CLASSIFICATION AUTOMATIQUE

PLACE DE MARCHÉ

SOMMAIRE

1) Contexte du projet
2) Etude de faisabilité de classification automatique
3) Classification supervisées des produits
4) Test de collecte de données

1) CONTEXTE DU PROJET

Key Features of Elegance Polyester Multicolor Abstract Eyelet Door Curtain Floral...

Specifications of Sathiyas Cotton Bath Towel (3 Bath Towel, Red, Yellow, Blue)...

Home Furnishing

Baby Care


```
const options = {
  method: 'GET',
  url: 'https://edamam-food-and-grocery-database.p.rapidapi.com/api/food-database/v2/parser',
  params: {
    'nutrition-type': 'cooking',
    'category[0]': 'generic-foods',
    'health[0]': 'alcohol-free'
  },
  headers: {
    'X-RapidAPI-Key': '7f7a73e0c7msh3a250c73c1e427cp179969jsn791ff966f28f',
    'X-RapidAPI-Host': 'edamam-food-and-grocery-database.p.rapidapi.com'
  }
```

Edamam Food and Grocery Database

1) CONTEXTE DU PROJET

Les données :

1050 images

description	Nom image	catégorie
Key Features of El…	55b85ea15a1536d46b7190ad6fff8ce7.jpg	Home Furnishing (150)
		Baby Care (150)
		Watches (150)
		Home Decor & Festive Needs (150)
		Kitchen & Dining (150)
		Beauty and Personal Care (150)
		Computers (150)

1050 lignes – 7 catégories

2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

A partir des descriptions

- Pré-traitement du texte

Mots de liaison (stopwords)
Ponctuation
Fréquences, taille…

```
texte initial :
Key Features of Elegance Polyester Multicolor Abstract Eyelet D

texte stemmé :
key featur eleg polyest multicolor abstract eyelet door curtai
temps pour stemmer :  0.00836  s.


texte lemmé :
key feature elegance polyester multicolor abstract eyelet door
temps pour lemmer :  2.36476  s.
```

	words	occurency	characters	contains_digit	nbr_descriptions
0	key	223	3	False	202
1	feature	404	7	False	267
2	elegance	11	8	False	6
3	polyester	68	9	False	28
4	multicolor	160	10	False	87

- +200 descriptions
- taille < 2
- chiffres

```
stop_words.update(stop_words_update)
```

2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

A partir des descriptions

Méthodes basiques

- **Comptage de mot**
- **Tf-IDF**

```
input_data = ['I am rich, and the world is mine !',
              'I will be rich, because I have to change my life.',
              'Time is money, but money does not give you back time.']
```

Comptage de mot

	back	change	give	life	mine	money	rich	time	world
0	0	0	0	0	1	0	1	0	1
1	0	1	0	1	0	0	1	0	0
2	1	0	1	0	0	2	0	2	0

Tf-IDF

back	change	give	life	mine	money	rich	time	world
0.000000	0.000000	0.000000	0.000000	0.622766	0.000000	0.47363	0.000000	0.622766
0.000000	0.622766	0.000000	0.622766	0.000000	0.000000	0.47363	0.000000	0.000000
0.316228	0.000000	0.316228	0.000000	0.000000	0.632456	0.00000	0.632456	0.000000

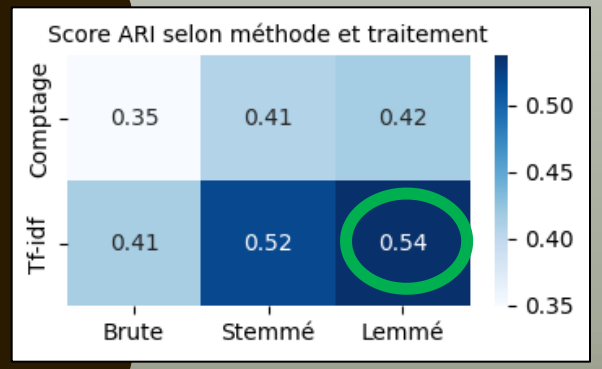
$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

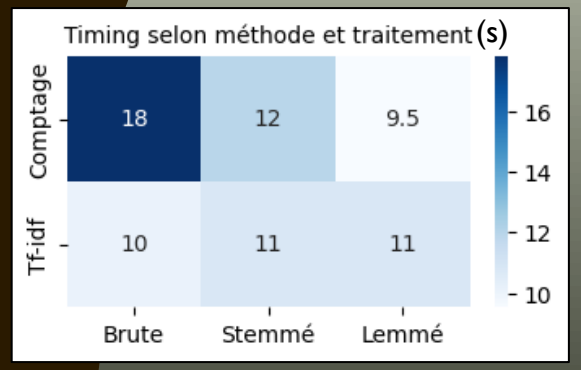
2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

