Projet numéro 6 :

Etudiez la faisabilité d'un moteur de classification d'articles

Thomas Zilliox

Mai 2025

Présentation du Projet

Contexte

- Sur cette place de marché anglophone, des vendeurs proposent des articles à des acheteurs en postant une photo et une description.
- **il devient nécessaire d'automatiser** l'attribution de la catégorie des articles, pour rendre l'expérience utilisateur des vendeurs plus fluide.

Mission

- Étudier la faisabilité d'un **moteur de classification** des articles en différentes catégories, à partir du texte (en anglais) et de l'image.

Sommaire

Résumé des étapes du projet :

1) Prétraitement et faisabilité du projet : Partie Texte

2) Prétraitement et faisabilité du projet : Partie Image

3) Apprentissage supervisé

4) Test de l'API

Première étape : Prétraitement et faisabilité du projet

Partie Texte :

Trois sous- parties :

- Prétraitement du texte (colonne *description*) pour 7 catégories

- Faisabilité et résultats par les méthodes classiques :
  - Bag-Of-Words
  - Tf-idf

- Faisabilité et résultats par les méthodes de Word-Embedding :
  - Word2Vec
  - BERT : Tenserflow Hub et Hugging Face Transformers
  - USE

Première étape : Prétraitement et faisabilité du projet

Partie Texte : Prétraitement du texte (colonne *description*)

→ Nettoyage de base du texte

- Mise en minuscule
- Retrait de la ponctuation
- Suppression des chiffres si non pertinents

→ Tokenisation

- Diviser le texte en unités : **mots**, **tokens** ou **phrases**.
- Avec les modèles modernes comme **BERT**, on utilise des tokenizers spéciaux (bert-base-uncased par exemple)

→ Stopwords et Lemmatisation

- Supprimer les mots fréquents sans valeur (ex : *le*, *et*, *de*)
- Réduire un mot à sa forme de base ("mangé" → "manger")

Première étape : Prétraitement et faisabilité du projet

Partie Texte : Méthodes dites classiques

Bag of Words (BoW) :

- Représente un texte par un **compte brut de mots**.
- Chaque texte devient un vecteur où chaque dimension correspond à un mot du vocabulaire.

TF-IDF (Term Frequency-Inverse Document Frequency) :

- Comme BoW, mais les mots sont **pondérés**.
- Mots fréquents dans un document mais rares dans la collection ont un poids plus fort.

Première étape : Prétraitement et faisabilité du projet

Bag of Words (BoW) :

Bag of Words Model explanation

Corpus

↓

Tokenize

↓

Compte la fréquence des mots

↓

Encode les données

Example

Corpus:

The dog is happy. The child makes the dog happy. The dog makes the child happy

↓

Tokenization:

D1: [The] [dog] [is] [happy]
D2: [The] [child] [makes] [the] [dog] [happy]
D3: [The] [dog] [makes] [the] [child] [happy]

↓

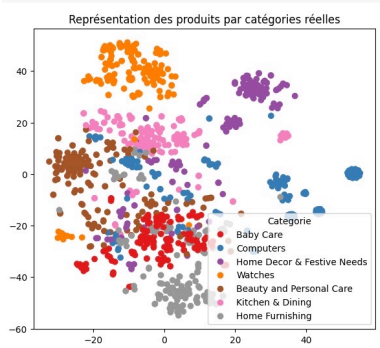
Documents	Counting word frequencies
D1	the: 1, dog: 1, is: 1, happy: 1
D2	the: 2, dog: 1, makes: 1, child: 1, happy: 1
D3	the: 2, child: 1, makes: 1, dog: 1, happy: 1

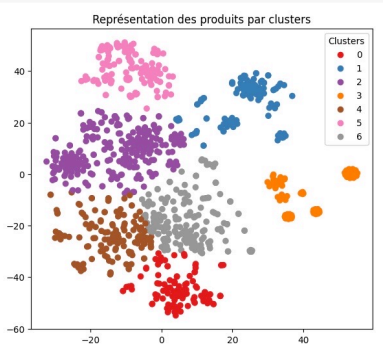
↓

Encode	child	dog	happy	is	makes	the	BoW Vector representations
D1	0	1	1	1	0	1	[0,1,1,1,0,1]
D2	1	1	1	0	1	2	[1,1,1,0,1,2]
D3	1	1	1	0	1	2	[1,1,1,0,1,2]

