



# *PROJET 6 : CLASSIFIER AUTOMATIQUEMENT DES BIENS DE CONSOMMATION*

BEN ABDALLAH HELMI



# ***Plan de Soutenance***

- I. Présentation du projet
- II. Etude de faisabilité
- III. Classification supervisée
- IV. Collecte API
- V. Conclusion

# I. Présentation du projet



## ***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.



# Data Wrangling



*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

**7 catégories**



	product_category_tree	n	f
0	WristWatches	149	14.190476
1	LaptopAccessories	87	8.285714
2	InfantWear	84	8.000000
3	CoffeeMugs	74	7.047619
4	Showpieces	71	6.761905
...	...	...	...
57	Consumables&Disposables	1	0.095238
58	TRUEHomeDecor&FestiveNeeds	1	0.095238
59	JMDHomeFurnishing	1	0.095238
60	Kripa'sHomeFurnishing	1	0.095238
61	BeautyAccessories	1	0.095238

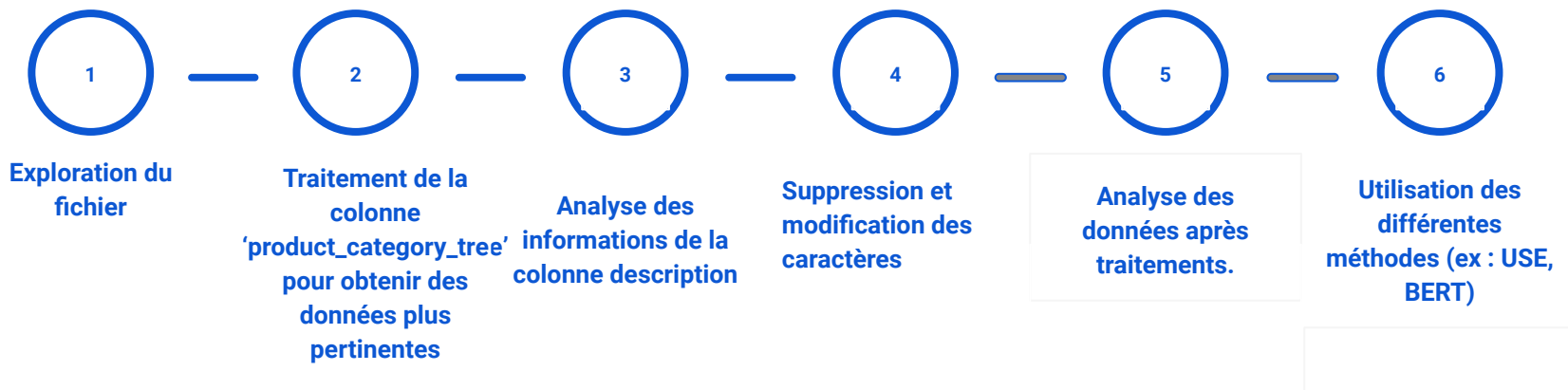
62 rows × 3 columns

	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

## II. Etude de faisabilité



# Étape du traitement de textes

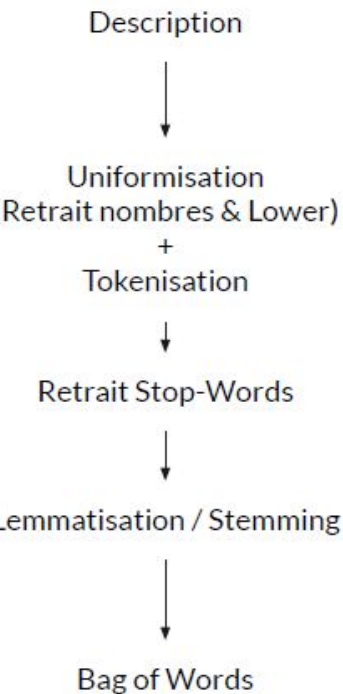




# Feature engineering



## A. Texte (Nettoyage)



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

```
'Buy Goldencollections GC4353 Makeup and Jewellery Vanity Pouch for Rs.783 online. Goldencollections GC4353 Makeup and Jewellery 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 Pouch 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']
```





# Feature engineering

## 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', 'garante']

Transformer texte -> valeurs numériques

$Tf-Idf = Tf \times Idf$

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)

Tf-Idf

exemple du bag of words stemmed :

nombre features :  
495 (lemmatisé)  
500 (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





# Définition des méthodes

Le modèle **CountVectorizer** permet de transformer un texte en un vecteur de fréquences de mots. Il crée un vocabulaire à partir des mots présents dans le texte et compte le nombre d'occurrences de chaque mot dans chaque document. Ainsi, chaque document est représenté par un vecteur de taille égale au nombre de mots dans le vocabulaire, où chaque élément du vecteur représente le nombre d'occurrences du mot correspondant dans le document. Le modèle CountVectorizer est très utile pour les tâches de classification de texte telles que la détection de spam ou la classification de documents.

