Saidali Bacar Abdallah

PROJET 6

CLASSIFICATION AUTOMATIQUE DE BIENS DE CONSOMMATION

SOMMAIRE

INTRODUCTION

Le Client: l'entreprise dans laquelle on travaille, "Place de marché"

La problématique: des vendeurs proposent des articles sur un site de marketplace e-commerce en postant une photo et une description. Ensuite ils attribuent manuellement une catégorie à chaque article.

La mission: automatiser le processus d'attribution des catégories afin de remédier aux erreurs constatées jusque la.

INTRODUCTION

Le fichier

1 seul fichier de type CSV, nb de variable : 15, nb de lignes: 1050.

- très peu de valeurs manquantes, 3 variables concernées.
- pas de doublon
- pas d'outlier

les variables

- **unique id:** identifiant unique du vendeur qui poste un produit le site marketplace
- **crawl timestamp:** date et heure d'enregistrement du produit
- **product url:** adresse web produit
- "product_name": nom du produit
- **product category tree:** arborescence produit
- **pid:**product id
- **retail price:** prix du produit
- **discount price:** montant de la remise sur le produit, solde
- **image:** image associé au produit
- **is_FK_Advantage_product:**
- **description:** description du produit
- **product rating:** evaluation du produit, semble avoir peu de renseignement
- **overall rating:** evaluation globale, semble avoir peu de renseignement
- **brand:** marque
- **product specifications:** spécifications sur le produit

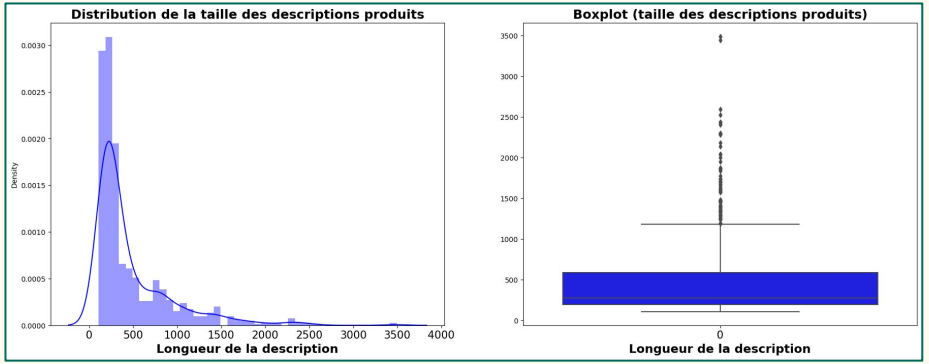
PARTIE I: EXTRACTION DE FEATURES SUR LES DONNÉES TEXTES

la variable : “description”

les différentes étapes de préparation des données

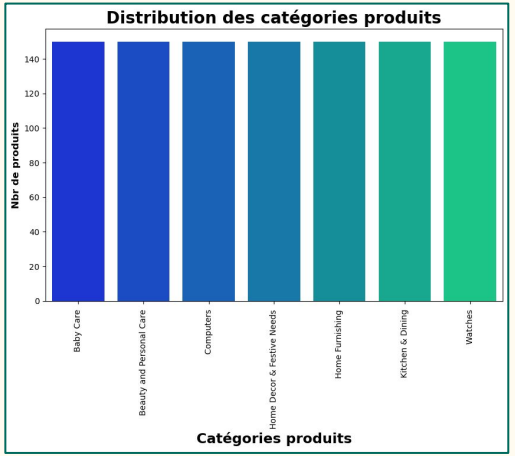
- tokenisation
- stopwords + mots <2 lettres
- les caractères alphanumériques
- la lemmatisation