ARI (Adjusted Rand Index)

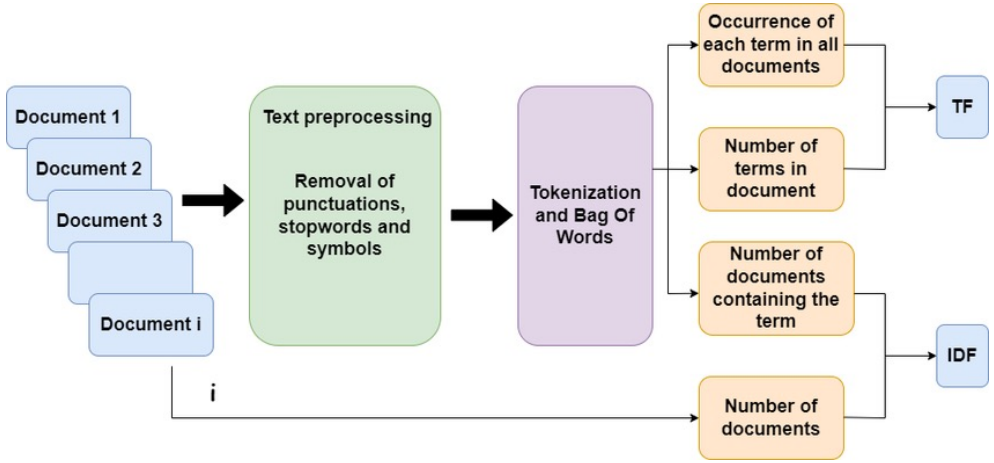
0,3675





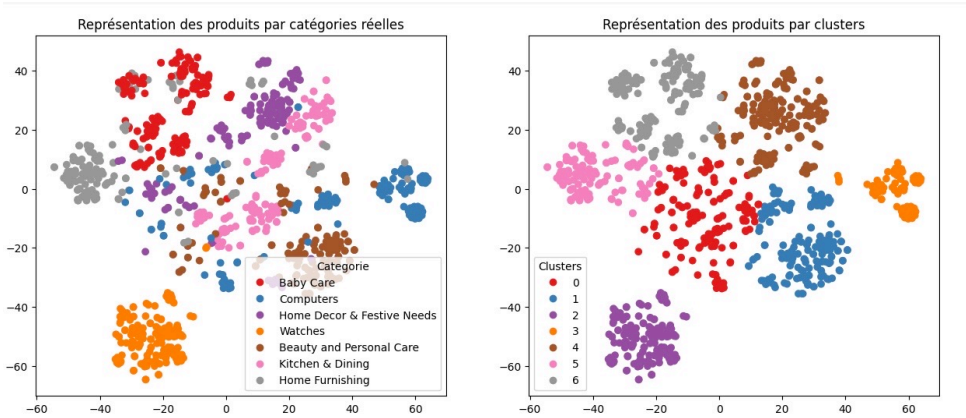
Première étape : Prétraitement et faisabilité du projet

TF-IDF (Term Frequency-Inverse Document Frequency) :



ARI (Ajdusted Rand Index)

0,4875



Première étape : Prétraitement et faisabilité du projet

Partie Texte : Méthodes de Word-Embedding

→ Word2Vec :

- Représente chaque **mot** par un **vecteur dense et continu**, appris via un réseau de neurones.
- Ces vecteurs capturent le **sens** des mots, leur **proximité sémantique** et les analogies.

