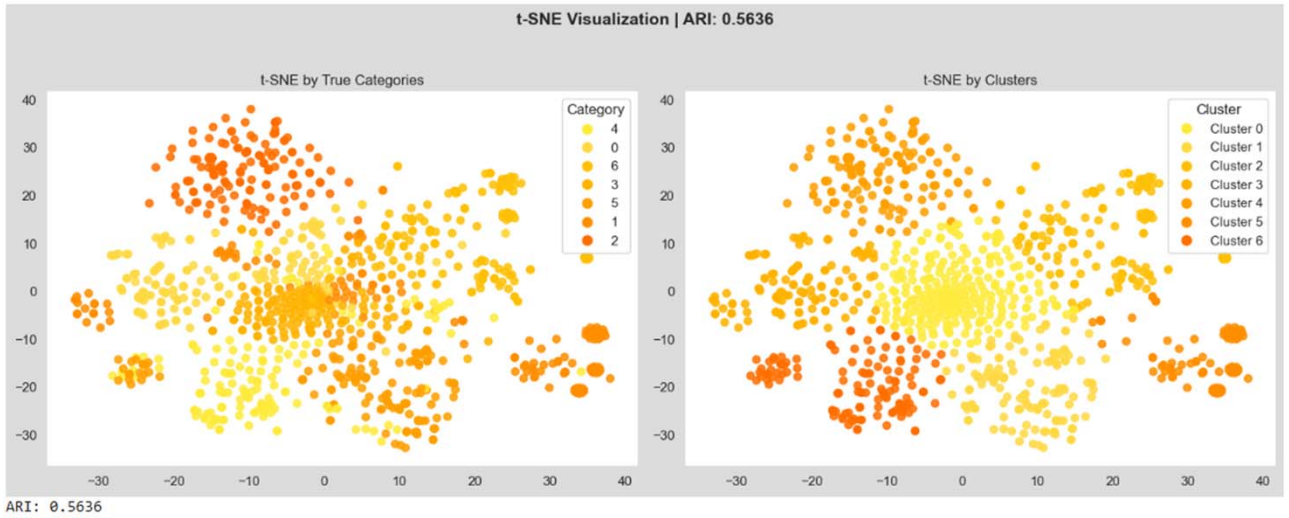


1 – Évaluation de Faisabilité

1.2.1 – Étude de Faisabilité - Retraitements & Pré-Classification du Texte

❖ Résultats:

Model	ARI	Fitting Time (s)
Count Vectorizer	0.4167	16.09
Tf-idf	0.5636	7.46
Word2Vec	0.5317	9.82
BERT	0.3388	7.4
USE	0.4481	22.27



❖ Le t-SNE fait apparaître des groupes relativement distincts

❖ Score ARI > 0.5 avec classification non supervisée (clustering K-Means avec k = 7)

➢ Faisabilité de la classification supervisée du texte confirmée

1 – Évaluation de Faisabilité

1.2.2 – Étude de Faisabilité - Retraitements & Pré-Classification des Images

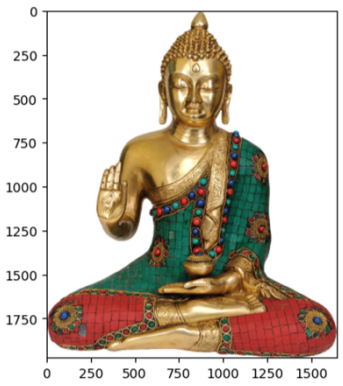
- Étude de faisabilité => 3 questions préalables à la classification supervisée:
  - Peut-on extraire des images des features caractéristiques des catégories de produits?
  - Ces features sont-elles suffisamment discriminantes pour assurer la séparabilité des catégories en classes distinctes?
  - Peut-on quantifier la similarité entre les classes de features extraites et les catégories réelles de produits?

- Démarche:

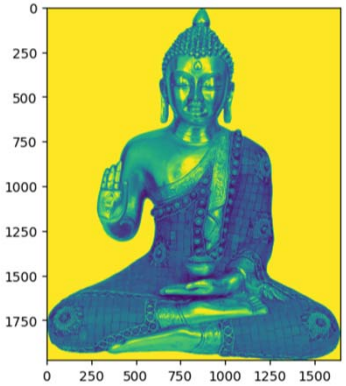
Pré-processing	Extraction des features	Réduction de dimension	Clustering & similarité
Niveaux de gris	SIFT	Mini-batch K-Means	K-Means (7 clusters)
Équalisation des histogrammes	SIFT on downsampled images	PCA	Analyse graphique
Ajout de bruit	ORB	t-SNE	ARI
Filtrage	ORB on image pyramids		
	VGG16		
	VGG19		
	ResNet50		
	Xception		
	InceptionV3		

1 – Évaluation de Faisabilité

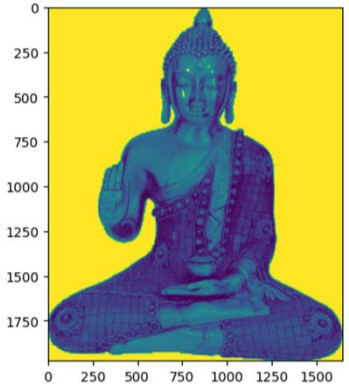
1.2.2 – Étude de Faisabilité - Retraitements & Pré-Classification des Images



