

Classifiez automatiquement des biens de consommation



par Ethan VUILLEMIN





Contexte & Objectifs

Contexte

Le but est de **tester la faisabilité** de classifier automatiquement des biens en fonction de leurs **descriptions textuelles** et de leurs **images**.

Objectif

- **Montrer la faisabilité** d'une classification supervisé en ce basant sur des données textuelles ou des images
- Si la faisabilité est prouvé faire un **modèle de classification supervisé** d'image

Règlement général sur la protection des données (RGPD)

Licéité

Fondé sur une base légale

Finalité

Collectées pour un but précis et légitime

Minimisation

Seules les données strictement nécessaires

Exactitude

Les données doivent être exactes et à jour

Limitation de la conservation

Ne doivent pas être conservées plus longtemps que nécessaire

Intégrité et confidentialité

Protégées contre les accès non autorisés ou les fuites

Transparence

Les personnes concernées doivent être informées

Droits des personnes concernées

Les individus ont des droits sur leurs données (accès, rectification, effacement, opposition)

Étude de faisabilité d'un moteur de classification automatique

Classification d'image

Mettre en œuvre des algorithmes pour l'**extraction de caractéristiques**. Puis comparer les données transformées avec un clustering et de la visualisation pour prouver la faisabilité du modèle

Classification de Texte

Appliquer des techniques telles que la **tokenisation**, l'élimination des **stop words**, et la **lemmatisation** pour réduire les mots à leur forme de base. Ensuite, envisagez des méthodes de **vectorisation** ou **d'embeddings** pour représenter le texte de manière numérique. Puis comparer les données de base avec un clustering pour prouver la faisabilité en fonction de la visualisation et du score ARI



Creation des labels

Creation des labels grace a la colonne **product_category_tree**

Creation d'un regex d'extraction de la main category



\[".*?\]\s?>>

1. Faisabilité de la classification d'image

Lecture du dataframe et des images

Extraction et lecture des données fournies.

1

Features extraction

Extraction des caractéristiques (SIFT, ORB, SURF, VGG16, USE).

2

Preprocessing des images

Normalisation et tests de transformations de base.

3

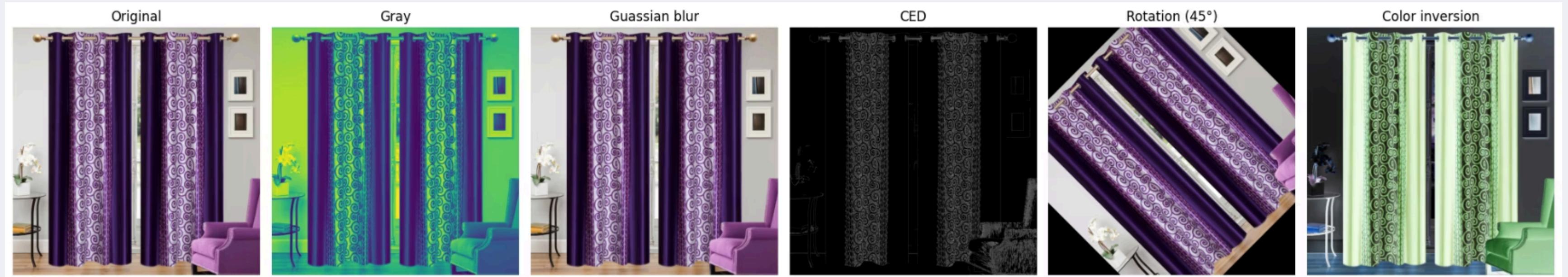
Analyse Graphique et Similarité

Analyse 2D/3D (TSNE), Clustering (Kmeans) et Adjusted Rand Score.