outil: librairie nltk



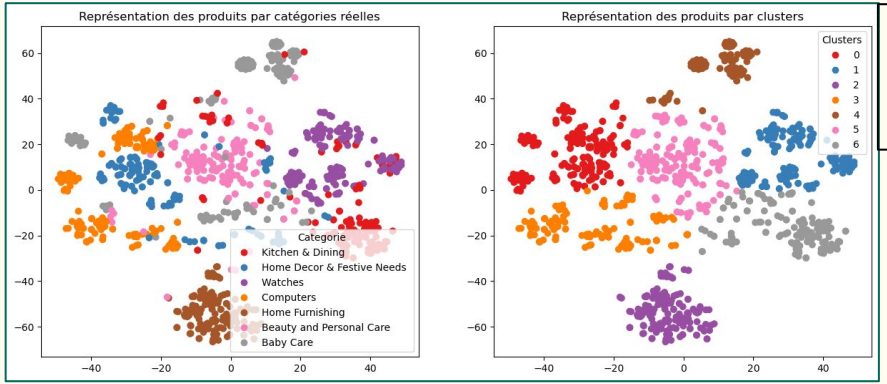
les 7 catégories:

- baby care
- beauty and personal care
- computers
- home decors and festive needs
- home furnishing
- kitchen and dining
- watches



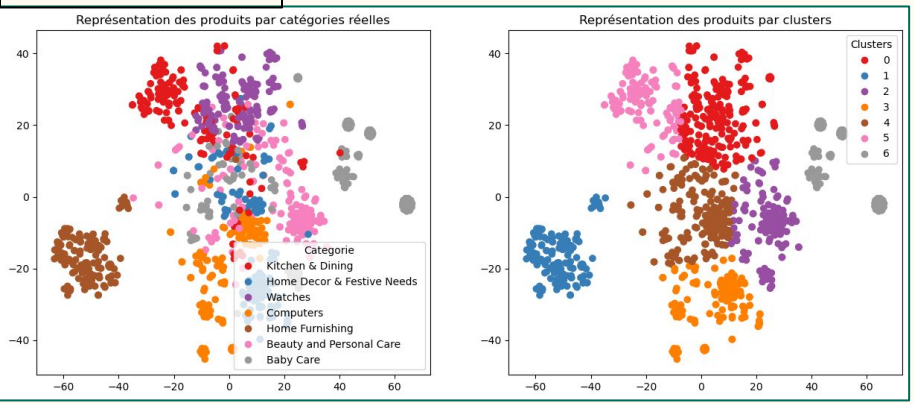
PARTIE I: EXTRACTION DE FEATURES SUR LES DONNÉES TEXTES

bag-of-words : count vectoriser et tf-idf vectoriser



tf-idf vectoriser
ARI: 0,53

count vectoriser
ARI: 0,41

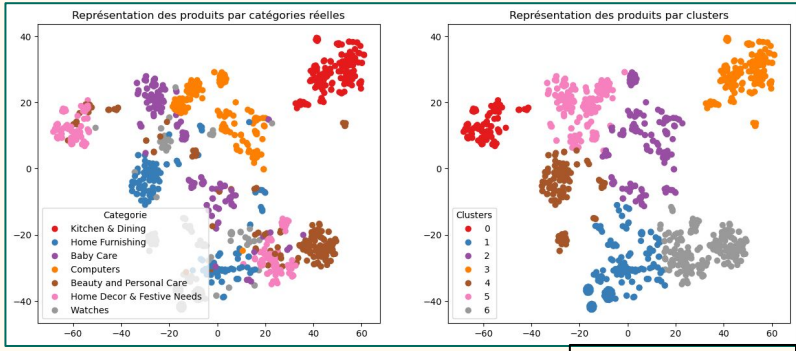


<u>la méthode</u>

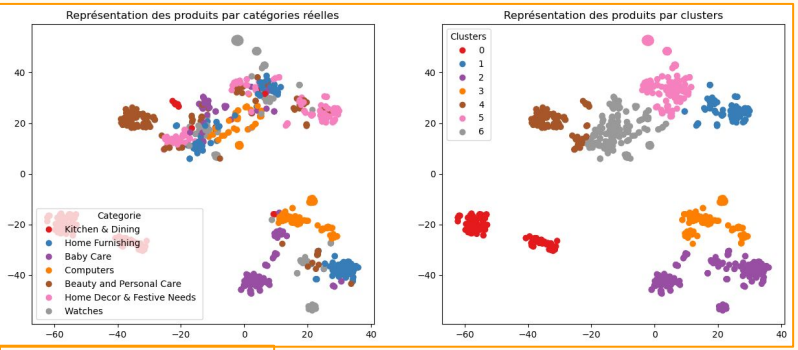
- preparation des donnees textes
- réduction de dimension, T-SNE (2 dimensions)
- clustering avec l'algorithme Kmeans
- ARI catégories réelles, Kmeans