❖ Image originelle



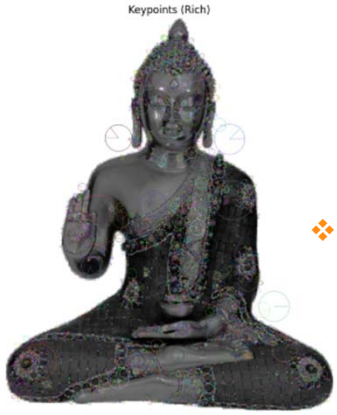
❖ Niveaux de gris



❖ Équalisation des histogrammes

❖ Descripteurs simples

Descriptors shape: (14926, 128)
Processing time: 1.37 secs



❖ Descripteurs riches

* Exemple ligne 97 – photo f4d4c2eec77732f56e47722d7a355f2b.jpg

1 – Évaluation de Faisabilité

1.2.2 – Étude de Faisabilité - Retraitements & Pré-Classification des Images

❖ Résultats:

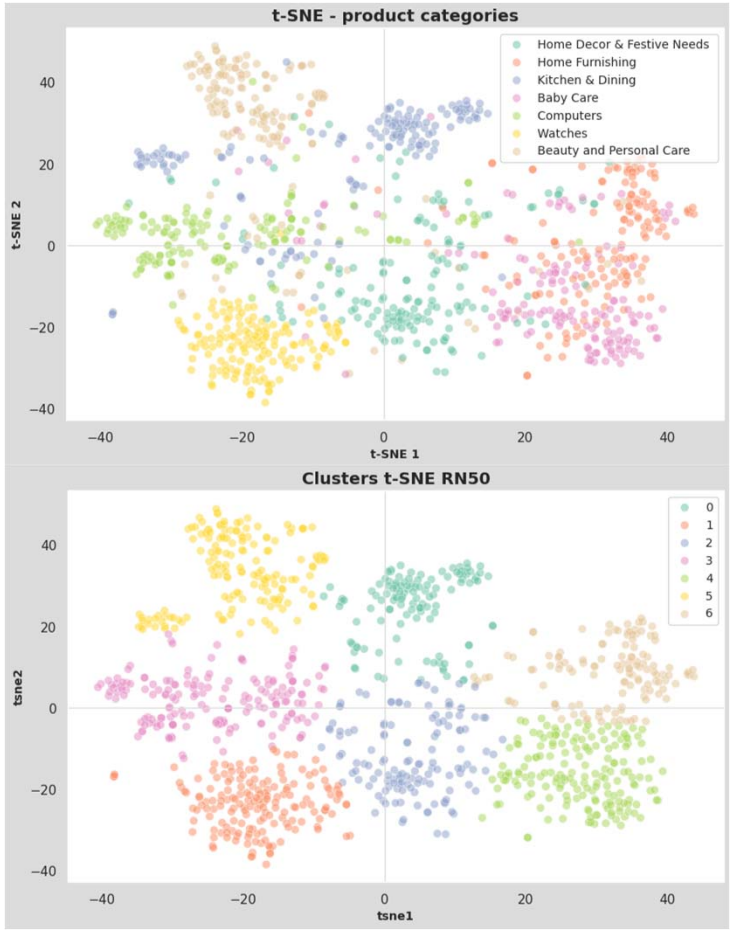
Extractor	Fit time	ARI
SIFT-BA	348.9878	0.0456
SIFT-DI	96.8513	0.0501
ORB-BA	33.2189	0.0391
ORB-IP	59.6772	0.0299
VGG-16	95.54	0.4660
VGG-19	104.8187	0.5215
ResNet50	98.0764	0.5554
Xception	100.3474	0.5431
InceptionV3	112.6795	0.5431

```
Dataset dimensions before PCA reduction :  (1050, 2048)
Dataset dimensions after PCA reduction :  (1050, 722)
```

❖ Le t-SNE fait apparaître des groupes relativement distincts

❖ Score ARI > 0,5 avec classification non supervisée
(clustering K-Means avec k = 7)

➢ Faisabilité de la classification supervisée des images confirmée



2 – Modélisation

2.1 – Méthodologie

Extraction des features → Modélisation des images seules → Modélisation images & texte → Evaluation finale

- **Tf-idf pour le texte**
  - Data augmentation avec synonymes, random insertions, swaps & deletions
- **ResNet50 pour les images**
  - Data augmentation avec flip, rotation, cropping, color jitter, gaussian blur & coarse dropout

- **Modèles Utilisés:**
  - ResNet50
  - ResNet50 Augmented
- **Contrôle de l'Overfitting:**
  - Epochs/early stopping
  - Feature augmentation
- **Métriques d'évaluation:**
  - F1
  - fit time
- **Données Utilisées:**
  - Train
  - Train + Validation
  - (Train + Validation) + Augmentation