4



Lecture et Processing des images



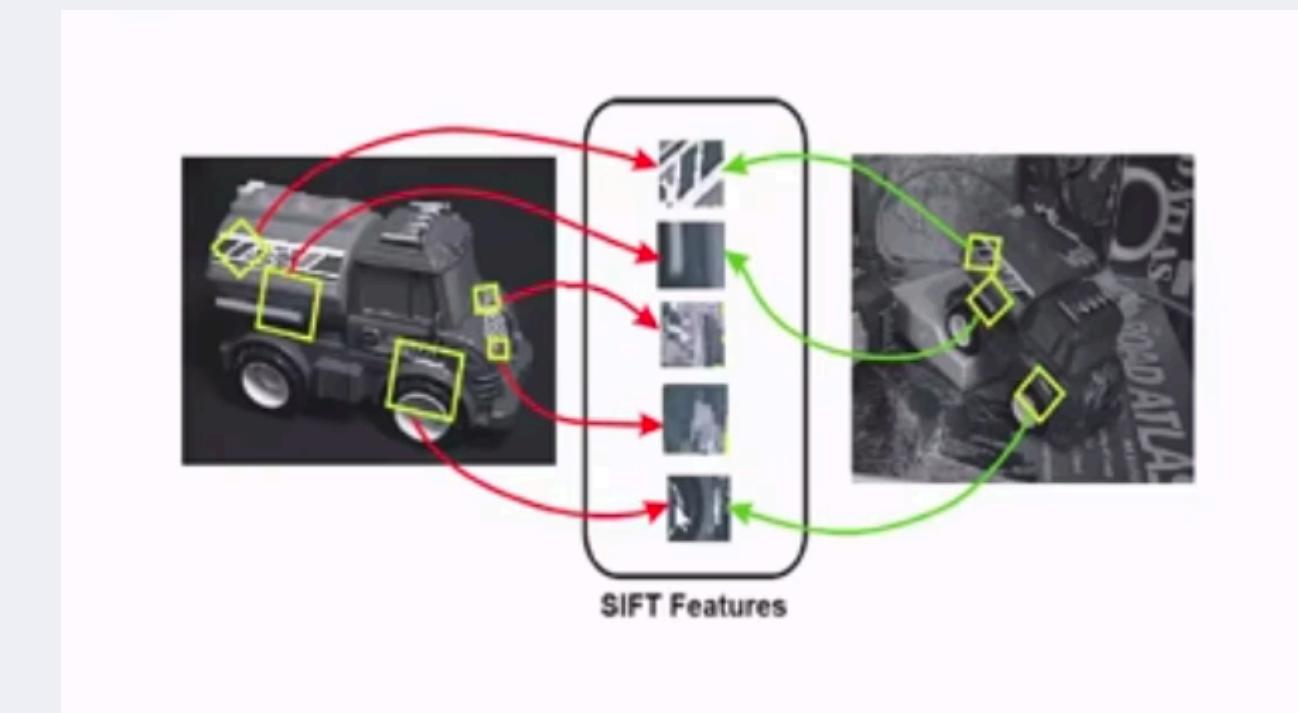
Features Extraction

SIFT, ORB, VGG16



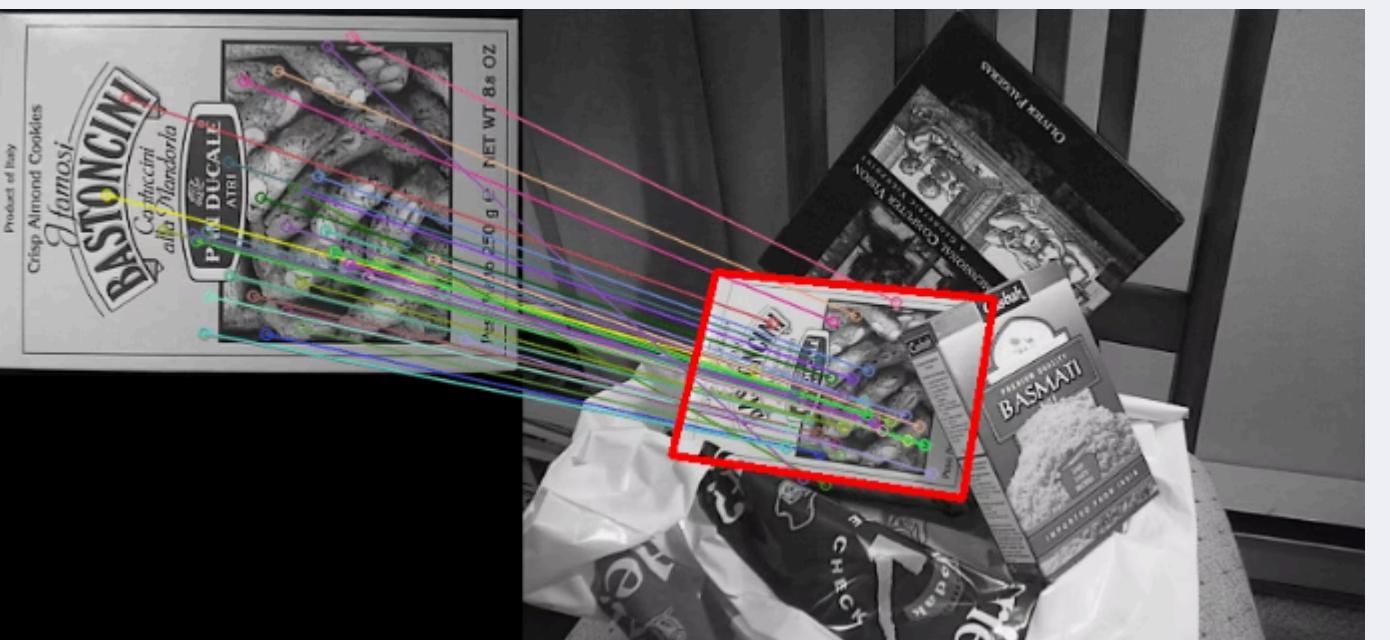
SIFT

SIFT (**Scale-Invariant Feature Transform**) est un algorithme qui **déetecte des points clés** dans une image. Ces points restent identifiables même si l'image **change de taille, de rotation ou de luminosité**. SIFT décrit chaque point de manière unique, permettant de reconnaître des objets ou des scènes dans différentes conditions. C'est utile pour la **reconnaissance d'images** et la **comparaison de scènes**.