PARTIE I: EXTRACTION DE FEATURES SUR LES DONNÉES TEXTES

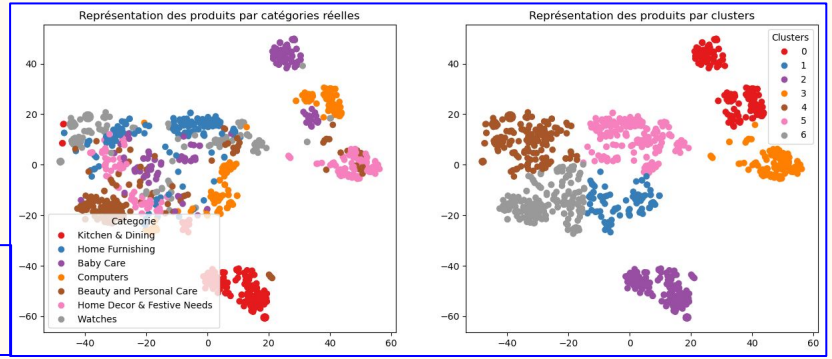
bag-of-words : word2vec, BERT et USE
algorithmes de deep learning



USE
ARI 0,43



word2vec
ARI 0,31

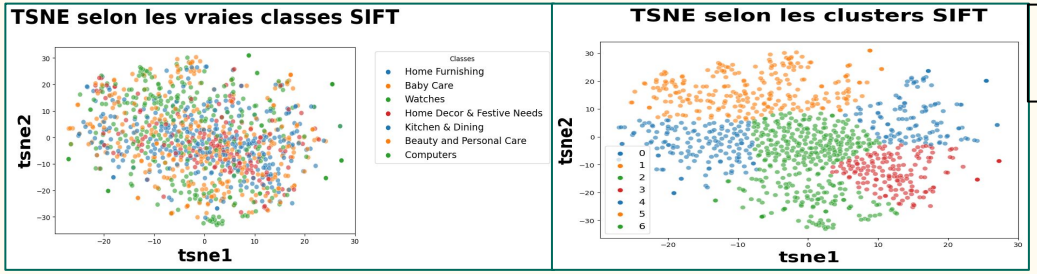


BERT
ARI 0,33

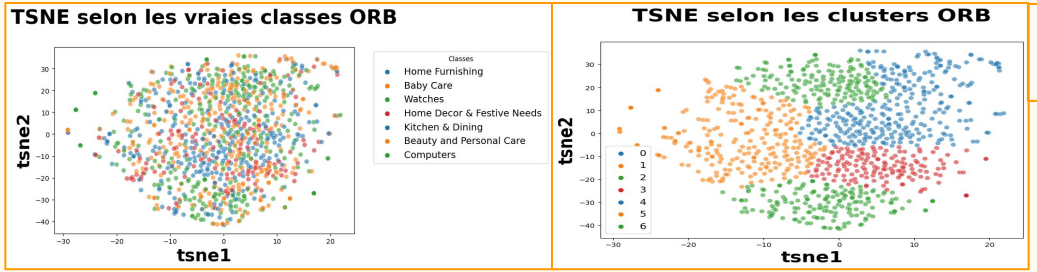
PARTIE II: EXTRACTION DE FEATURES SUR LES DONNÉES IMAGES

les différentes étapes:

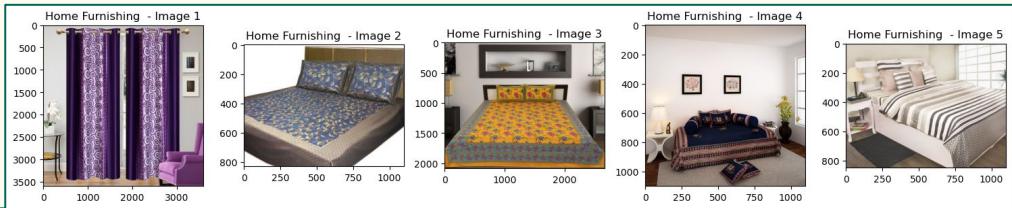
- ➢ création des descripteurs d'images
- ➢ création de clusters des descripteurs avec le mini batch Kmeans
- ➢ création des features des images
- ➢ réduction de dimension PCA, 99% de la variance
- ➢ réduction de dimension T-SNE, 2 composantes
- ➢ Kmeans avec 7 clusters
- ➢ score ARI entre les catégories réelles et celles issues du Kmeans
- ➢ représentation graphique



SIFT
ARI 0,046



ORB
ARI 0,026



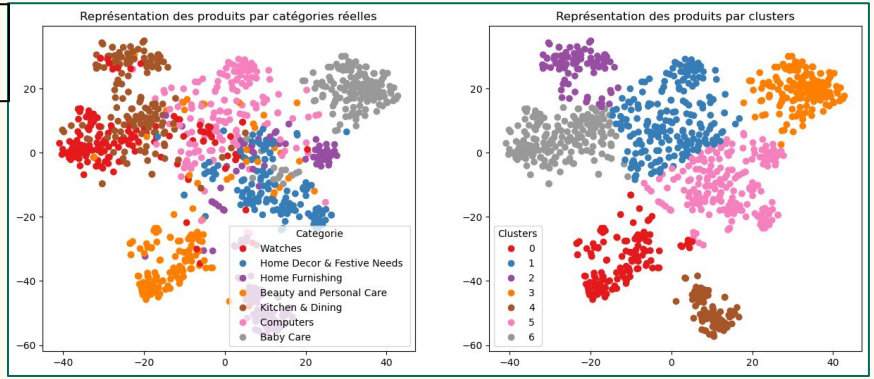
PARTIE II: EXTRACTION DE FEATURES SUR LES DONNÉES IMAGES

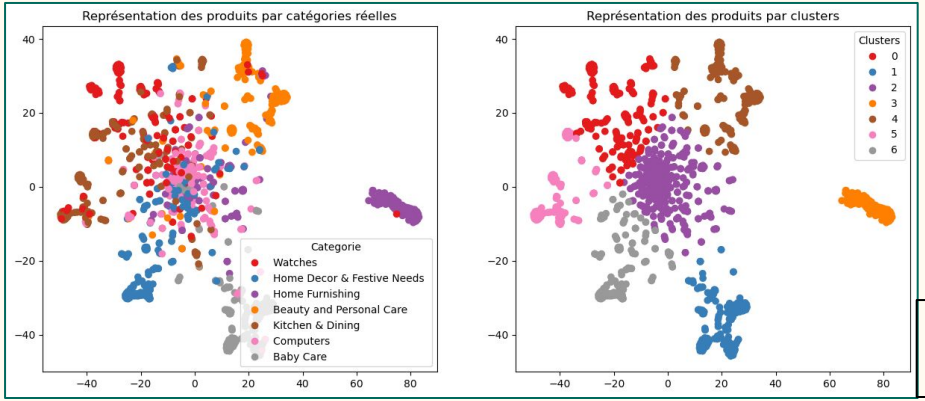
les différentes étapes:

- ➢ création des features des images
- ➢ réduction de dimension PCA, 99% de la variance
- ➢ réduction de dimension T-SNE, 2 composantes
- ➢ Kmeans avec 7 clusters
- ➢ score ARI entre les catégories réelles et celles issues du Kmeans
- ➢ représentation graphique

deux modèles pré-entraînés: VGG16 et ResNet50

VGG16
ARI 0,5





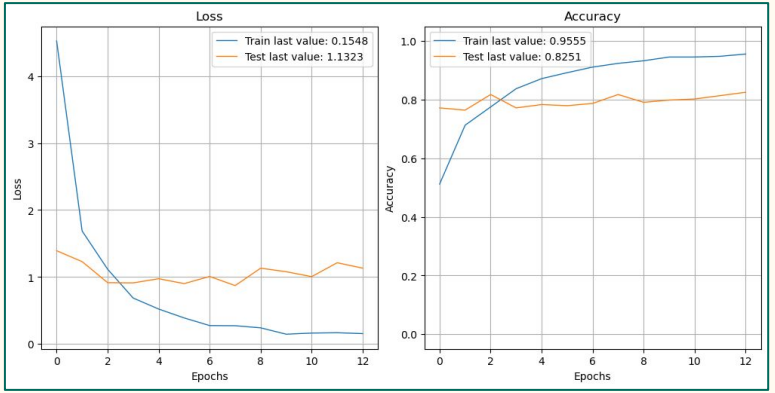
ResNet50
ARI 0,37

PARTIE III: CLASSIFICATION SUPERVISÉE SUR LES IMAGES, DATA AUGMENTATION, VGG 16 ET RESNET 50

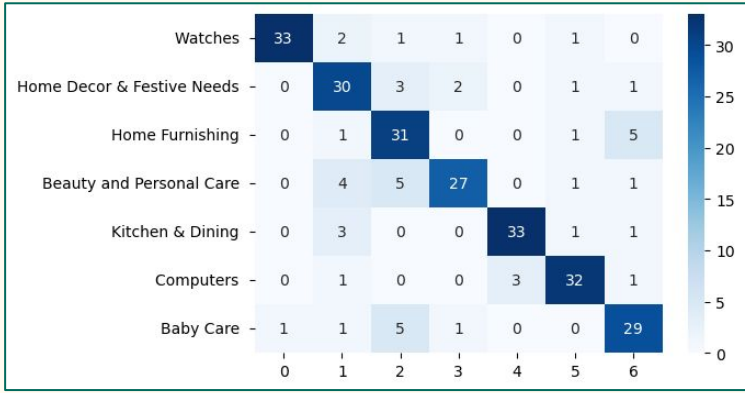
première approche: sans la data augmentation.
VGG16

la méthode

- ★ creation du modele
- ★ préparation initiale des images
- ★ détermination de X et y
- ★ split des données en deux parties: entraînement et validation, respectivement 75% et 25%
- ★ création et entraînement du modèle: early stopping



	precision	recall	f1-score	support
0	0.97	0.87	0.92	38
1	0.71	0.81	0.76	37
2	0.69	0.82	0.75	38
3	0.87	0.71	0.78	38
4	0.92	0.87	0.89	38
5	0.86	0.86	0.86	37
6	0.76	0.78	0.77	37
accuracy			0.82	263
macro avg	0.83	0.82	0.82	263
weighted avg	0.83	0.82	0.82	263

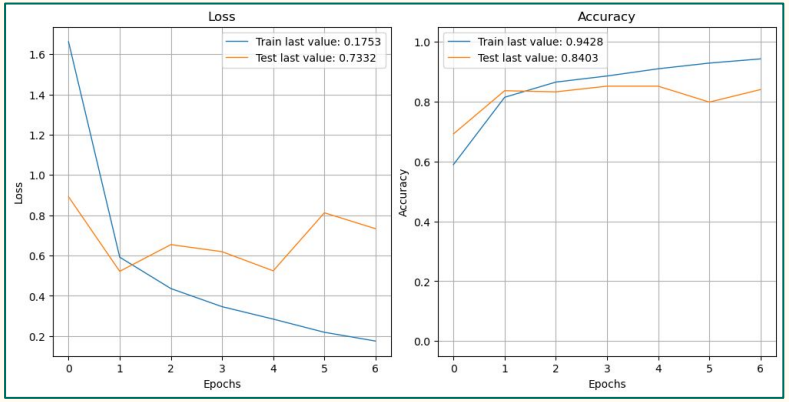


PARTIE III: CLASSIFICATION SUPERVISÉE SUR LES IMAGES, DATA AUGMENTATION, VGG 16 ET RESNET 50

première approche: sans la data augmentation.
RESNET 50

la méthode

- ★ creation du modele
- ★ préparation initiale des images
- ★ détermination de X et y
- ★ split des données en deux parties: entraînement et validation, respectivement 75% et 25%
- ★ création et entraînement du modèle: early stopping