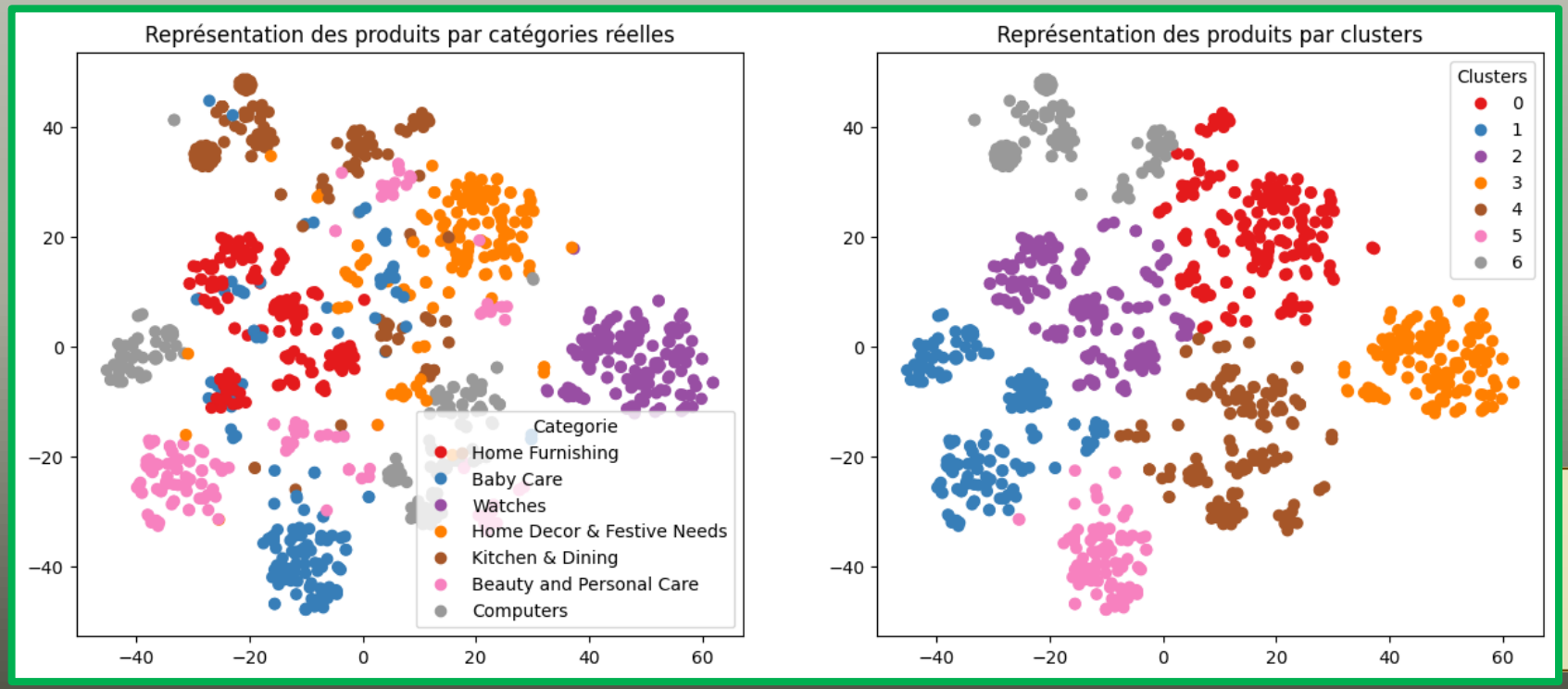
A partir des descriptions

Méthodes basiques

- **Comptage de mot**
- **Tf-IDF**







2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

A partir des descriptions

Méthodes plongement de mots (Word Embedding)

- **Word2vec**
- **BERT**
- **USE**

Réseau peu profond

Window = 3 Skip-gram

```
input_data = ['I am rich, and the world is mine !',
              'I will be rich, because I have to change my life.',
              'Time is money, but money does not give you back time.']
```

Vector_size = 300 Dim([0.23, 0.04, ..., 0.78]) = (300,)

Train

Représentation vectorielle de chaque mot **dim(nb_mots, 300)**

[Vecteur_mot_1
Vecteur_mot_2
Vecteur_mot_3
Vecteur_mot_4
Vecteur_mot_5
...
Vecteur_mot_**nb_mots**]

Moyenne pour chaque phrase

2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

A partir des descriptions

Méthodes plongement de mots (Word Embedding)

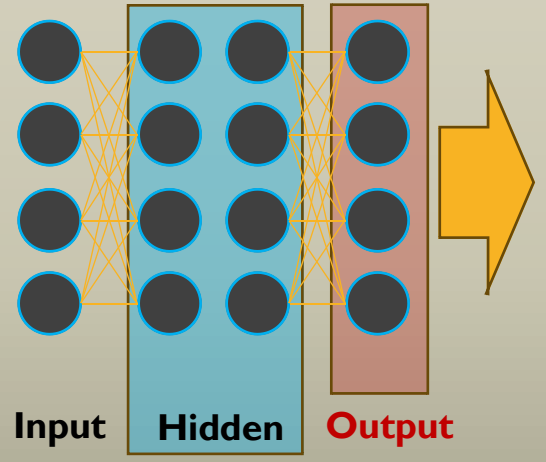
- **Word2vec**
- **BERT**
- **USE**

Transformer, Réseau profond avec attention

Texte

tokenizer.encode_plus

- **attention_mask**
- **input_ids**



Poids bert-base-uncased pré-entraîné

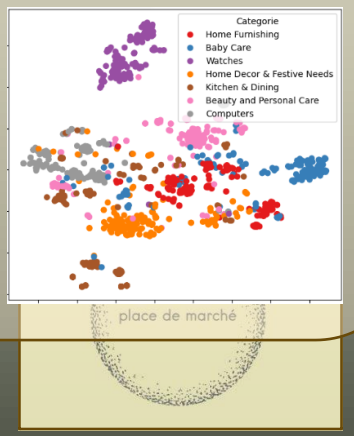
Array(batch, len_phrase, Output_dim)

Mean

Array(batch, [redacted], Output_dim)

Array(batch, Output_dim)

Pca 0,99
TSNE 2D



2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

A partir des descriptions

Méthodes plongement de mots (Word Embedding)

- **Word2vec**
- **BERT**
- **USE**

Réseau profond

```
input_data = ['I am rich, and the world is mine !',
              'I will be rich, because I have to change my life.',
              'Time is money, but money does not give you back time.']
```

Représentation vectorielle de chaque phrase : shape(512 ,)

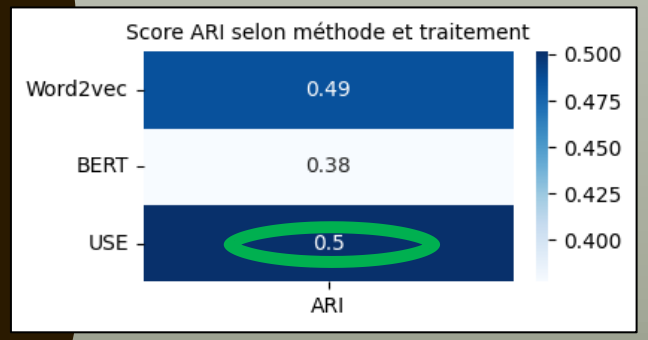
[Vecteur_phrase_1
Vecteur_ phrase _2
Vecteur_ phrase _3]

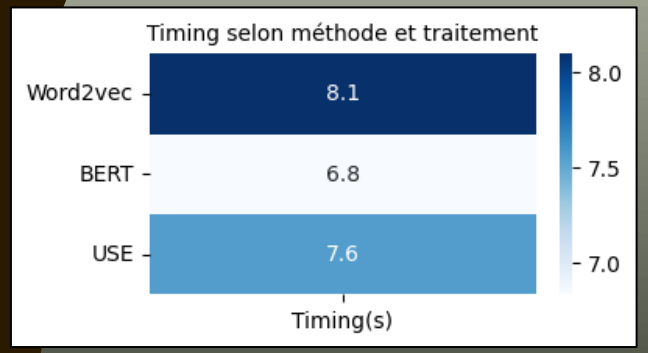
2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