ORB

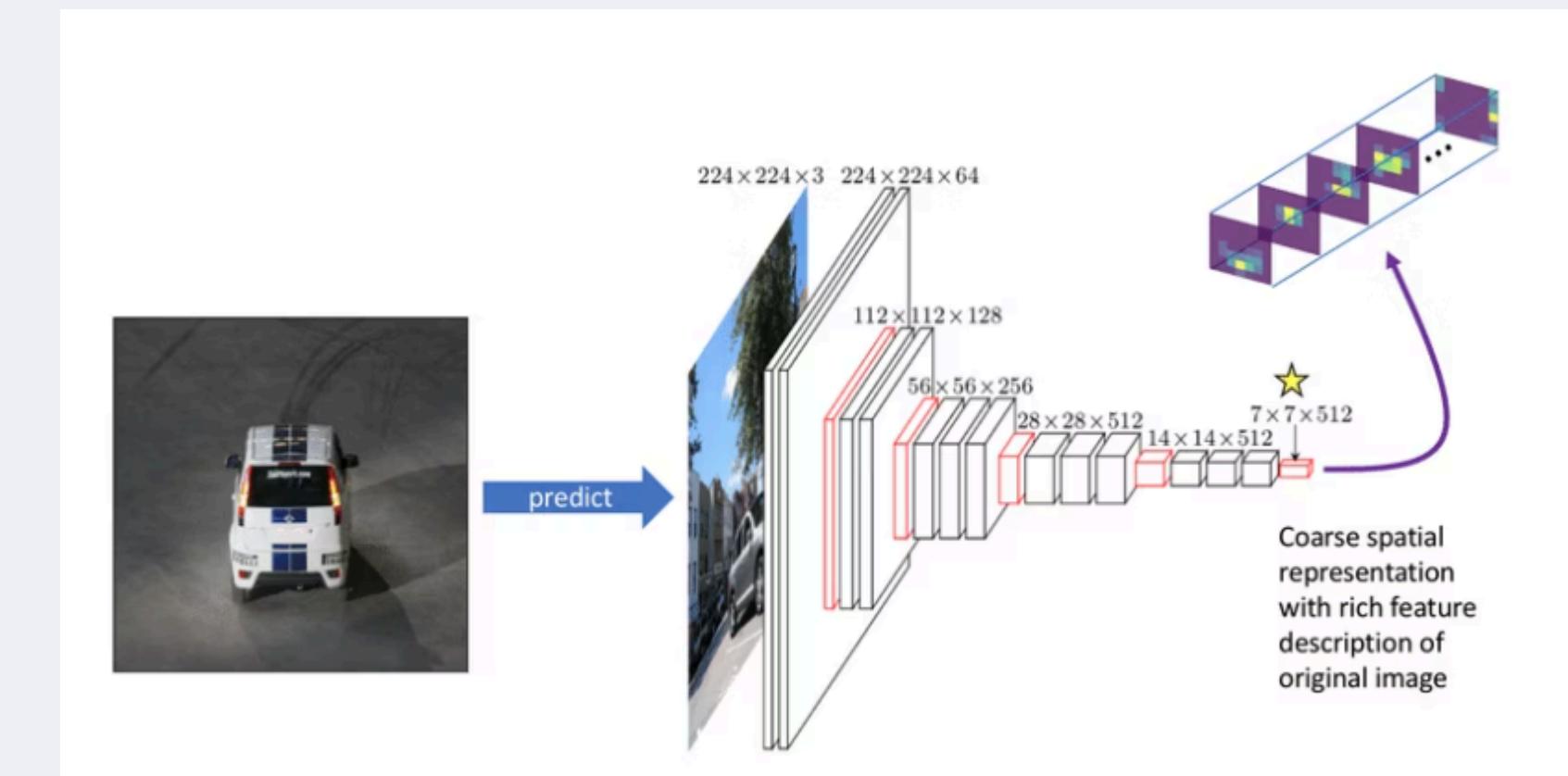
ORB (Oriented FAST and Rotated BRIEF) est un algorithme qui **déetecte et décrit des points clés dans une image**. Il est rapide et efficace, identifiant des points même si l'image change de rotation ou d'échelle. ORB décrit chaque point de manière unique, permettant de comparer et reconnaître des objets ou des scènes dans différentes conditions. C'est utile pour la reconnaissance d'images en temps réel.