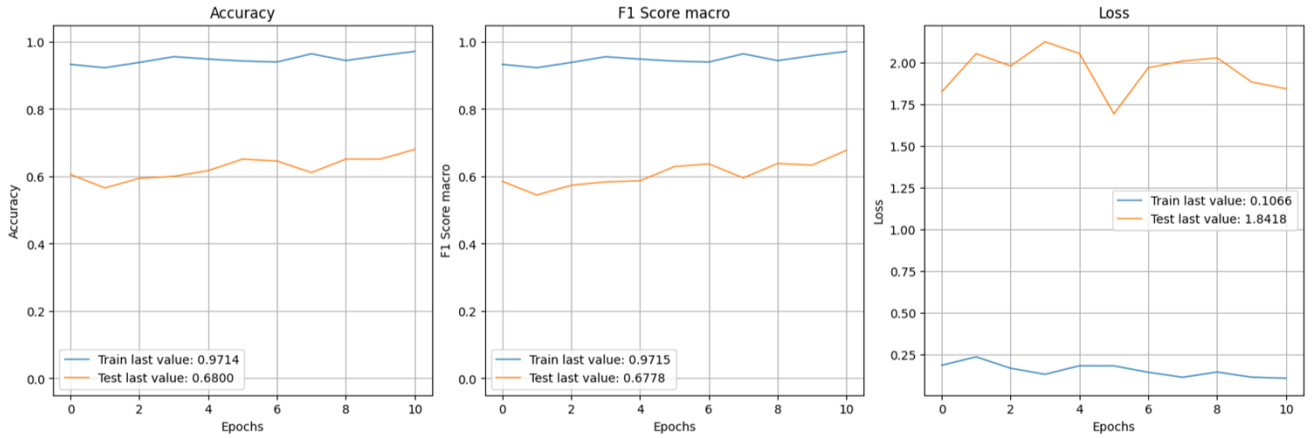
- **Modèles Utilisés:**
  - Random Forest
  - LightGBM
  - XGBoost
- **Contrôle de l'Overfitting:**
  - Cross-validation
  - Stratified sampling
- **Métriques d'évaluation:**
  - F1
  - fit time
- **Données Utilisées:**
  - Train
  - Train + Augmentation
  - Train + Validation

- Sur le jeu de test
- Matrice de confusion

2 – Modélisation

2.2 – Classification Supervisée des Images Seules

model_name	epochs_run	early_stopping_epoch	fit_time	Train Accuracy (Hist)	Val Accuracy (Hist)	Train f1_macro (Hist)	Val f1_macro (Hist)	Val Accuracy (Sklearn)	Val Precision (Sklearn)	Val Recall (Sklearn)	Val f1_macro (Sklearn)	Val f2_macro (Sklearn)	Train size	Val/Test size
ResNet50 External Data Augmentation	50	11	102.64	0.9143	0.5429	0.9141	0.5002	0.5429	0.7546	0.5429	0.5002	0.5051	700	125
ResNet50 Integrated Data Augmentation	50	25	117.18	0.9186	0.6229	0.9183	0.6100	0.6229	0.7802	0.6229	0.6100	0.5990	700	125
ResNet50 Original Data	50	11	58.61	0.9071	0.5943	0.9069	0.5749	0.5943	0.7227	0.5943	0.5749	0.5703	700	125