Le modèle **TF-IDF (Term Frequency-Inverse Document Frequency)** est une méthode similaire au CountVectorizer, mais il prend également en compte la fréquence des mots dans le corpus global. Le TF-IDF mesure l'importance relative d'un mot dans un document en multipliant la fréquence d'apparition du mot dans le document (TF) par l'inverse de sa fréquence d'apparition dans tous les documents du corpus (IDF). Cela permet de mettre en évidence les mots qui sont fréquents dans un document mais peu fréquents dans l'ensemble du corpus, ce qui peut aider à identifier des mots-clés importants dans un document. Le modèle TF-IDF est souvent utilisé pour la recherche d'information, l'extraction de mots-clés et la recommandation de contenu personnalisé.

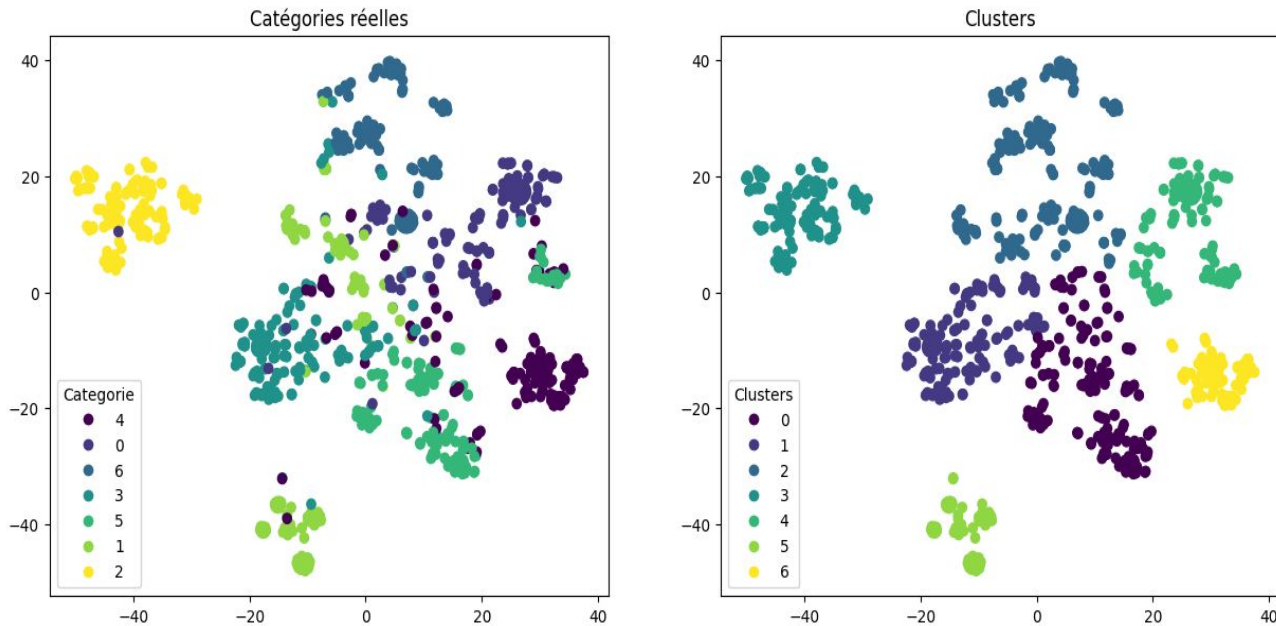
**USE (Universal Sentence Encoder)** est un modèle de traitement de langage naturel développé par Google qui permet de convertir du texte en vecteurs de grande dimension. Ces vecteurs permettent de représenter le sens des phrases ou des documents de manière plus riche que les approches traditionnelles de sac de mots.

**BERT (Bidirectional Encoder Representations from Transformers)** est un modèle de traitement de langage naturel développé par Google qui est capable de comprendre le sens des phrases en contexte. Il a été entraîné sur un large corpus de données textuelles, et est utilisé pour de nombreuses tâches de NLP, telles que la classification de textes ou la réponse à des questions.