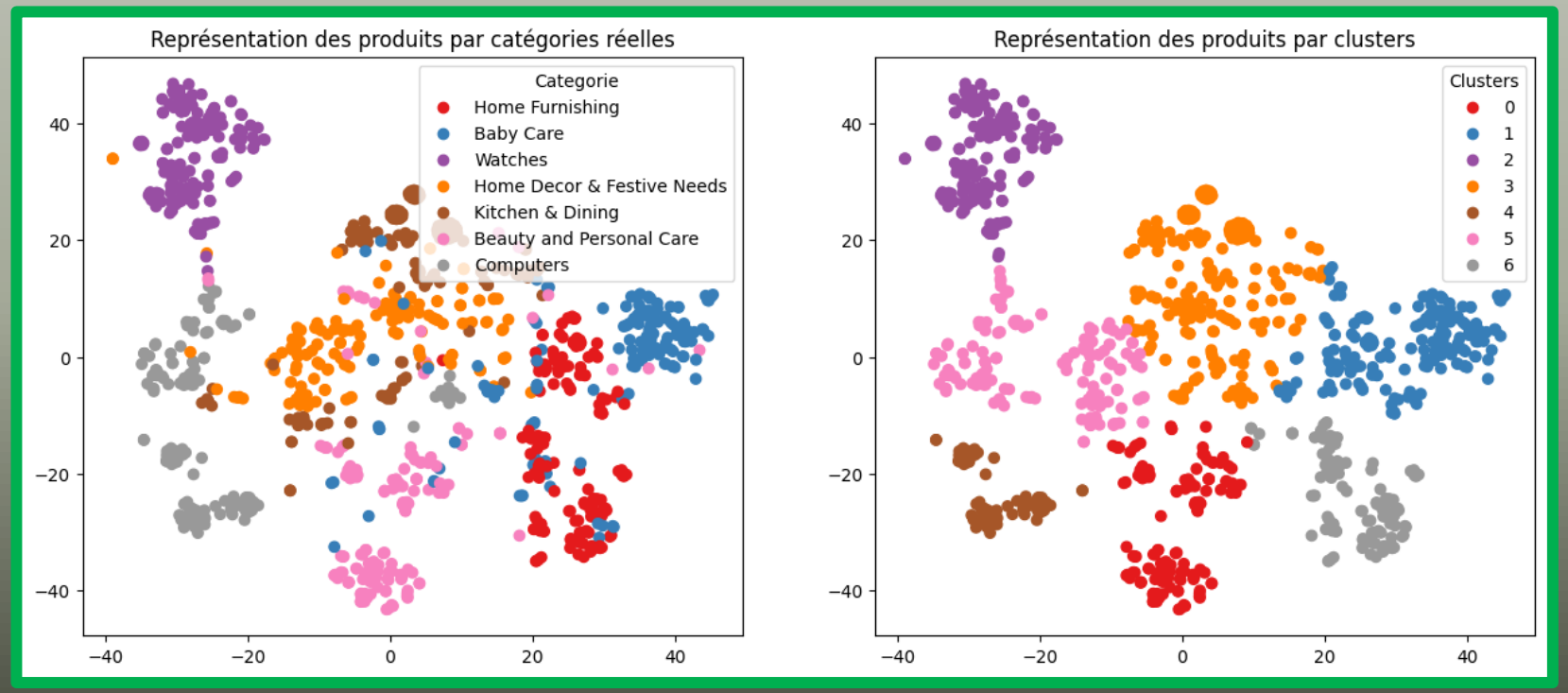
A partir des descriptions

Méthodes plongement de mots (Word Embedding)