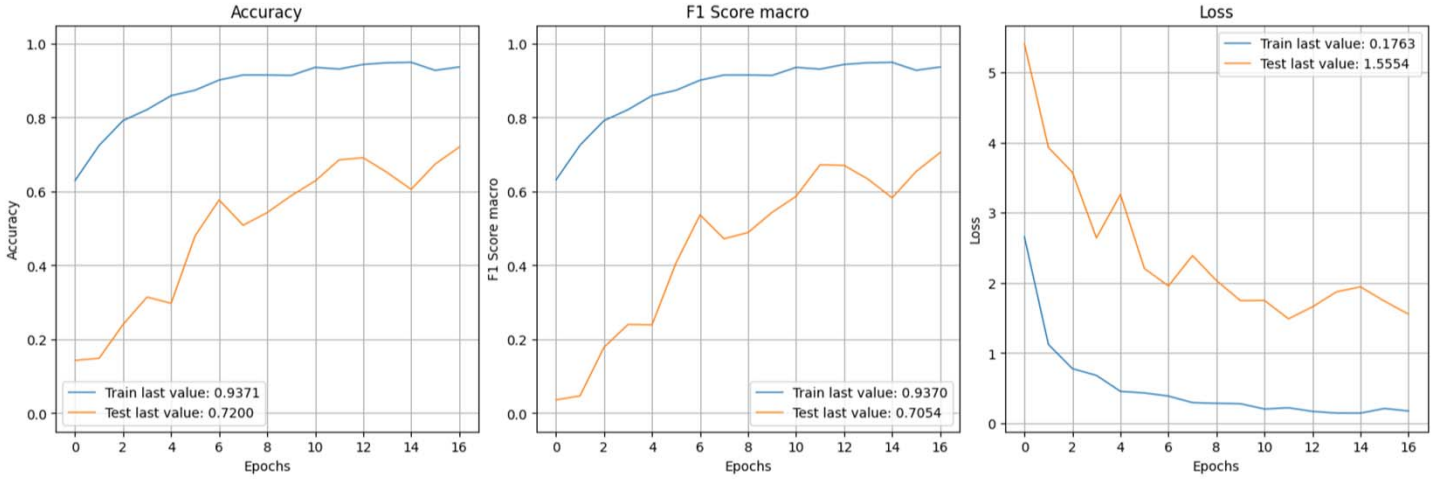


- Écart de performance train / validation
  - Overfitting

2 – Modélisation

2.2 – Classification Supervisée des Images Seules – Data Augmentation

model_name	epochs_run	early_stopping_epoch	fit_time	Train Accuracy (Hist)	Val Accuracy (Hist)	Train f1_macro (Hist)	Val f1_macro (Hist)	Val Accuracy (Sklearn)	Val Precision (Sklearn)	Val Recall (Sklearn)	Val f1_macro (Sklearn)	Val f2_macro (Sklearn)	Train size	Val/Test size
ResNet50 External Data Augmentation	50	11	102.6436	0.9143	0.5429	0.9141	0.5002	0.5429	0.7546	0.5429	0.5002	0.5051	700	175
ResNet50 Integrated Data Augmentation	50	25	117.1815	0.9186	0.6229	0.9183	0.6100	0.6229	0.7802	0.6229	0.6100	0.5990	700	175
ResNet50 Original Data	50	11	58.6104	0.9071	0.5943	0.9069	0.5749	0.5943	0.7227	0.5943	0.5749	0.5703	700	175
ResNet50 Original Data Train + Val	50	17	101.3423	0.9371	0.7200	0.9370	0.7054	0.7200	0.8141	0.7200	0.7054	0.7030	875	175
ResNet50 Augmented Data Train + Val	50	11	55.4925	0.9280	0.6400	0.9281	0.6274	0.6400	0.8066	0.6400	0.6274	0.6173	875	175
ResNet50 Original + Augmented Data Train & Val	50	11	65.0602	0.9571	0.6171	0.9571	0.5847	0.6171	0.8000	0.6171	0.5847	0.5874	1750	175



- Performance ameliorée mais overfitting persiste

2 – Modélisation

2.3 – Classification Supervisée des Images & du Texte

❖ Sans augmentation :

Model	Training set	Train f1_macro	Train fit time	Val f1_macro	Val f2_macro	Val Accuracy	Val Precision	Val Recall
RandomForestClassifier	X_train_no_aug	0.8020	48.73	0.8383	0.8372	0.8400	0.8566	0.8400
LGBMClassifier	X_train_no_aug	0.2039	74.98	0.2454	0.2481	0.2514	0.2474	0.2514
XGBClassifier	X_train_no_aug	0.7382	1174.72	0.8062	0.8017	0.8057	0.8410	0.8057

❖ Avec augmentation :

Model	Training set	Train f1_mac	Train fit time	Val f1_macr	Val f2_macr	Val Accurac	Val Precisio	Val Recall
RandomForestClassifier	X_train_aug	0.7792	50.68	0.8184	0.8163	0.8171	0.8302	0.8171
LGBMClassifier	X_train_aug	0.1736	73.54	0.2348	0.2435	0.2514	0.2282	0.2514
XGBClassifier	X_train_aug	0.7261	1301.93	0.7469	0.7456	0.7486	0.7658	0.7486

Paramètres :

➢ {'model__max_depth': 10,
'model__max_features': 'sqrt',
'model__min_samples_split': 2,
'model__n_estimators': 800}

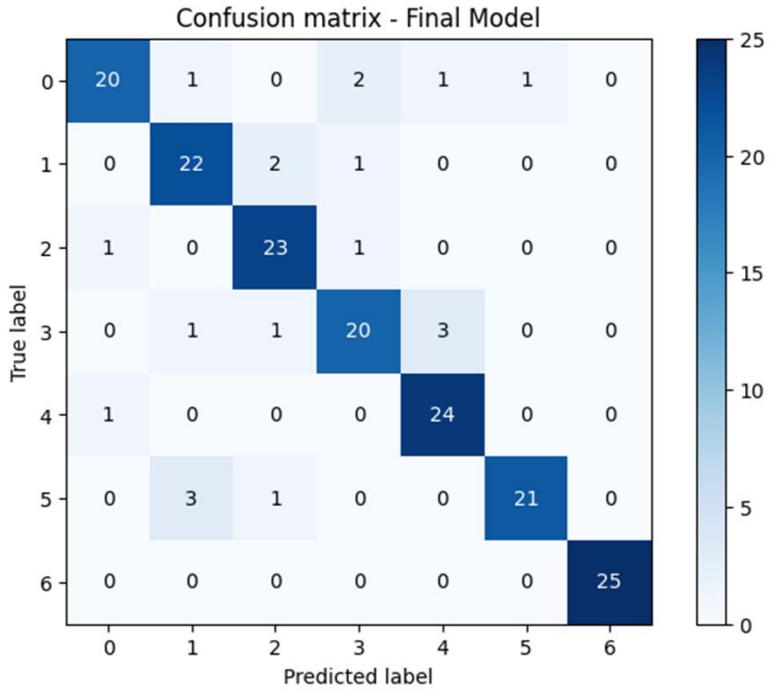
➢ {'model__max_depth': 10,
'model__max_features': 'sqrt',
'model__min_samples_split': 2,
'model__n_estimators': 800}

}

2 – Modélisation

2.3 – Classification Supervisée des Images & du Texte

- Réentrainement du meilleur modèle*
  - Sur les données originelles (sans augmentation)
  - Sur la totalité des jeux train + validation
  - Évaluation des performances sur le jeu de test
- Score F1 = 0.8853
- Classe 6 (watches) parfaitement prédite, suivie des classes 4 (Home Furnishing) et 2 (Computers)



*RandomForestClassifier(max_depth=10, max_features= sqrt, min_samples_split=2 et n_estimators = 800)