# Analyse du traitement textes

## Visualisation du modèle TF-IDF



ARI: 0.5651 Temps d'exécution: 2.0

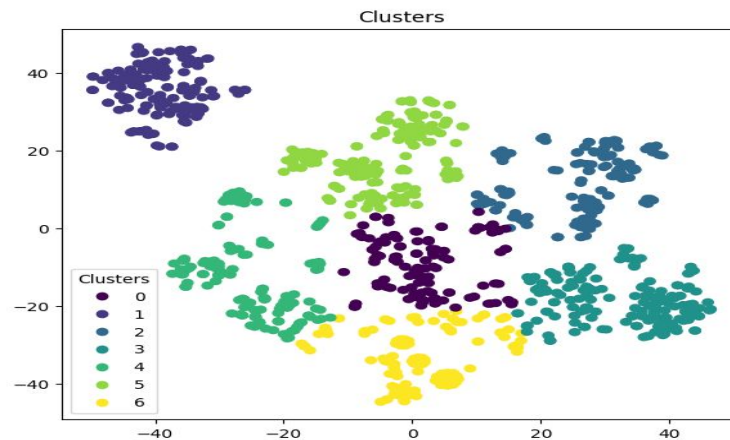
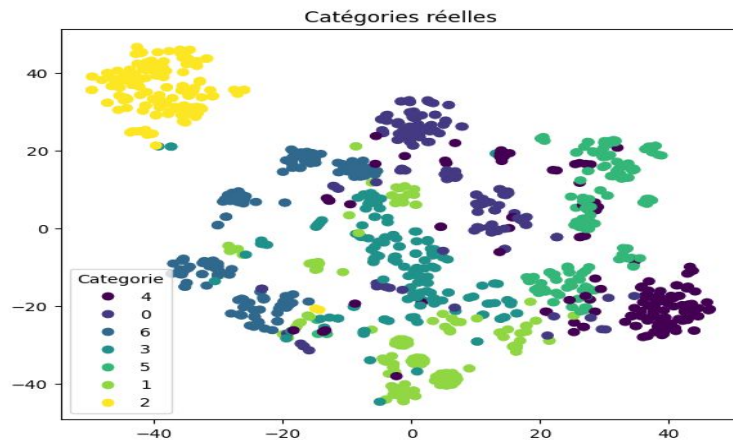
Le modèle TF-IDF présente le meilleur score à hauteur de 0.5651 pour les modèles sur le texte.

Cette méthode sera conservée dans le cadre de l'étape du traitement de textes.

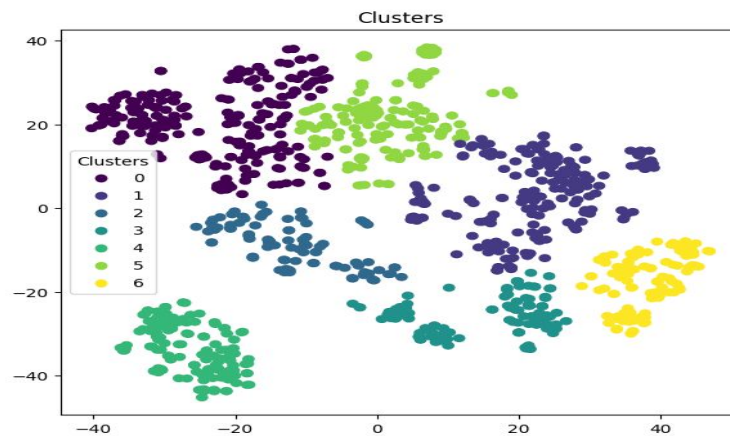
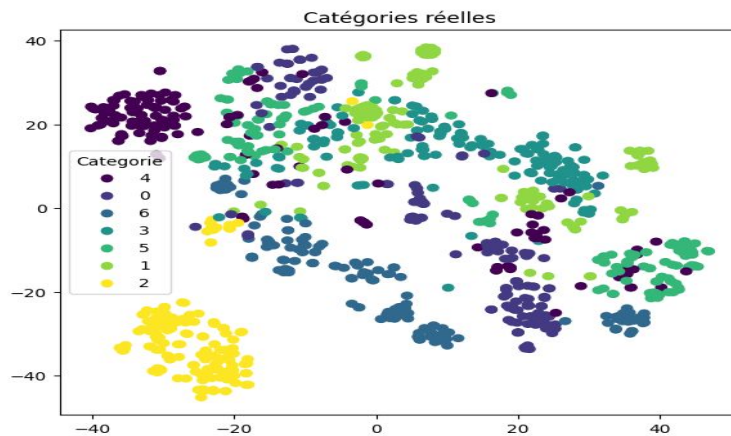




## Visualisation du modèle USE : ARI : 0.454 Temps d'exécution : 3.0



## Visualisation du modèle Bert: ARI: 0.3225 Temps d'exécution: 69.0





# Traitement des images

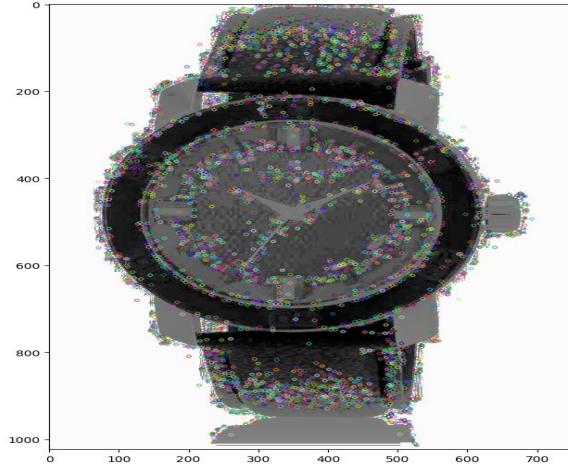
**SIFT (Scale-Invariant Feature Transform)** est une méthode de traitement d'images qui permet de détecter et de décrire les caractéristiques distinctives d'une image de manière invariante à l'échelle, à la rotation et à la translation. C'est un modèle développé en 1999.