- **Word2vec**
- **BERT**
- **USE**



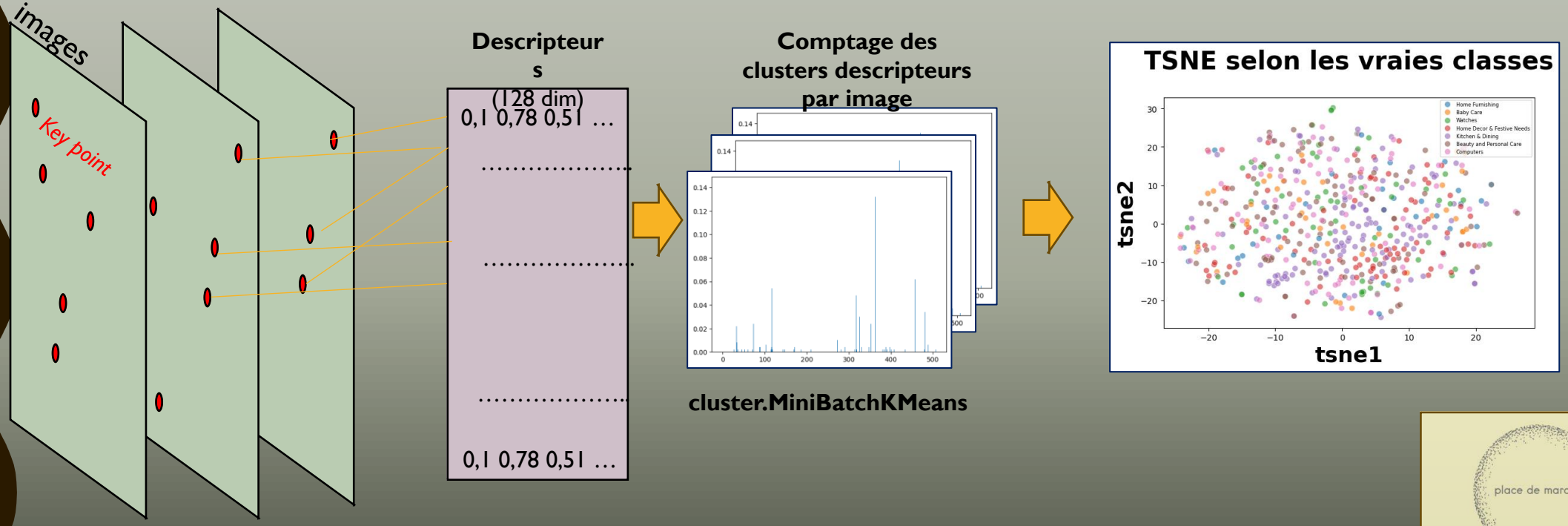




2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

A partir des images

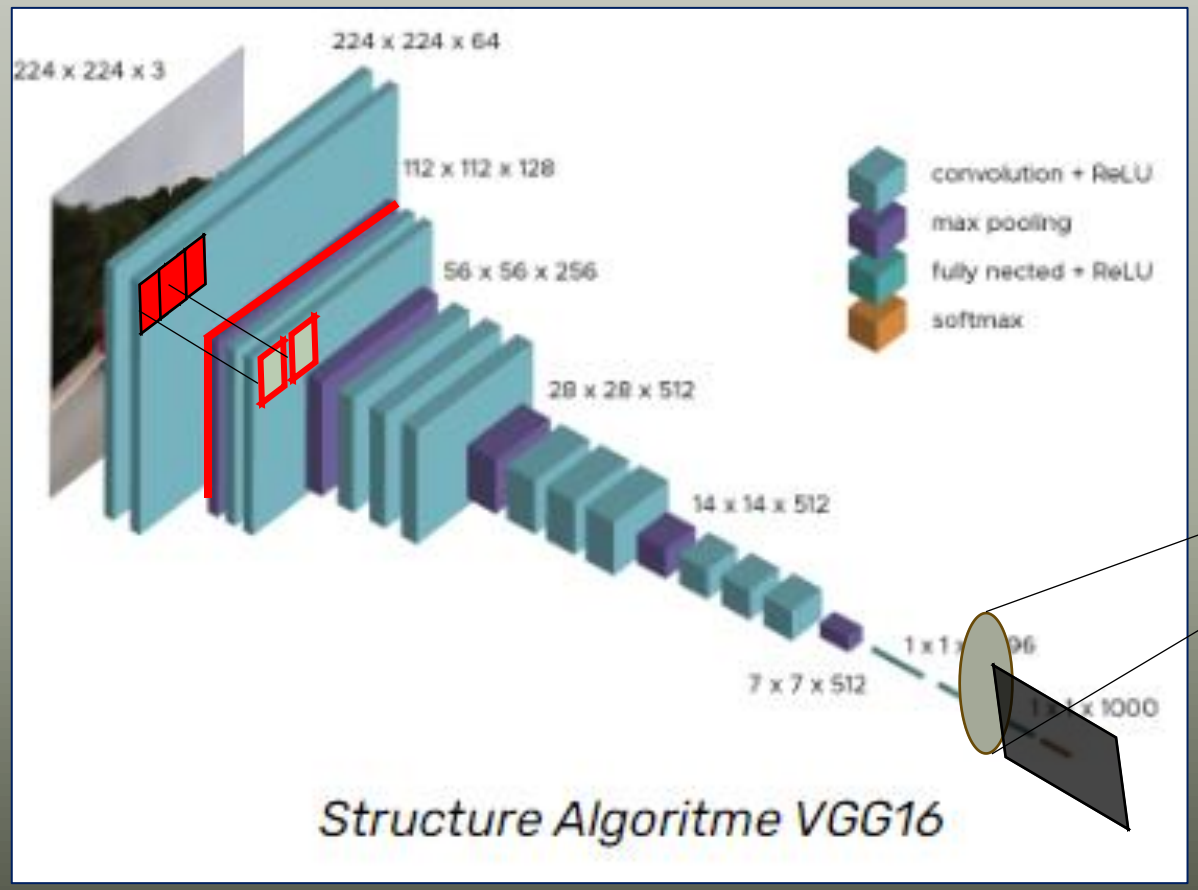
- **SIFT**
- **VGG16**



2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

A partir des images

- **SIFT**
- **VGG16**

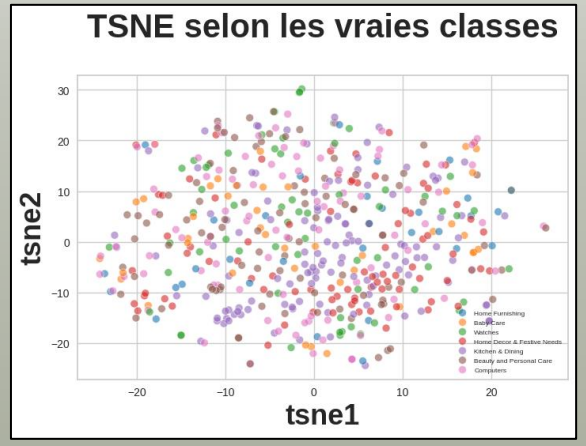


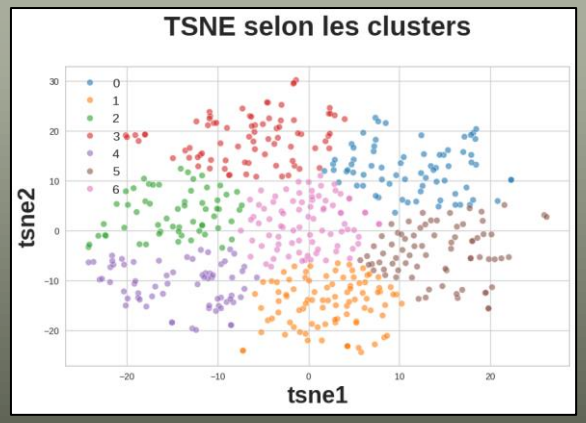
2) ETUDE DE FAISABILITÉ DE CLASSIFICATION AUTOMATIQUE

A partir des images

- **SIFT**
- **VGG16**

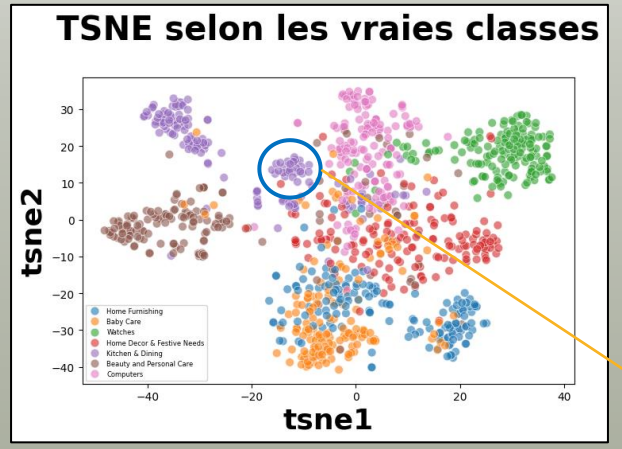
SIFT