→ BERT (Bidirectional Encoder Representations from Transformers) :

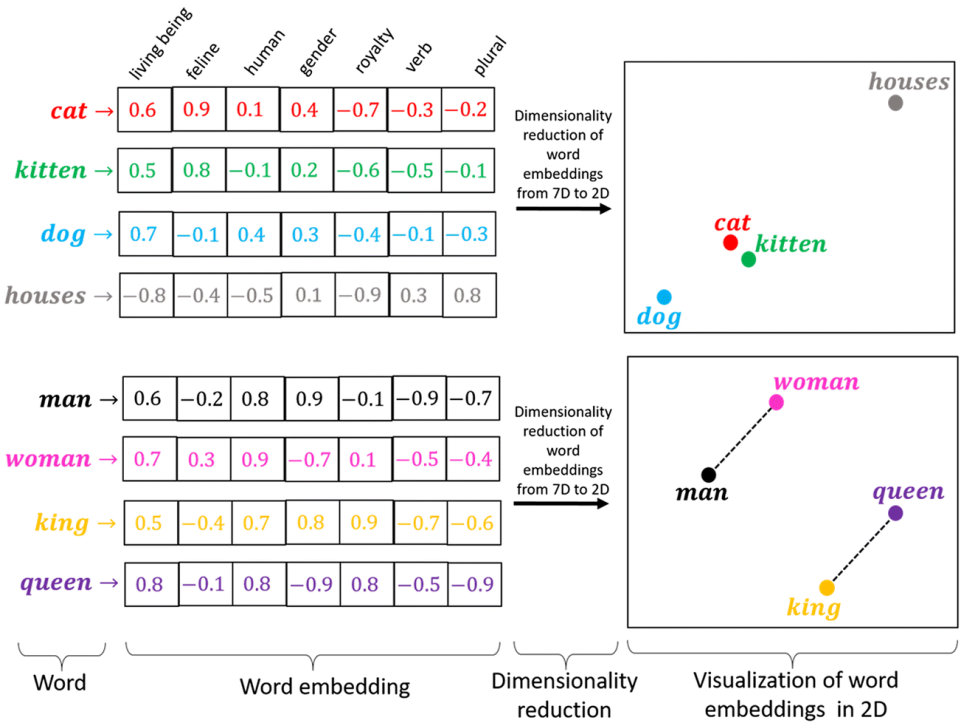
- Compréhension **fine du contexte** (grâce à la bidirectionnalité).
- Le modèle apprend à **prédire les mots masqués** à partir du contexte complet.

→ USE : Universal Sentence Encoder

- Transforme **des phrases entières** (et pas seulement des mots) en **vecteurs numériques** (aussi appelés **embeddings**).
- **Encode directement la signification de la phrase.**

Première étape : Prétraitement et faisabilité du projet

Word2Vec :



- Transformation des mots en vecteurs numériques.
- Obtenir une représentation vectorielle des mots
- But : Le modèle doit comprendre le langage brut.

⇒ Résultats non-concluants.

Première étape : Prétraitement et faisabilité du projet

BERT :

Deux choix utilisés :

1. BERT Tensorflow Hub
2. BERT Hugging Face Transformers

	TensorFlow Hub	**Hugging Face Transformers**
🎛️ Configuration	Simple et directe	Très flexible et fine-grainée
🧠 Accès aux couches	Limité	Complet
📚 Variété de modèles	Faible (BERT principalement)	Énorme (des centaines de modèles)

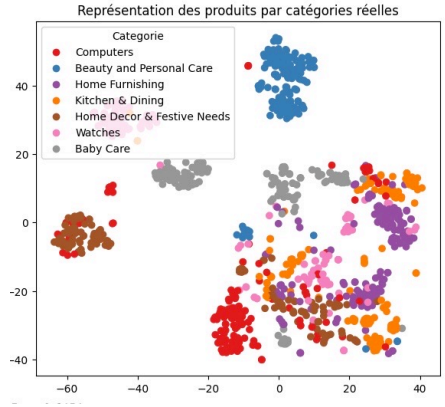
Première étape : Prétraitement et faisabilité du projet