**ORB (Oriented FAST and Rotated BRIEF)** est une méthode de traitement d'images qui combine la rapidité et l'efficacité de la méthode FAST (Features from Accelerated Segment Test) pour la détection de points d'intérêt avec la précision de la méthode BRIEF (Binary Robust Independent Elementary Features) pour la description de ces points d'intérêt. C'est un modèle développé en 2001.

**Un réseau de neurones convolutif (Convolutional Neural Network)** est un type de réseau de neurones artificiels qui est particulièrement bien adapté pour l'analyse d'images et de vidéos. Les CNN sont constitués de plusieurs couches, chacune ayant une fonction spécifique dans le traitement de l'image.

# Utilisation des descripteurs d'images

## Méthode SIFT



## Méthode ORB



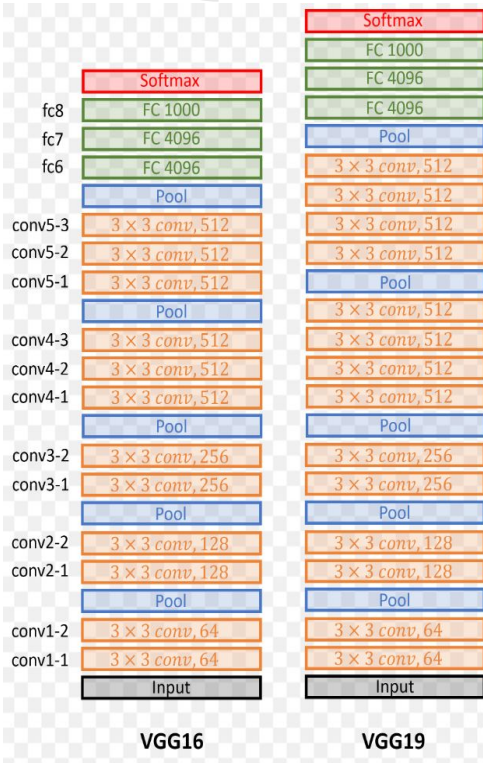
La méthode SIFT présente une analyse plus précise de l'image avec un nombre de descripteurs (4292, 128) qui s'adapte en fonction de notre donnée.

La méthode ORB utilise un système de 500 descripteurs.

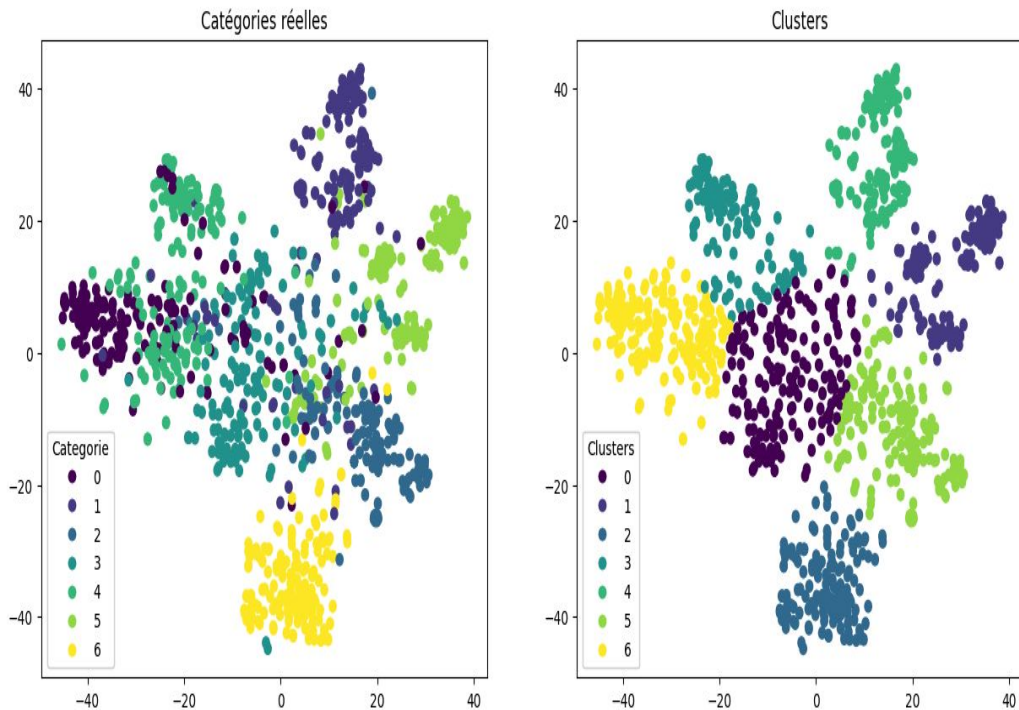
Résultat : conservation de la méthode SIFT pour le test de cette méthode de traitement.