3 – Collecte de Données via API

3.1 – Considérations RGPD

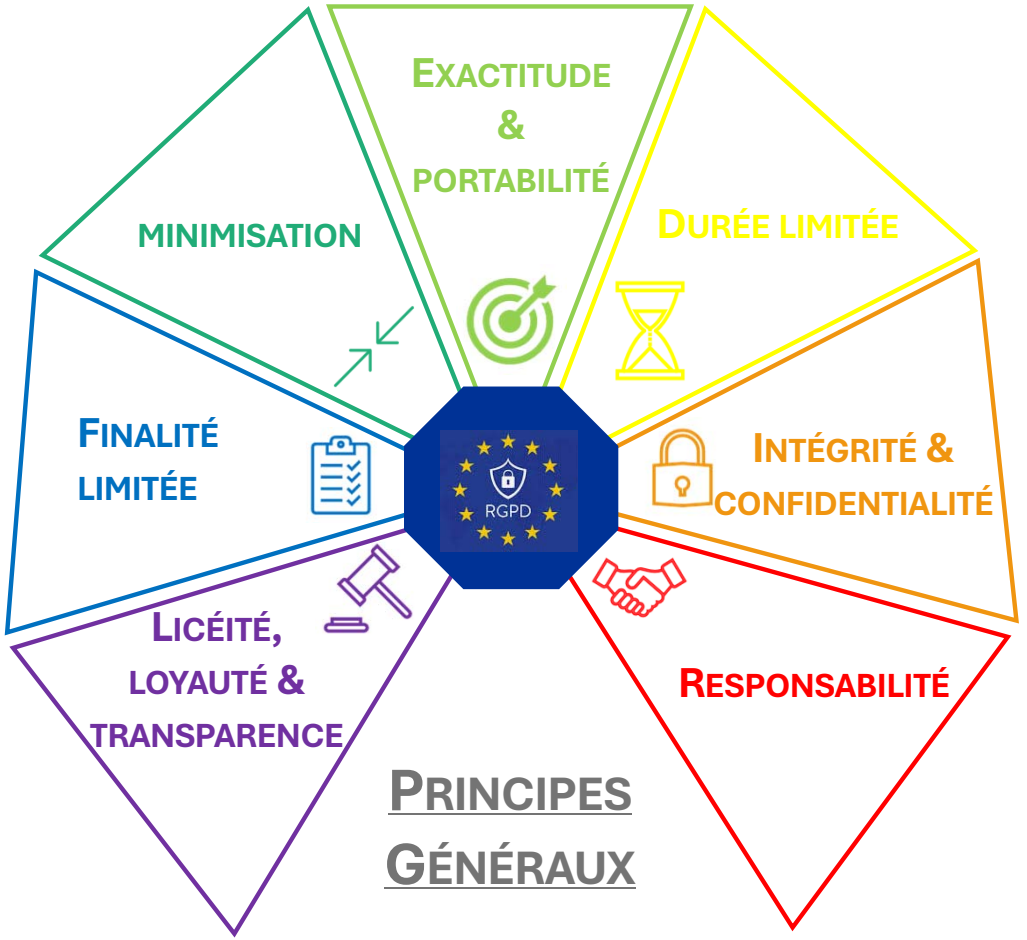
Champ d'application :

Données personnelles

Citoyens européens

Tous supports

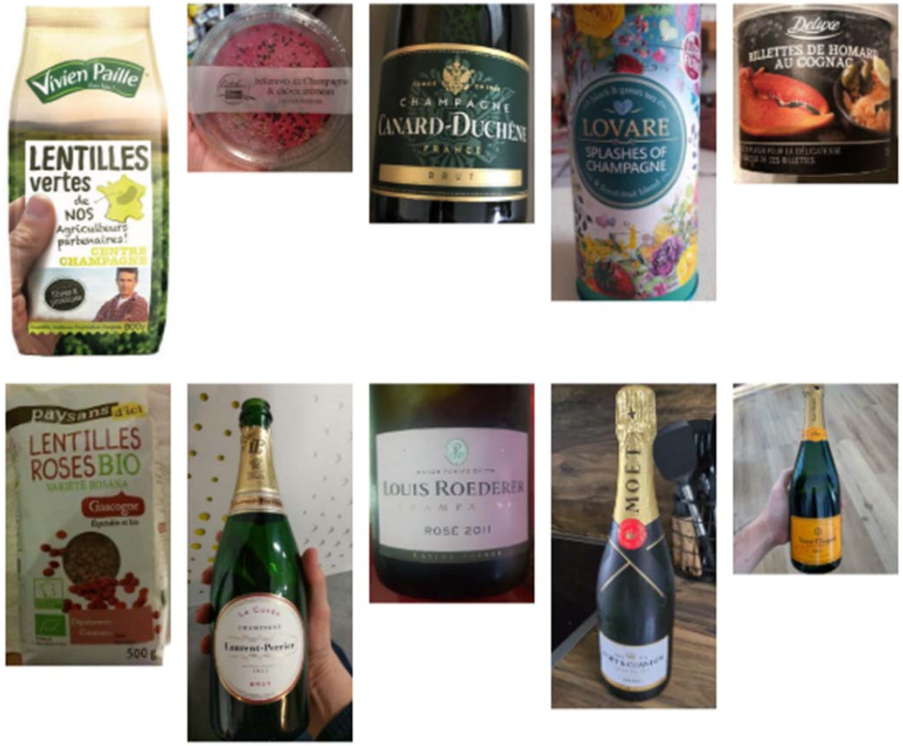
- https://lincnil.github.io/Guide-RGPD-du-developpeur/