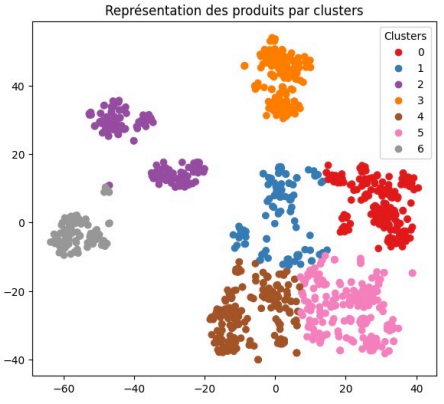
Résultats :

BERT :

ARI

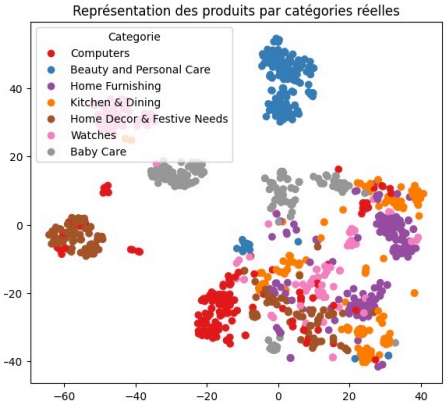
1 BERT Tensorflow Hub

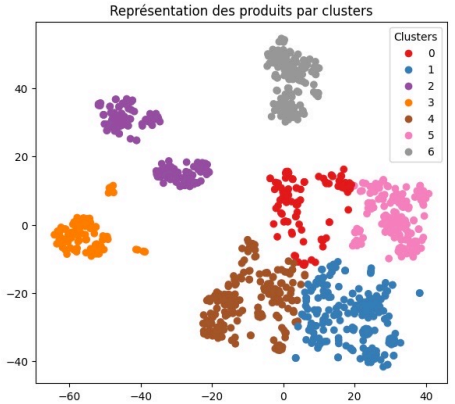




0,323

2 BERT Hugging Face Transformers

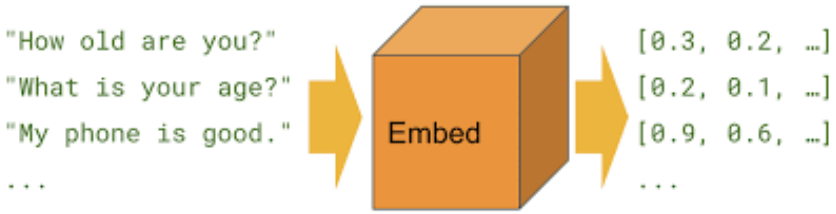


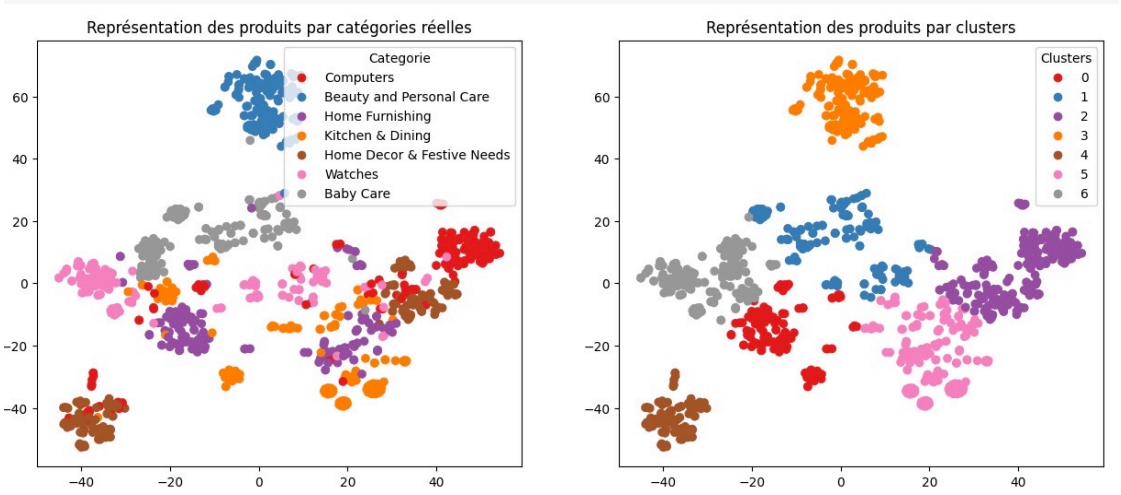


0,3154

Première étape : Prétraitement et faisabilité du projet

USE :





ARI

0,4723

Modèle le plus performant.