# Analyse traitement images

## Architecture d'un modèle CNN



## Segmentation avec le VGG19







# Résultats étude de faisabilité



## Textes

	ARI Score	Temps (s)
CountVectorizer	0.4182	3
TF-IDF	0.5017	2
BERT	0.3225	70
USE	0.454	4

Les méthodes TF-IDF et USE présentent les meilleurs résultats et délai de traitement rapide.

## Images

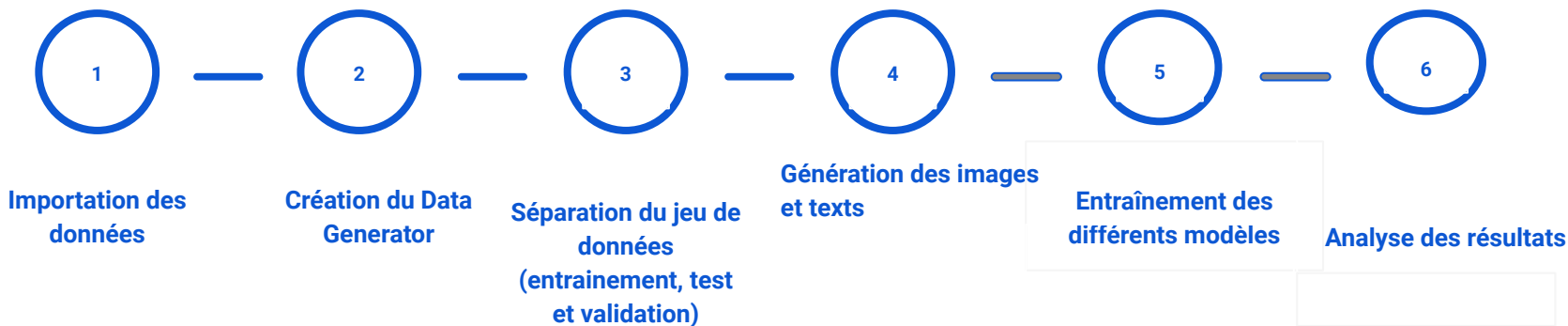
	ARI Score
SIFT	0.454
VGG16	0.49
VGG19	0.5339

La méthode VGG19 présente les meilleurs résultats.


# III. Classification supervisée



# *Étape de la classification*

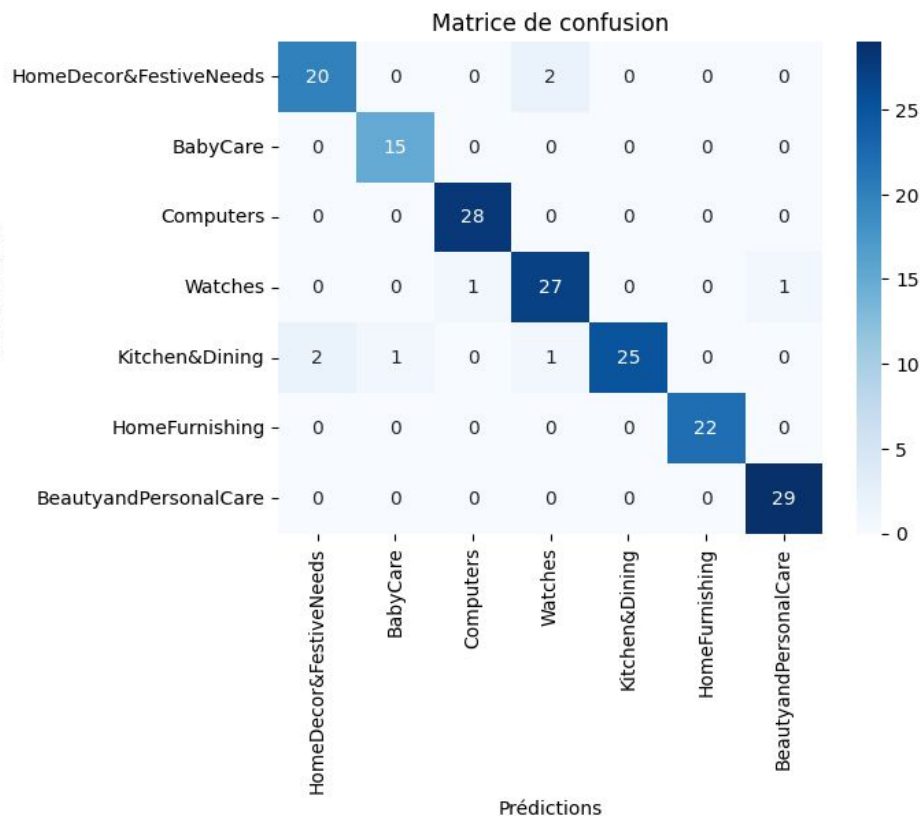


# Résultats de la classification

	Accuracy	Délai entraînement	Epochs entraînés
CNN	0.43	352.94 seconds	12
VGG19	0.5339	56 minutes	20
RESNET 50	0.48	10 minutes	20
BERT	0.647	73.88 seconds	1
BERT + ResNet-50	0.6936	229.77 seconds	1
ALBEF	0.745	1775 seconds	1
BERT	0.9367	25 minutes	20
BERT + ResNet-50	0.9482	55 minutes	20
ALBEF	0.95402	160 minutes	20