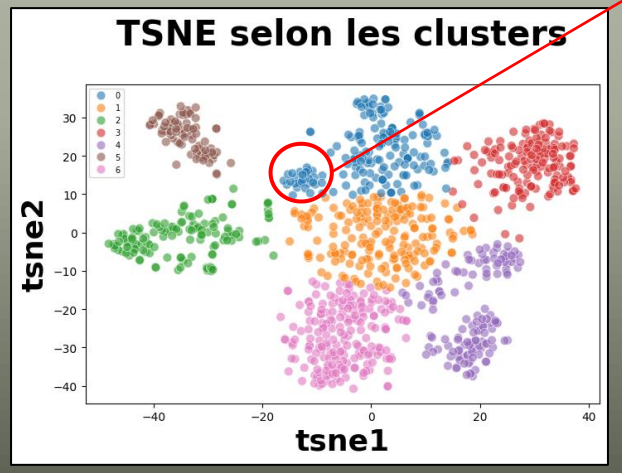




ARI : 0.071

VGG16





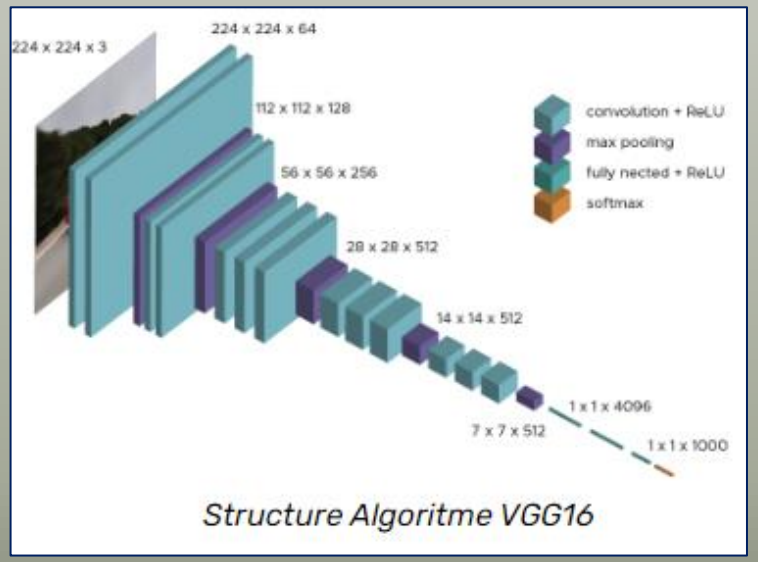
ARI : 0. 452

Kitchen dans computer

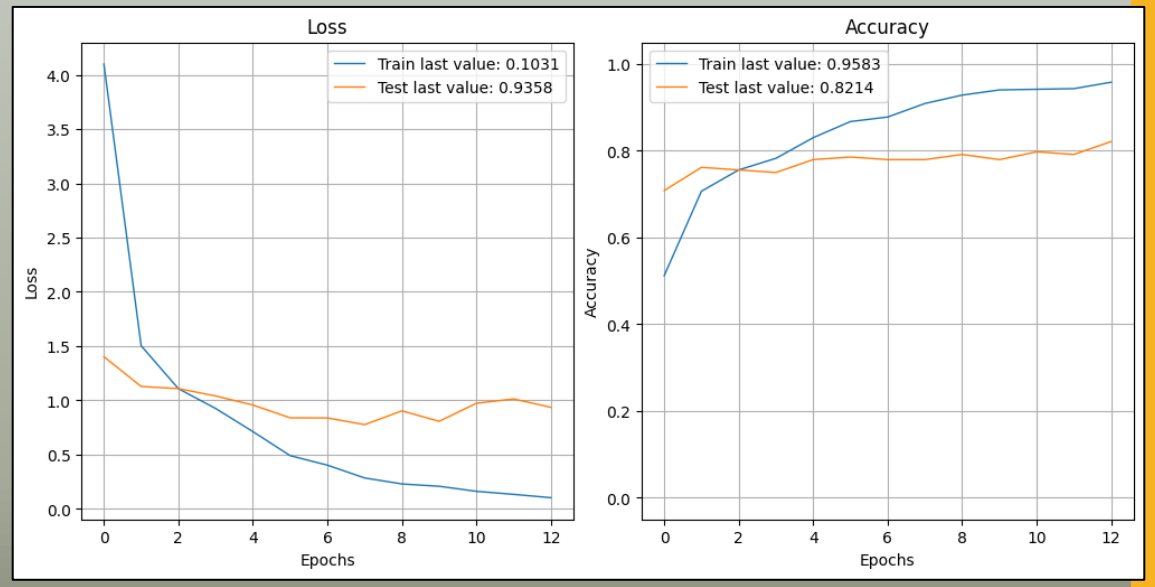
3) CLASSIFICATION SUPERVISÉE DES PRODUITS

4 approches

Approche 1 : préparation amont des images



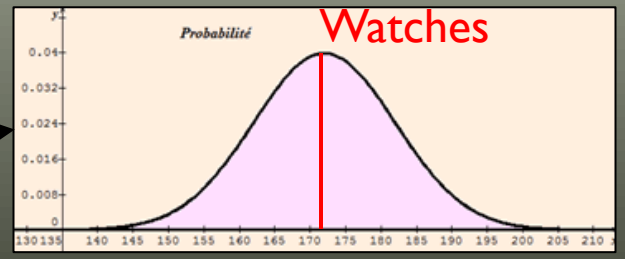
Structure Algoritme VGG16



- Array
- Normalisation
- Reshape(224, 224, 3)

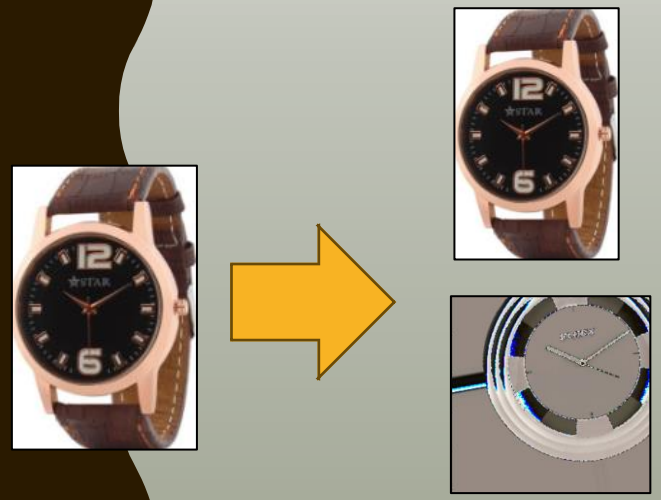
```
Dense(7, activation='softmax')
```

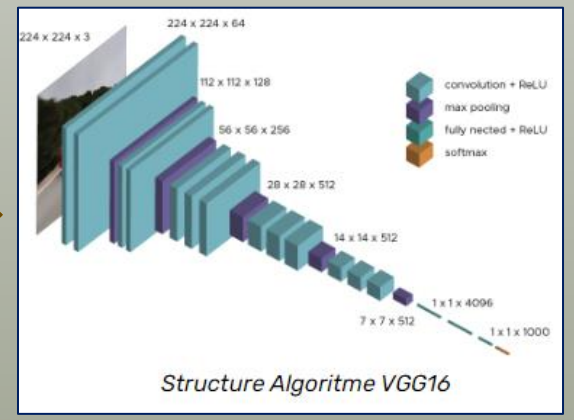
$$\mathrm{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}}$$