Première étape : Prétraitement et faisabilité du projet

Partie Texte : Conclusions

Méthodes classiques : Simples mais pas les plus efficaces

Méthodes d'Embeddings : USE semble être le plus efficace

Possibilités d'amélioration :

- Essayer d'autres modèles de classification
- Améliorer le fine tuning
- Améliorer le prétraitement des textes

Deuxième étape : Prétraitement et faisabilité du projet

Partie Image :

Deux sous- parties :

- Prétraitement des images : Couleurs, égalisation, descripteurs et standardisation

- Faisabilité et résultats par les images des produits :
  - Générateur de descripteurs (SIFT)
  - Réseaux de neurones (CNN TransferLearning)

Deuxième étape : Prétraitement et faisabilité du projet

Partie Image : Prétraitement des images

Transformation des couleurs en noir et blanc

- Réduction de la complexité
- Moins de bruit visuel
- Gain en temps d'entraînement

Égalisation de l'histogramme de l'image

- Meilleure visibilité des détails
- Amélioration des performances des modèles
- Préparation standardisée des images

Génération de descripteurs

- Extraire les éléments pertinents d'une image en forme
- Réduit la complexité
- Facilite la comparaison d'images

Standardisation de la résolution

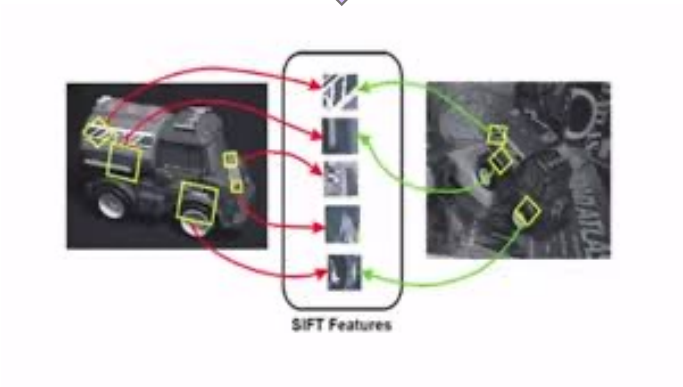
- Assure une cohérence en taille entre toutes les entrées du modèle
- Accélère l'entraînement en réduisant la consommation de mémoire

Deuxième étape : Prétraitement et faisabilité du projet

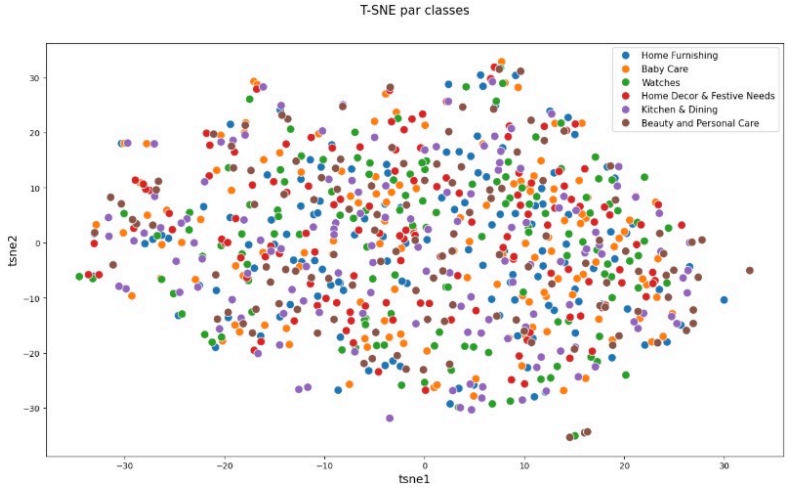
Partie Image : Méthode SIFT (Scale-Invariant Feature Transform)

Détecte et décrit les points d'intérêt (keypoints) au sein des images, efficace pour détecter des points caractéristiques robustes et distinctifs au sein des images.