# rapport de classification ALBEF

Valeurs réelles

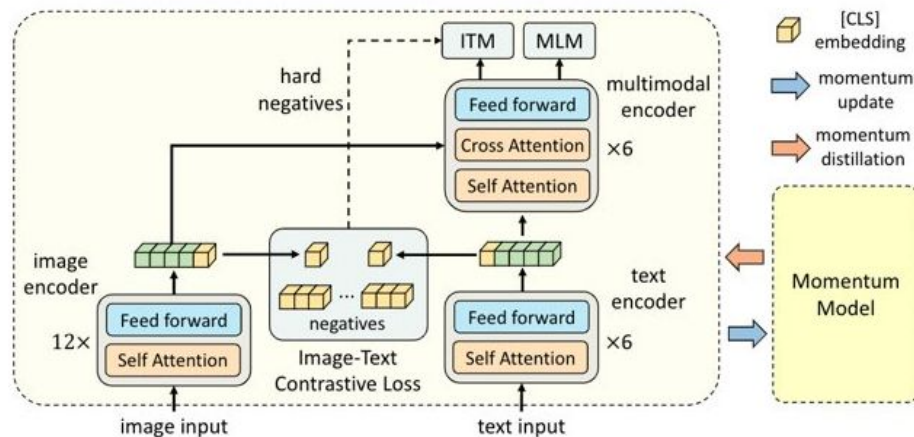


	precision	recall	f1-score	support
HomeDecor&FestiveNeeds	0.91	0.91	0.91	22
BabyCare	0.94	1.00	0.97	15
Computers	0.97	1.00	0.98	28
Watches	0.90	0.93	0.92	29
Kitchen&Dining	1.00	0.86	0.93	29
HomeFurnishing	1.00	1.00	1.00	22
BeautyandPersonalCare	0.97	1.00	0.98	29
accuracy			0.95	174
macro avg	0.95	0.96	0.95	174
weighted avg	0.96	0.95	0.95	174



Output Classes: **7**

Fully Connected Layer



ALBEF  
combined  
text-image  
embedding  
dimension:  
**768**

"Align before fuse: Vision and language representation learning with momentum distillation." Li, et al. NeurIPS, 2021.

## IV. Collecte API





# *Étape de la collecte*





# V. Conclusion



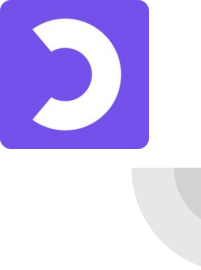


En conclusion, ce projet a été un succès grâce aux différentes méthodes de traitement d'images et de texte que nous avons utilisées. Malgré un nombre limité de données, soit 1050 lignes, nous avons réussi à segmenter les catégories de produits de manière efficace.

Nous avons également réalisé avec succès la classification d'images en utilisant un générateur d'images pour augmenter la quantité de données et en utilisant le Transfer Learning pour sélectionner le meilleur modèle.

Grâce à notre collecte API, nous avons également pu obtenir de nouvelles informations à faible coût.

Pour améliorer encore davantage les performances de classification , nous avons envisager l'encodeur conjoint texte-image dans ALBEF pour combiner les encodeurs de texte et d'image, permettant au modèle de classifieurs les produits automatiquement à partir des deux modalités.