3 – Collecte de Données via API

3.2 – Scripting & Résultats

- API OpenFoodFacts.org
- Filtrage sur le terme “champagne”
- <u>Informations collectées:</u>
  - foodId
  - label
  - category
  - foodContentsLabel
  - Image
- Voir exemple en Annexe 3



Conclusion & perspectives

- Les CNN requièrent des données volumineuses pour éliminer le risque de surapprentissage
  - early stopping/epochs & augmentation des features/images => effet limité
  - augmentation du nombre d'images => effet plus notable

- La classification sur les features images + texte peut être une meilleure option dans ce contexte

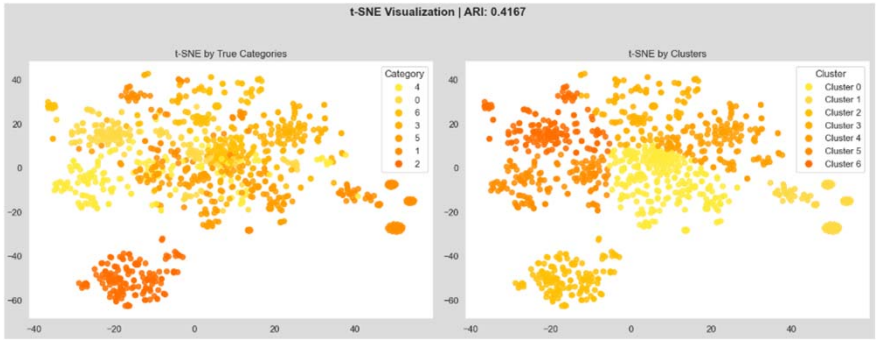
- Critère métier manquant => connaître le % d'erreur de l'étiquetage manuel actuel permettrait de juger de l'intérêt de la démarche de modélisation

- Les résultats présentés ne sont que des exemples => pas de « meilleure modélisation » absolue & autres modèles, paramètres, hypothèses & groupes de données à tester