Trouve les keypoints



ARI (Ajusted Rand Index) → 0,0671



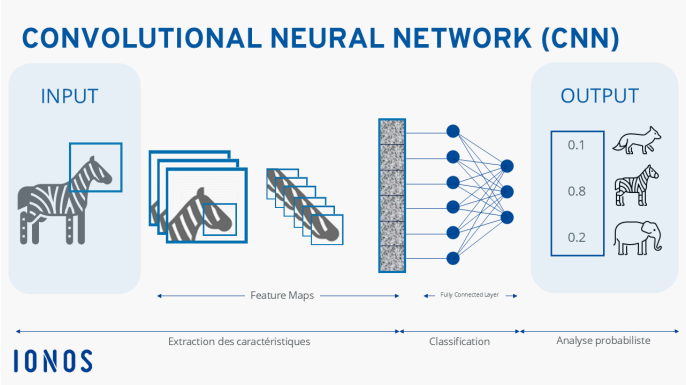
Deuxième étape : Prétraitement et faisabilité du projet

Partie Image : CNN (Conventional Neural Network)

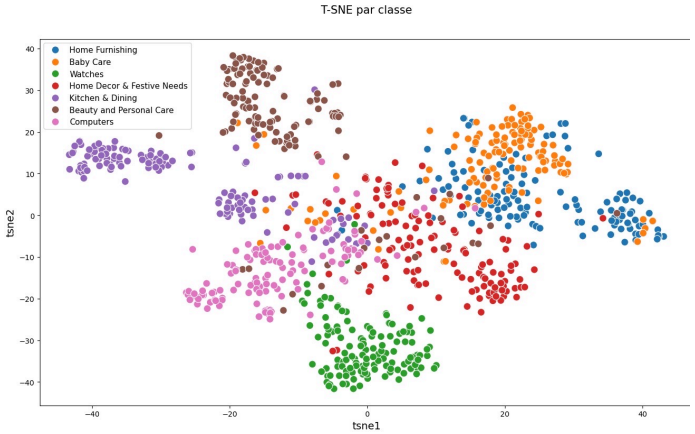
Apprentissage Non - Supervisée

Type de réseau neuronal artificiel utilisé principalement pour la reconnaissance et le traitement d'images, grâce à sa capacité à identifier des motifs

Identifie les motifs à l'aide de plusieurs couches



ARI (Ajusted Rand Index) → 0,4894



Deuxième étape : Prétraitement et faisabilité du projet

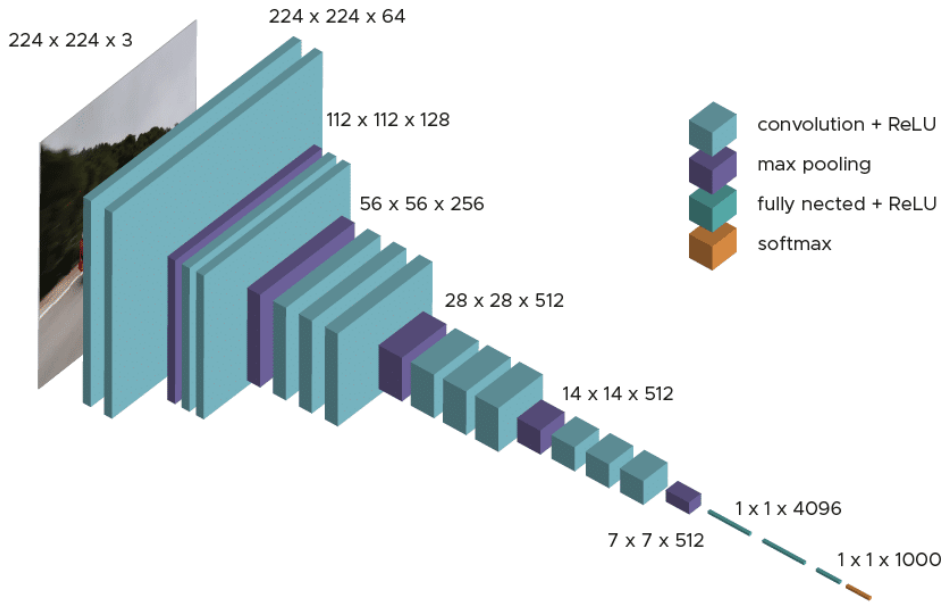
Partie Image : CNN (Conventional Neural Network) → Apprentissage Supervisée