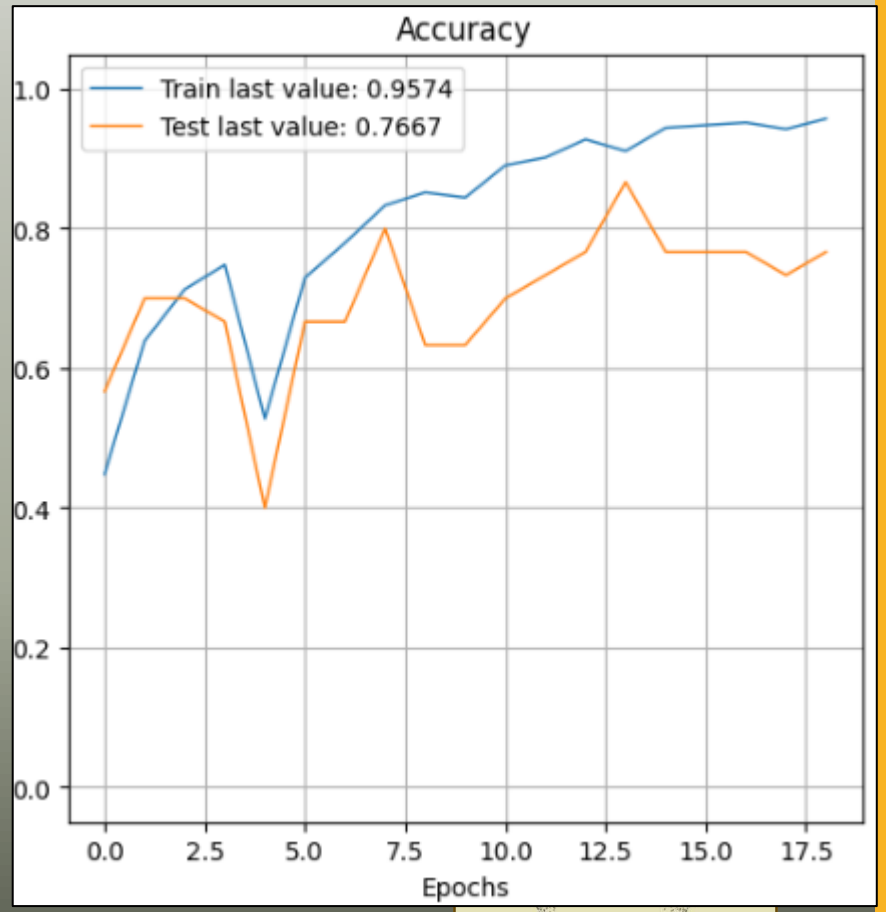
3) CLASSIFICATION SUPERVISÉE DES PRODUITS