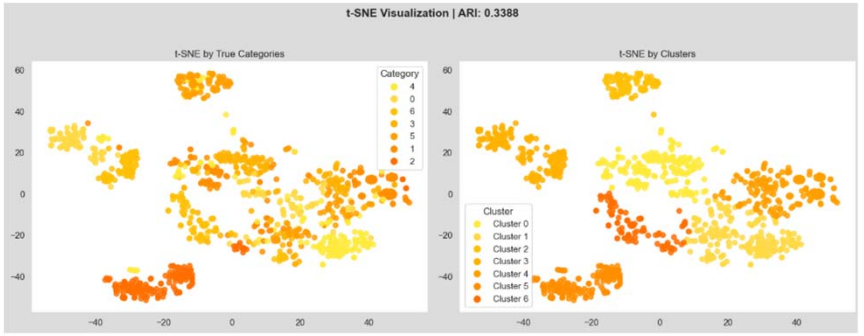
ANNEXES

ANNEXE 1 - ÉTUDE DE FAISABILITÉ TEXTE – AUTRES RÉSULTATS

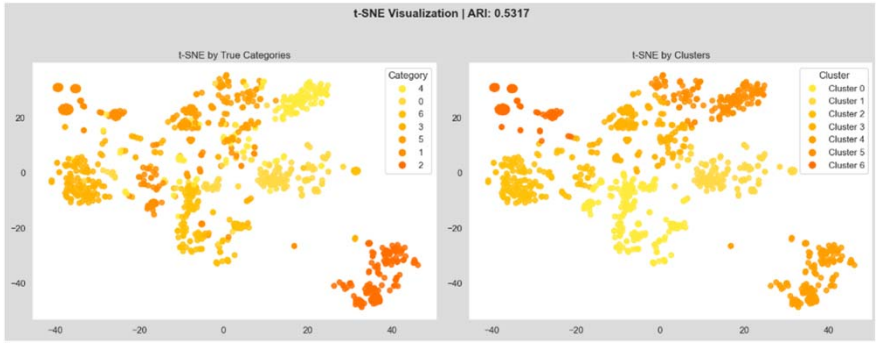
Count Vectorizer



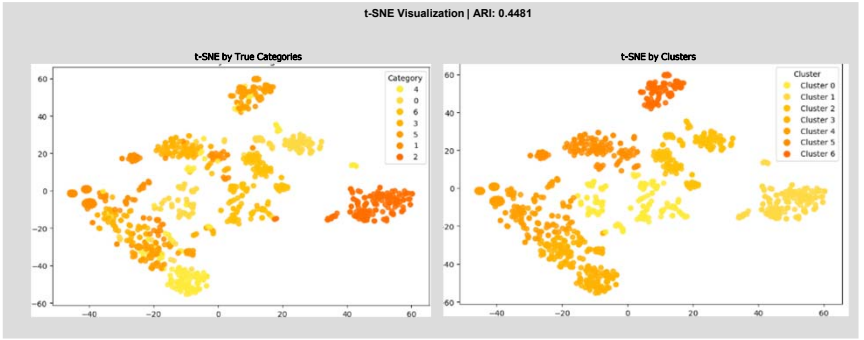
BERT with Tensorflow



Word2Vec

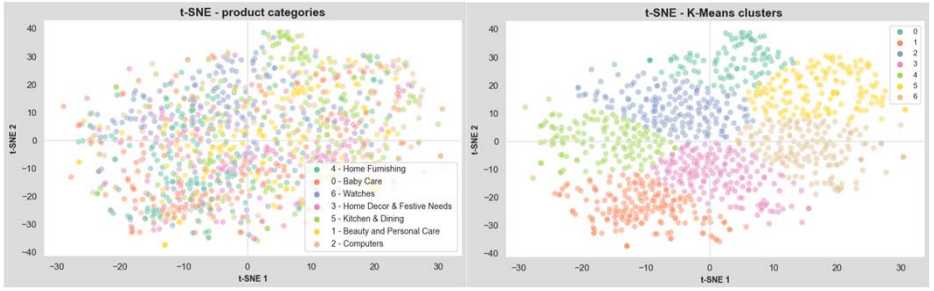


USE

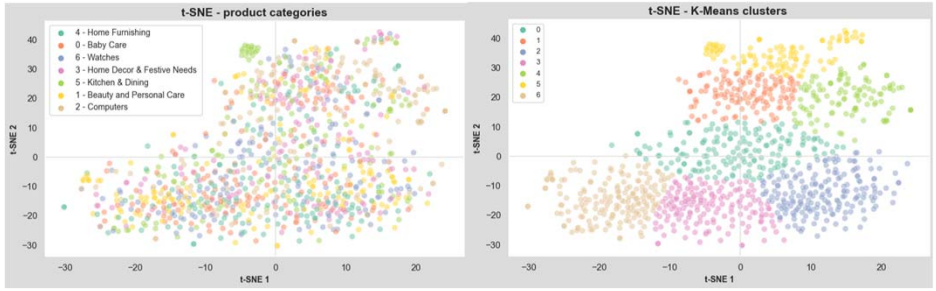


ANNEXE 2 - ÉTUDE DE FAISABILITÉ IMAGES – AUTRES RÉSULTATS (1/2)

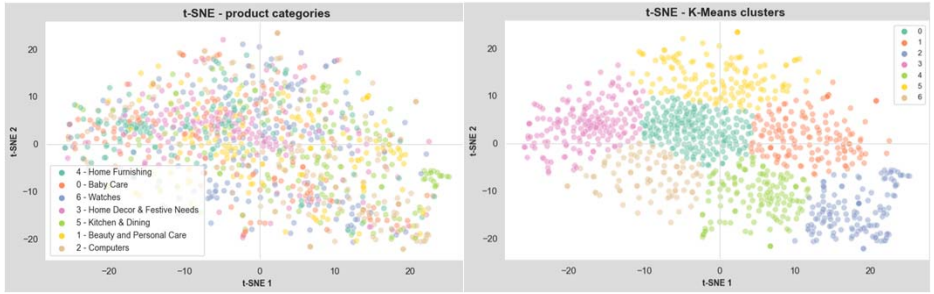
❖ SIFT (ARI = 0.0483)



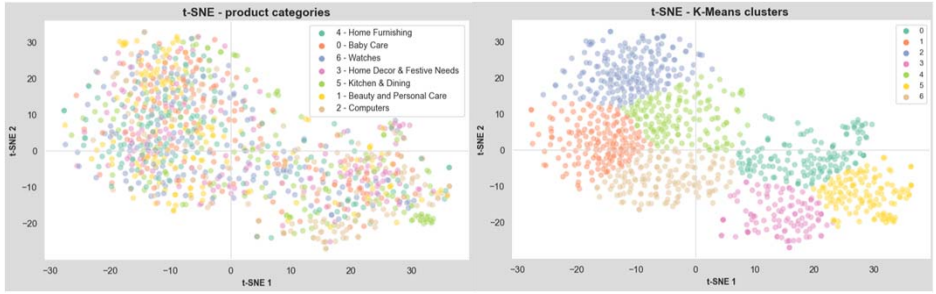
❖ ORB (ARI = 0.0346)



❖ SIFT on downsampled images (ARI = 0.0533)

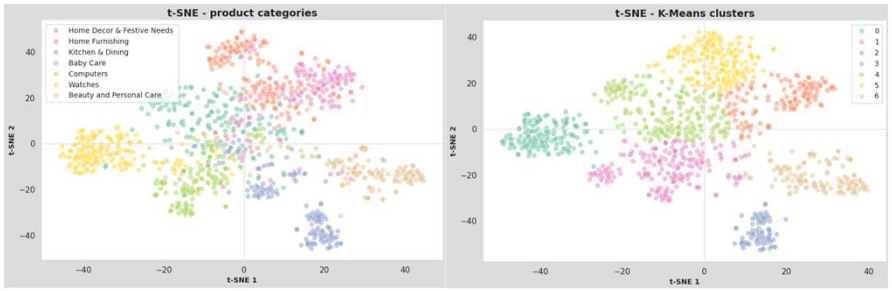


❖ ORB on image pyramids (0.0326)