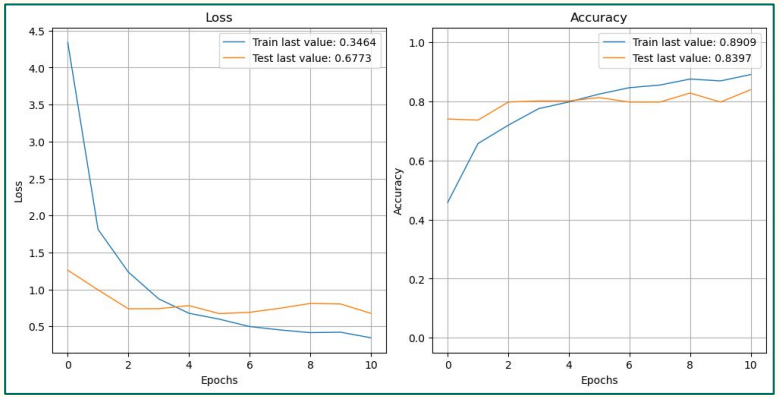
	precision	recall	f1-score	support
0	1.00	0.95	0.97	38
1	0.70	0.84	0.77	37
2	0.83	0.76	0.79	38
3	0.96	0.71	0.82	38
4	0.94	0.87	0.90	38
5	0.72	0.97	0.83	37
6	0.80	0.76	0.78	37
accuracy			0.84	263
macro avg	0.85	0.84	0.84	263
weighted avg	0.85	0.84	0.84	263

PARTIE III: CLASSIFICATION SUPERVISÉE SUR LES IMAGES, DATA AUGMENTATION, VGG 16 ET RESNET 50

deuxième approche: avec la data augmentation.
VGG16

la méthode

- ★ creation du modele
- ★ préparation initiale des images
- ★ détermination de X et y
- ★ split des données en deux parties: entraînement et validation, respectivement 75% et 25%
- ★ création et entraînement du modèle: early stopping



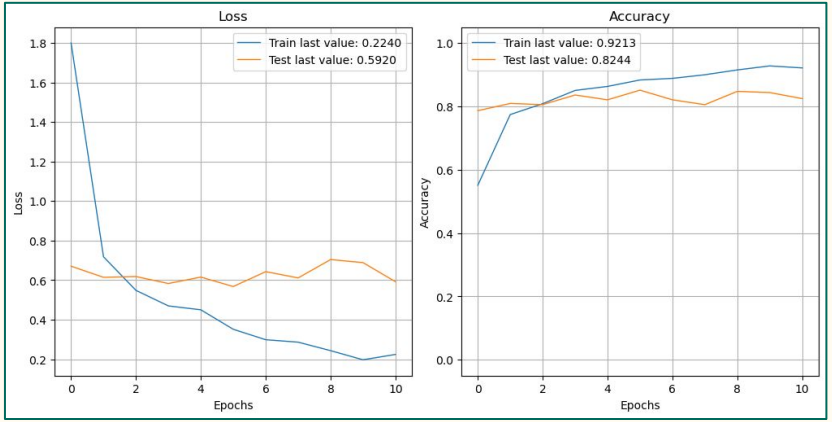
	precision	recall	f1-score	support
0	0.95	0.97	0.96	38
1	0.97	0.89	0.93	37
2	0.86	1.00	0.93	38
3	0.95	1.00	0.97	38
4	1.00	1.00	1.00	38
5	0.97	0.97	0.97	37
6	1.00	0.84	0.91	37
accuracy			0.95	263
macro avg	0.96	0.95	0.95	263
weighted avg	0.96	0.95	0.95	263