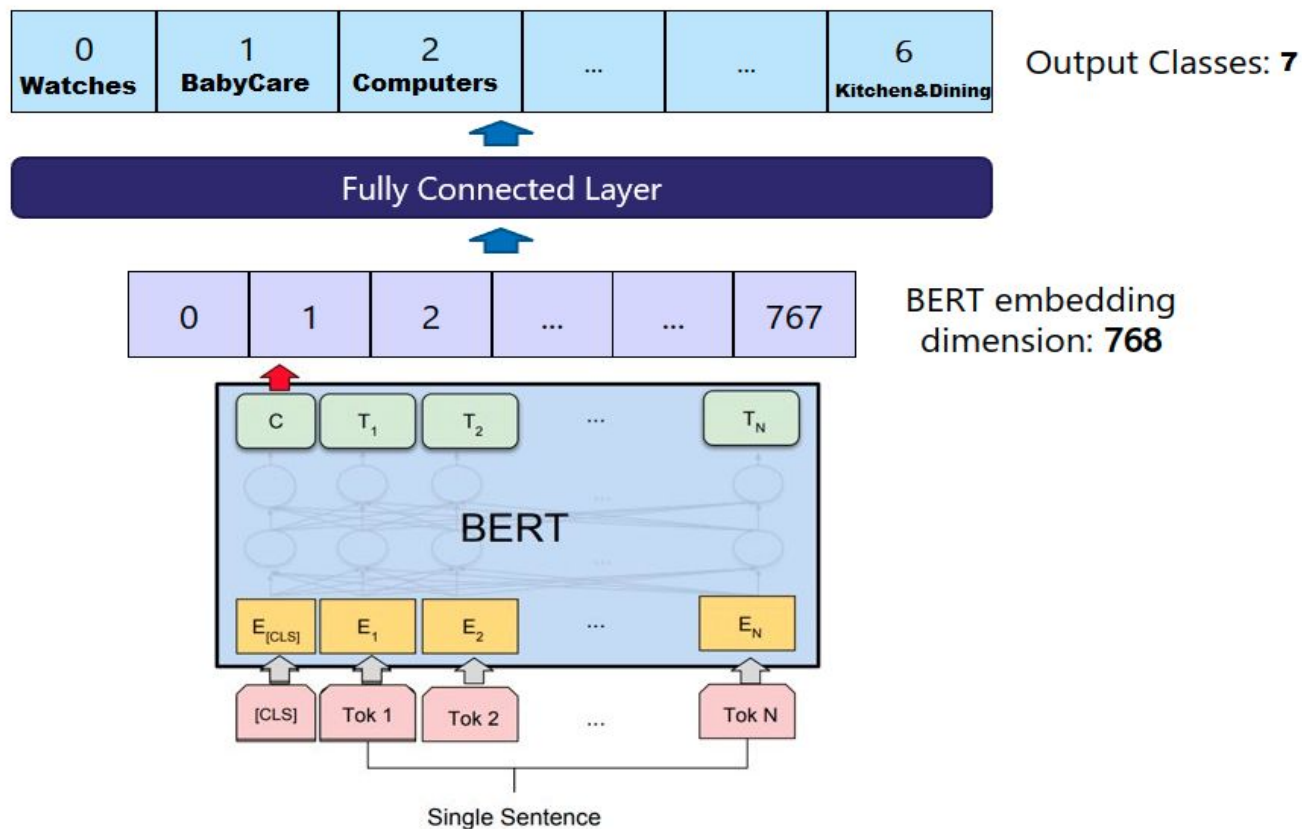
*merci*

– BEACOU –

**POUR VOTRE ATTENTION**

# Annexe





"Bert: Pre-training of deep bidirectional transformers for language understanding." Devlin, et al. *arXiv:1810.04805*, 2018.

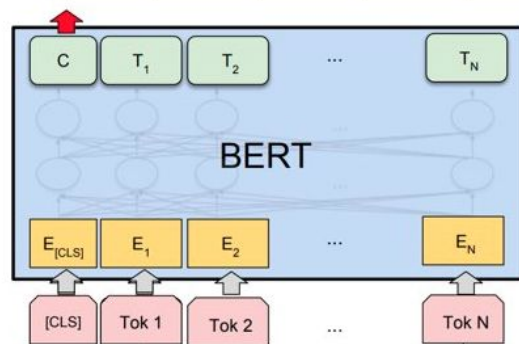
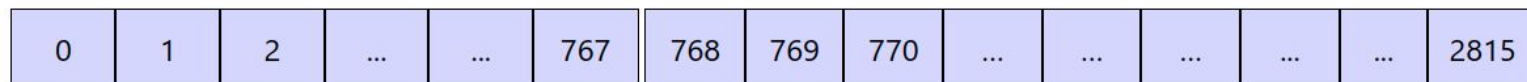