→ 4 modèles testés :

Modèle	Description	Validation Accuracy	Test Accuracy
VGG16 Classification supervisée simplifiée	16 couches, simple, adapté classification image.	0.84	0.82
VGG16 : Image generator avec augmentation des données	VGG16 avec génération d'images augmentées automatiquement.	0.8136	0.8213
VGG19 avec augmentation intégrée	19 couches, plus puissant avec données augmentées intégrées.	0.8475	0.8390
RESNET 50 avec augmentation intégrée	50 couches, résout surapprentissage via connexions résiduelles	0.8644	0.8517

Deuxième étape : Prétraitement et faisabilité du projet

Partie Image : CNN (Conventional Neural Network) → Exemple du VGG16



1. Convolution + ReLU : extrait des motifs visuels (bords, textures) dans l'image + garde les valeurs positives pour activer les neurones importants.

2. Max Pooling : réduit la taille des images en gardant les informations les plus fortes, ce qui simplifie les calculs et évite le surapprentissage.

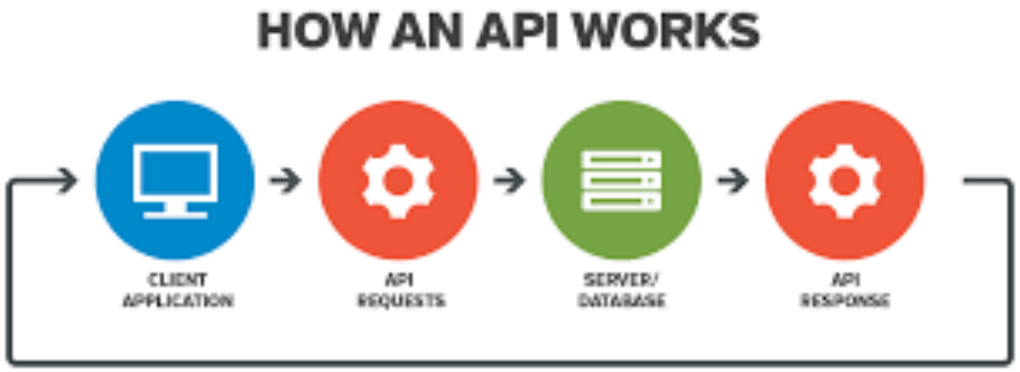
3. Fully Connected + ReLU : connectent tous les neurones entre eux pour combiner les informations extraites et prendre des décisions.

4. Softmax : transforme les scores finaux en probabilités pour chaque classe, afin de choisir la prédiction finale.

Troisième étape : Mise en place de l'API

API (Application Programming Interface)

Programme permettant à deux applications distinctes de communiquer entre elles.



Troisième étape : Mise en place de l'API

Démarche :

- Requête sur la catégorie Champagne
- Import des données
- Transformation en dataframe
- Filtrage en s'assurant du respect du RGPD
- Extraction au format csv

API Edamam :

Interface de programmation d'application qui permet aux développeurs d'accéder aux données nutritionnelles, des recettes et des informations alimentaires des produits disponibles sur le site edamam.com

RGPD :

Utiliser les données nécessaires exclusivement à notre projet.

Conclusion

La classification des produits à l'aide de leurs descriptions ainsi que de leurs images est donc possible. Les deux modèles à privilégier sont :

- Le modèle BERT pour la classification textuelle
- Le modèle ResNet50 pour la classification d'image

La méthode optimale afin d'obtenir le modèle le plus efficace serait de fusionner les deux algorithmes en un seul algorithme, ce que l'on appelle un algorithme d'apprentissage multimodal.

La mise en place d'une API afin de se connecter directement à OpenFoodfact est également possible, afin d'y recueillir les différentes données.

Merci pour votre attention.