PARTIE III: CLASSIFICATION SUPERVISÉE SUR LES IMAGES, DATA AUGMENTATION, VGG 16 ET RESNET 50

deuxième approche: avec la data augmentation.
RESNET 50

la méthode

- ★ mélange des images initialement classées par classe
- ★ la data augmentation
- ★ creation du modele
- ★ préparation initiale des images
- ★ détermination de X et y
- ★ split des données en deux parties: entraînement et validation, respectivement 75% et 25%
- ★ création et entraînement du modèle: early stopping



	precision	recall	f1-score	support
0	1.00	0.95	0.97	38
1	0.92	0.95	0.93	37
2	0.86	1.00	0.93	38
3	0.95	0.95	0.95	38
4	1.00	1.00	1.00	38
5	0.97	0.97	0.97	37
6	0.97	0.84	0.90	37
accuracy			0.95	263
macro avg	0.95	0.95	0.95	263
weighted avg	0.95	0.95	0.95	263

PARTIE IV: CONCEPT D'UNE TECHNIQUE RÉCENTE ET SCRIPT PYTHON

troisième approche: avec une technique récente
Creation d'un modele de deep learning adaptée à notre situation.

ARI 0,49

la méthode

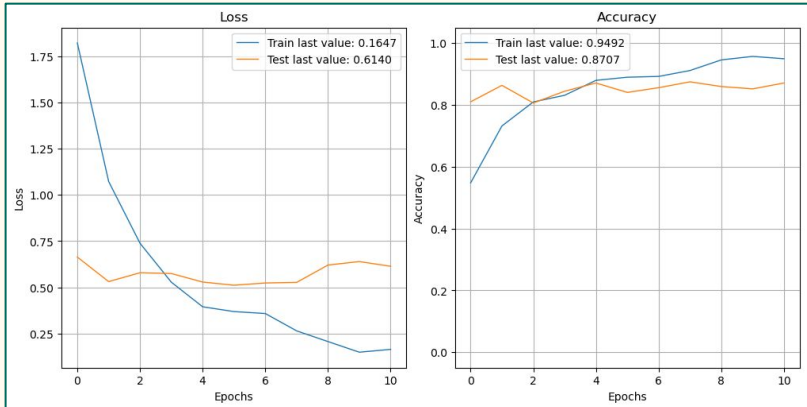
- ★ features texte (tif-idf)
- ★ features image (vgg16)
- ★ concaténation des features textes et images
- ★ faisabilité
- ★ split des données en deux parties: entraînement et validation, respectivement 75% et 25%
- ★ création et entraînement du nouveau modèle: early stopping

Model: "sequential"

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 1024)	8537088
dropout (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 512)	524800
dropout_1 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 256)	131328
dense_3 (Dense)	(None, 7)	1799

Total params: 9195015 (35.08 MB)
Trainable params: 9195015 (35.08 MB)
Non-trainable params: 0 (0.00 Byte)

ACCURACY 0,85



PARTIE IV: CONCEPT D'UNE TECHNIQUE RÉCENTE ET SCRIPT PYTHON

SCRIPT PYTHON + RGPD

la méthode

- ★ création de compte sur rapidapi
- ★ filtre sur les produits à base de champagne
- ★ récupération du code et exécution
- ★ création du data frame avec les features demandés
- ★ exportation puis lecture du data frame

bonus: les règles du RGPD

3 grands principes :

- garantir la conformité
- limiter les risques
- sensibiliser les opérationnels

les dix règles d'or

auto documentation	documenter ses recherches
finalité	déclarer un objectif pour ce traitement
base légale	avoir une raison juridique
durée de conservation	cohérente avec l'objectif
la protection des la conception	du RGPD dans ses cahiers de charge
l'obligation de sécurité	le P du RGPD
la minimisation	seulement ce qui est nécessaire
la transparence	jouer carte sur table
les données à risque	traitement interdit sauf si…
les droits des personnes	afficher et faciliter