Output Classes: 7

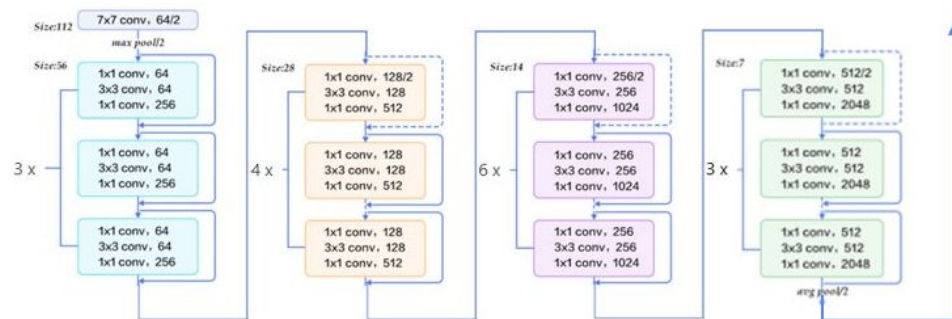
Fully Connected Layer

BERT embedding  
dimension: 768

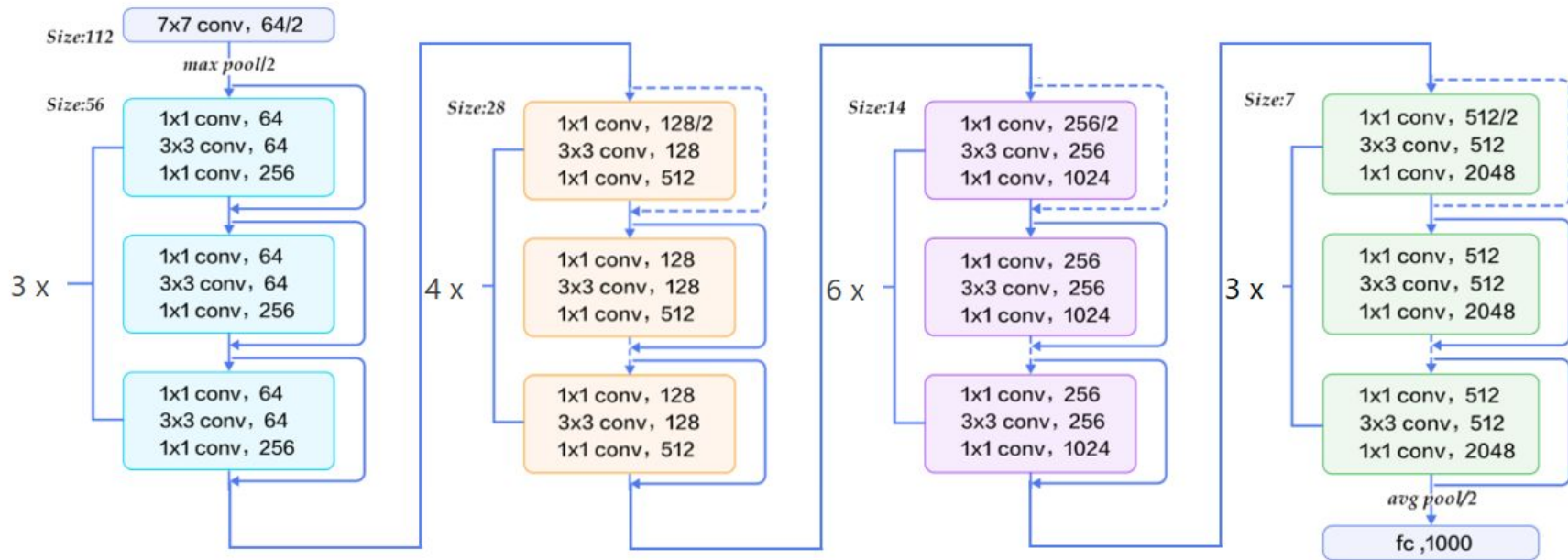
ResNet50 dimension  
(after avg pool):  
2048



Single Sentence

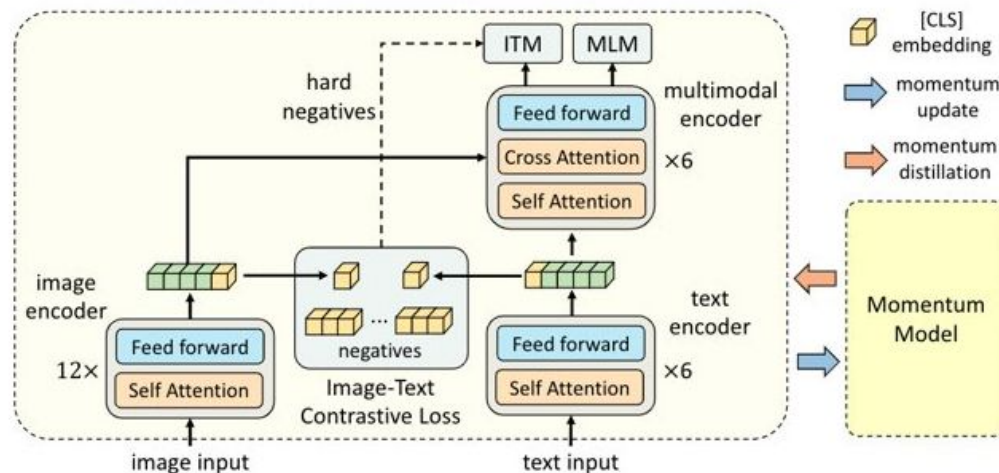
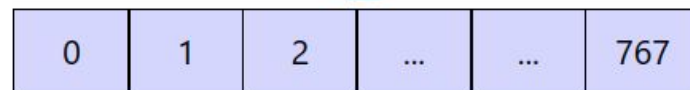


"Deep residual learning for image recognition." He, et al. CVPR, 2016.





Output Classes: 7



ALBEF  
combined  
text-image  
embedding  
dimension:  
**768**

"Align before fuse: Vision and language representation learning with momentum distillation." Li, et al. NeurIPS, 2021.