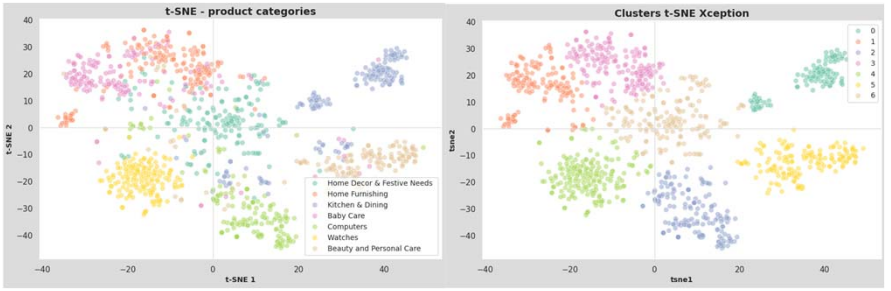


ANNEXE 2 - ÉTUDE DE FAISABILITÉ IMAGES – AUTRES RÉSULTATS (2/2)

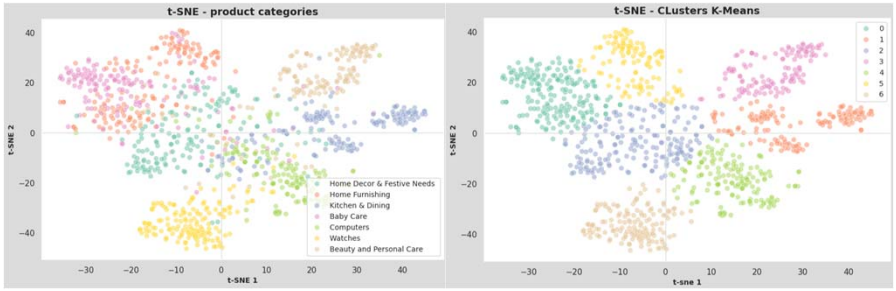
VGG16 (ARI = 0.4868)



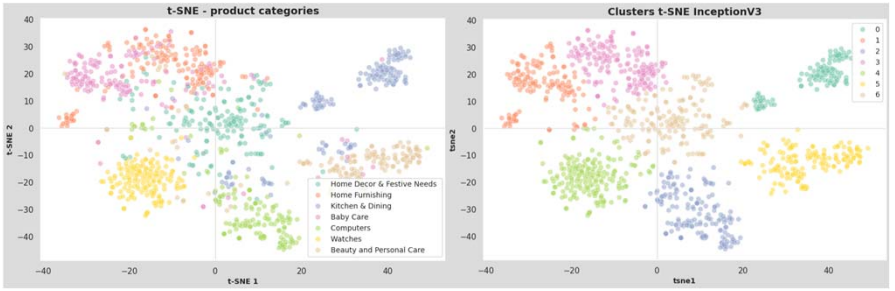
Xception (ARI = 0.5431)



VGG19 (ARI = 0.5215)



InceptionV3 (ARI = 0.5431)



ANNEXE 3 – INFORMATION PRODUITS – API OPENFOODFACTS.ORG

foodId	label	category	foodContentsLabel	image
3039820510250	Vivien Paille Lentilles vertes le paquet de 500 g	Aliments et boissons à base de végétaux	Lentilles vertes.	https://images.openfoodfacts.org/images/products/303/982/051/0250/front_fr.68.400.jpg
3292070010264	Betteraves de Champagne & chèvre crémeux, pointe de framboise	Salted spreads	Pois chiches 44% - purée de betterave rouge 37% - huile de colza - purée de framboise 3% - eau - fromage de chèvre (contient lait) 2% - concentré de jus de citron - sel - graines de sésame dorées - graines de nigelle - ail en poudre - conservateur: E202. Traces éventuelles de céréales contenant du gluten, crustacés, œufs, poissons, mollusques.	https://images.openfoodfacts.org/images/products/329/207/001/0264/front_fr.64.400.jpg
3113934004147	Canard Duchêne	Boissons	Pinots et de Chardonnay	https://images.openfoodfacts.org/images/products/311/393/400/4147/front_fr.4.400.jpg
4820097815556	Splashes of champagne	Non classé	Information non disponible	https://images.openfoodfacts.org/images/products/482/009/781/5556/front_it.3.400.jpg
4056489843696	Rillettes de homard au cognac	Seafood	Chair de homard américain 49%, huile de colza, colin d'Alaska, eau, double concentré de tomates, Champagne (contient sulfites), moutarde de Dijon (eau, graines de moutarde, vinaigre d'alcool, sel), fibre de blé, jaune d'œuf en poudre, farine de blé, sel, Cognac 0,5%, poivre blanc,	https://images.openfoodfacts.org/images/products/405/648/984/3696/front_fr.3.400.jpg
3760091726964	Lentilles roses bio	Aliments et boissons à base de végétaux	Lentilles	https://images.openfoodfacts.org/images/products/376/009/172/6964/front_fr.34.400.jpg
3258431220000	Laurent Perrier Champagne Brut	Boissons	Champagne	https://images.openfoodfacts.org/images/products/325/843/122/0000/front_en.4.400.jpg
3114080034057	Champagne rosé	Bebidas	Unknown	https://images.openfoodfacts.org/images/products/311/408/003/4057/front_es.3.400.jpg
3185370729960	Champagne Impérial Brut	Getränke und Getränkezubereitungen	Unknown	https://images.openfoodfacts.org/images/products/318/537/072/9960/front_de.3.400.jpg
3049610004104	Veuve Clicquot Champagne Ponsardin Brut	Boissons et préparations de boissons	Champagne	https://images.openfoodfacts.org/images/products/304/961/000/4104/front_fr.39.400.jpg