Approche 2 : préparation amont des images et data augmentation



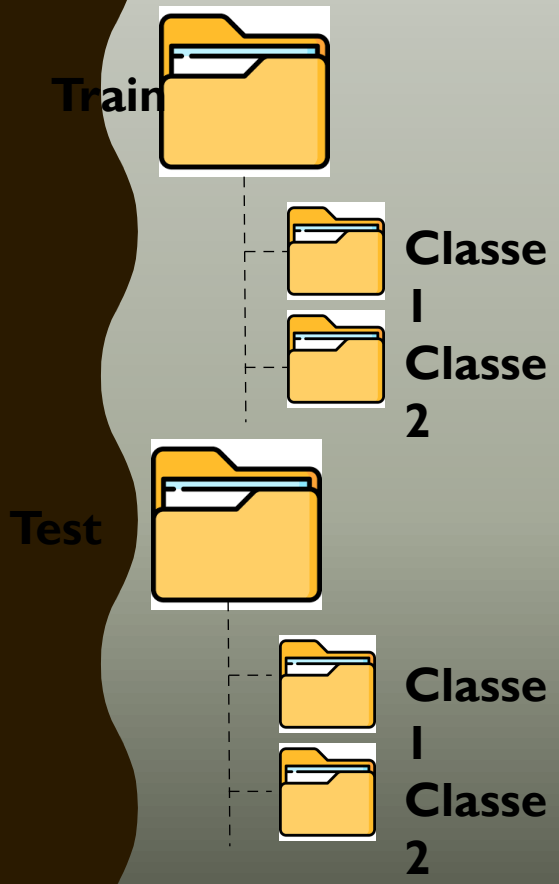


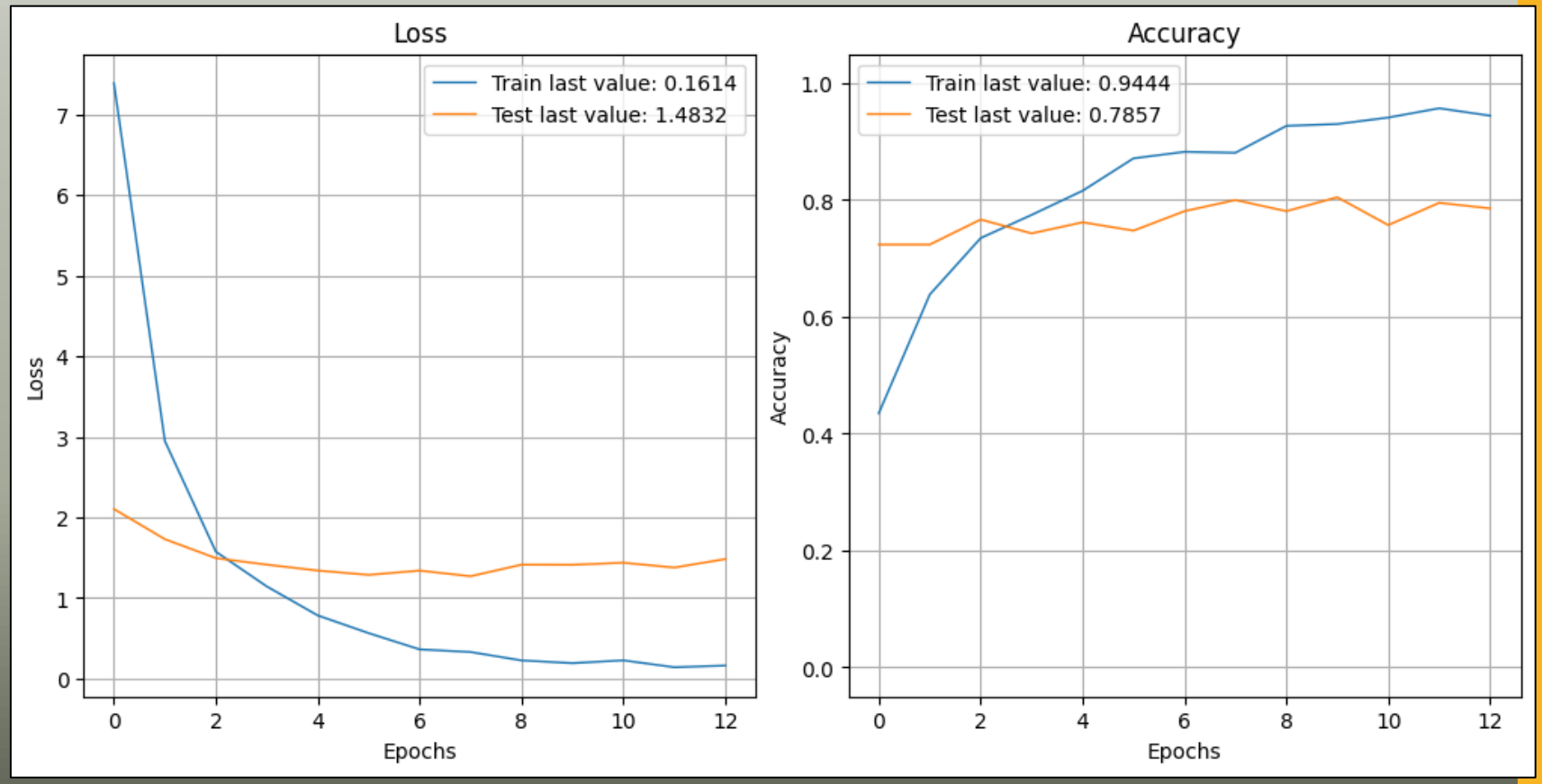
Structure Algoritme VGG16



3) CLASSIFICATION SUPERVISÉE DES PRODUITS

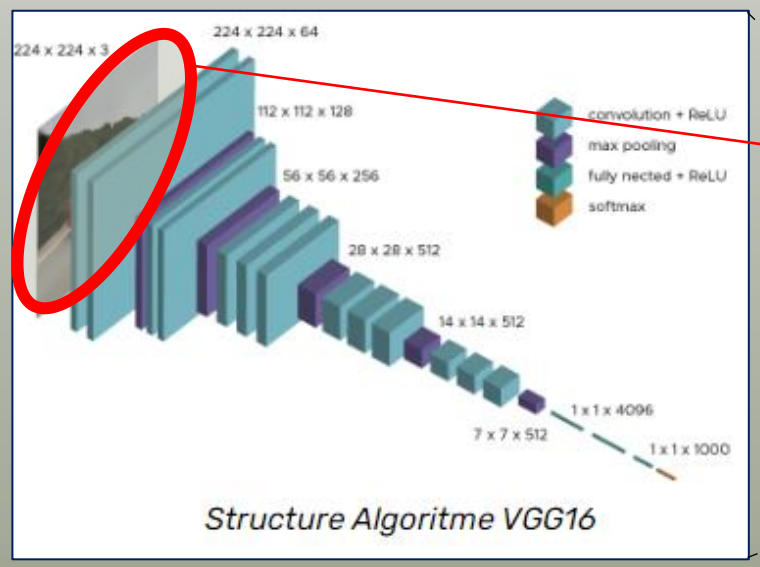
Approche 3 : image dataset



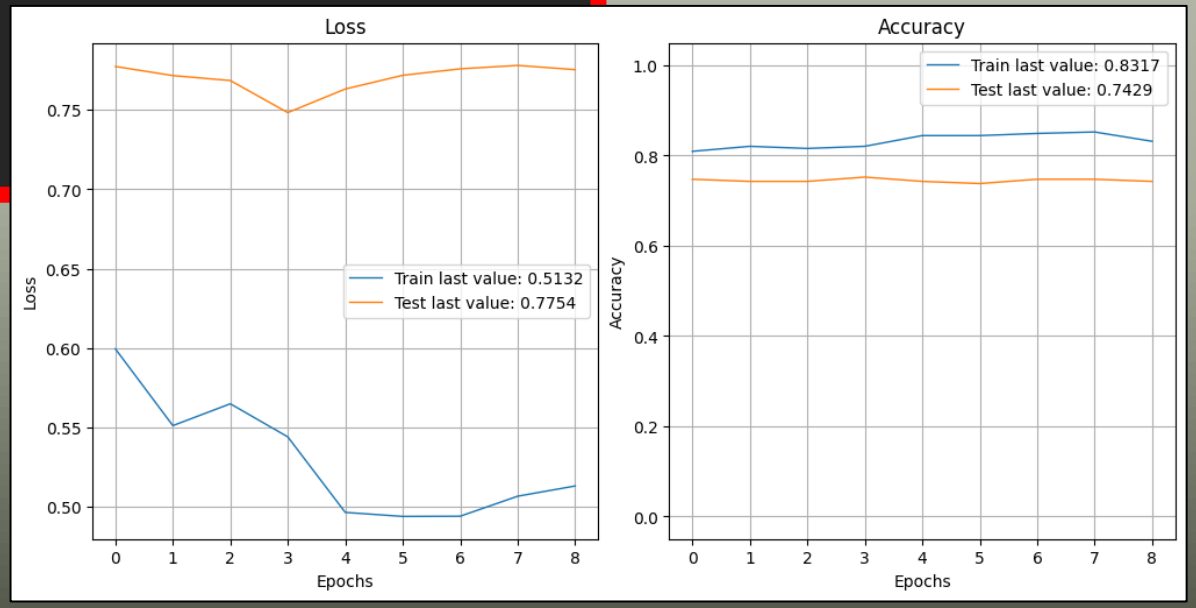


3) CLASSIFICATION SUPERVISÉE DES PRODUITS

Approche 4 : image dataset avec data augmentation intégrée

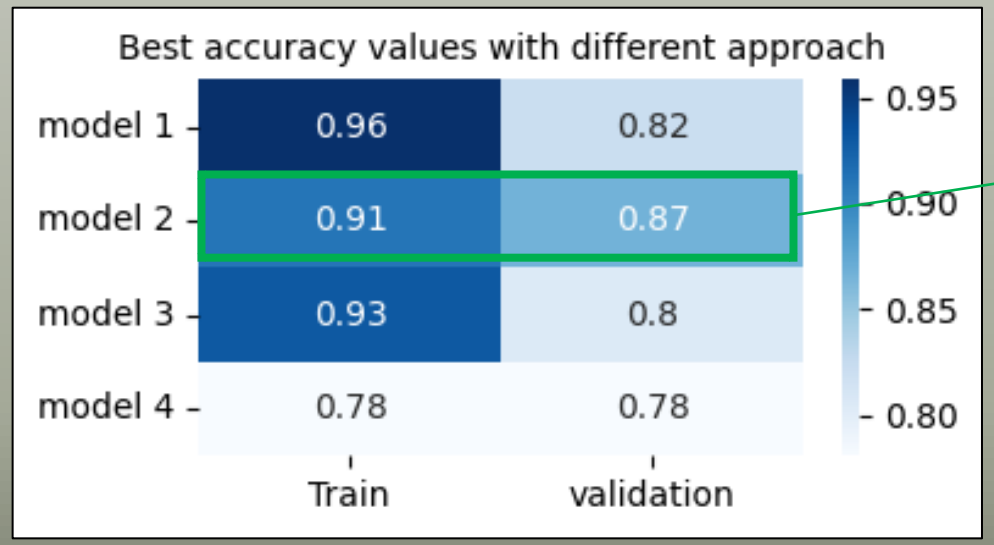



```
model = Sequential([
    data_augmentation,
    Rescaling(1./127.5, offset=-1),
    model_base,
    GlobalAveragePooling2D(),
```



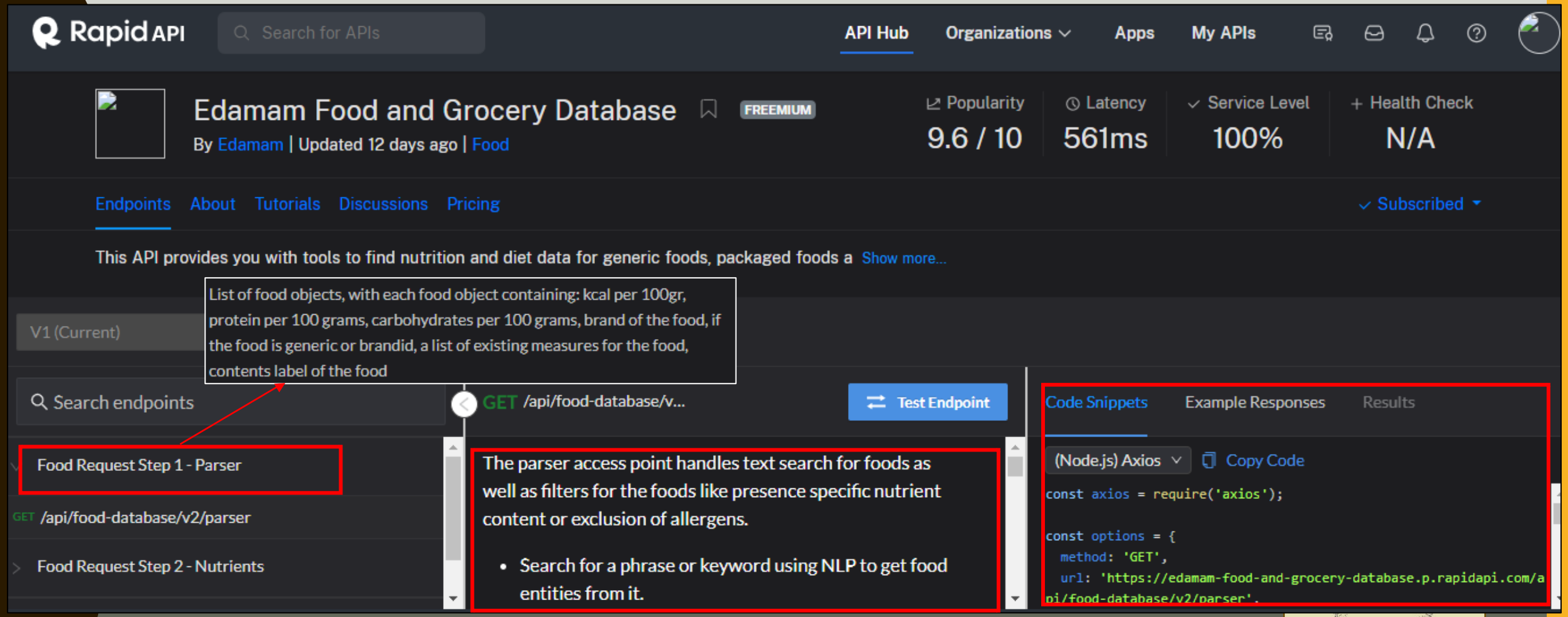
3) CLASSIFICATION SUPERVISÉE DES PRODUITS

Approche 2 : préparation amont des images et data augmentation



Evaluation accuracy test set : 0.8762

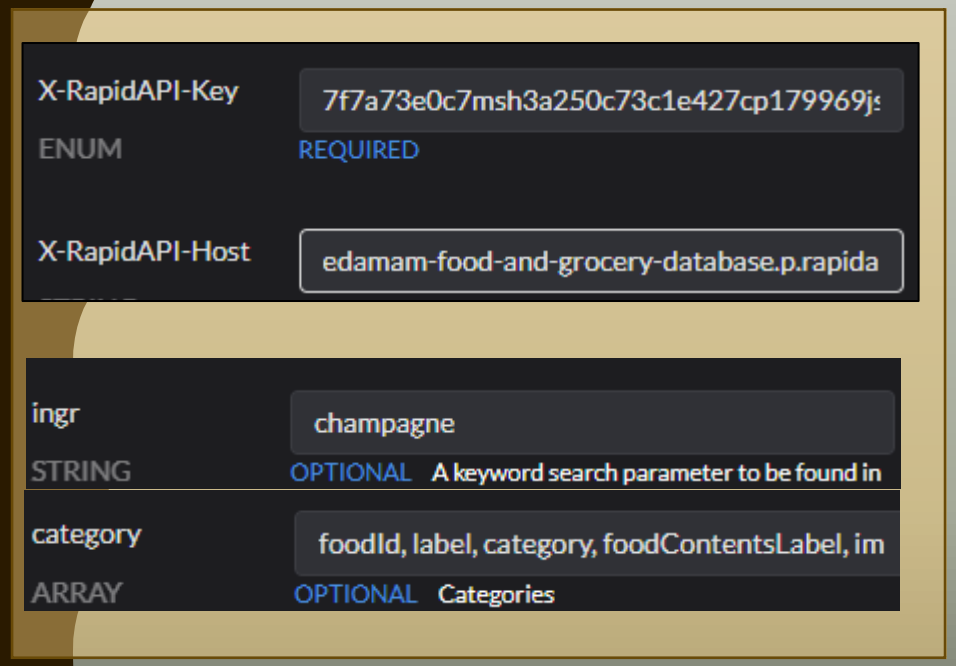
4) TEST DE COLLECTE DE DONNÉES

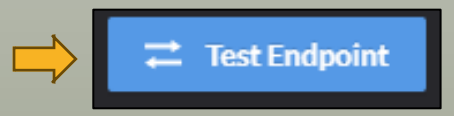


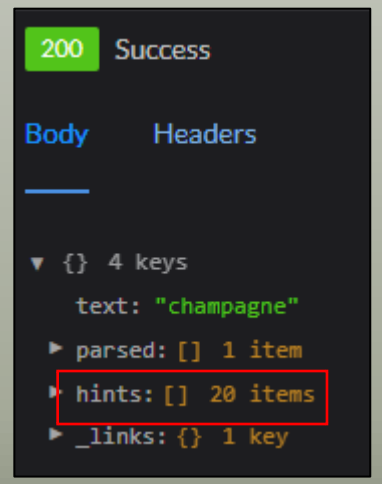
Clés API, paramètres

Code généré et résultat

4) TEST DE COLLECTE DE DONNÉES







Nous obtenons 20 produits.

FIN

Merci de votre attention