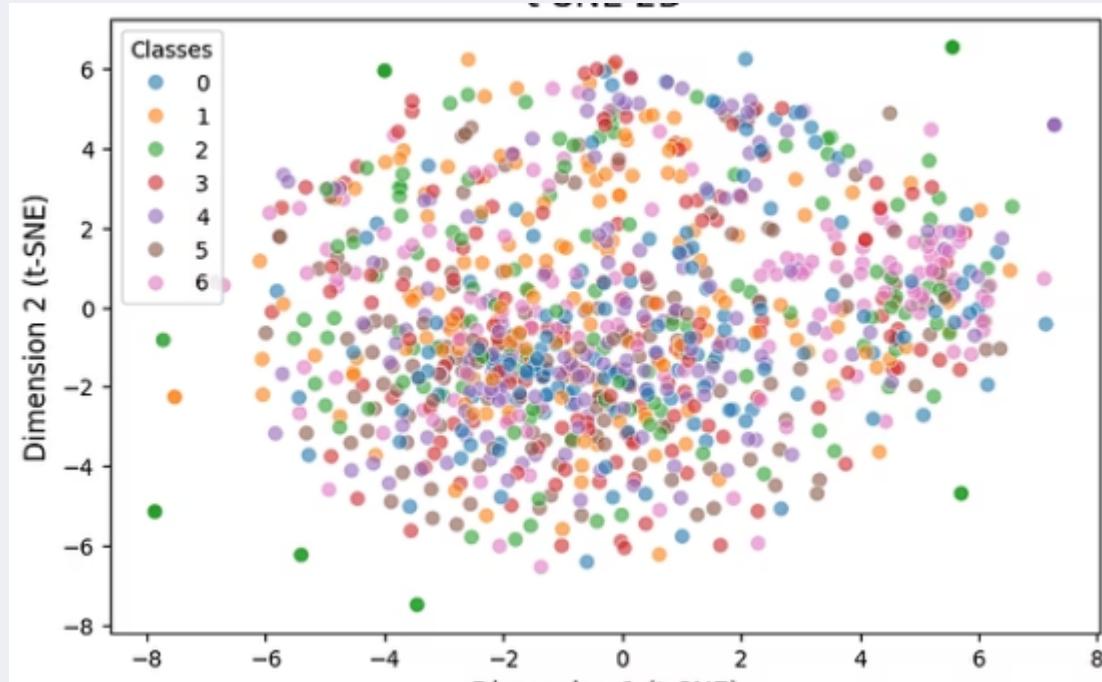


VGG16

VGG16 est un **réseau de neurones convolutif** utilisé pour l'extraction de caractéristiques dans les images. Il **analyse une image en plusieurs couches**, capturant des détails de plus en plus complexes. VGG16 transforme l'image en une **représentation compacte et riche en informations**, utile pour la classification et la reconnaissance d'objets.

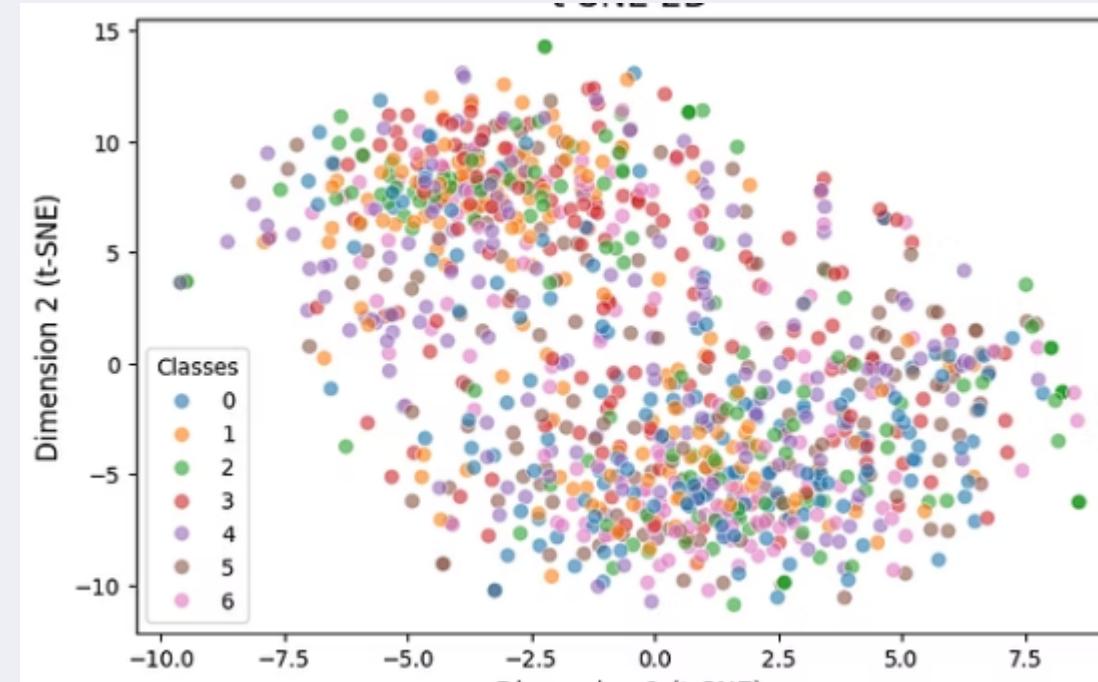


Analyse Graphique 2D / 3D



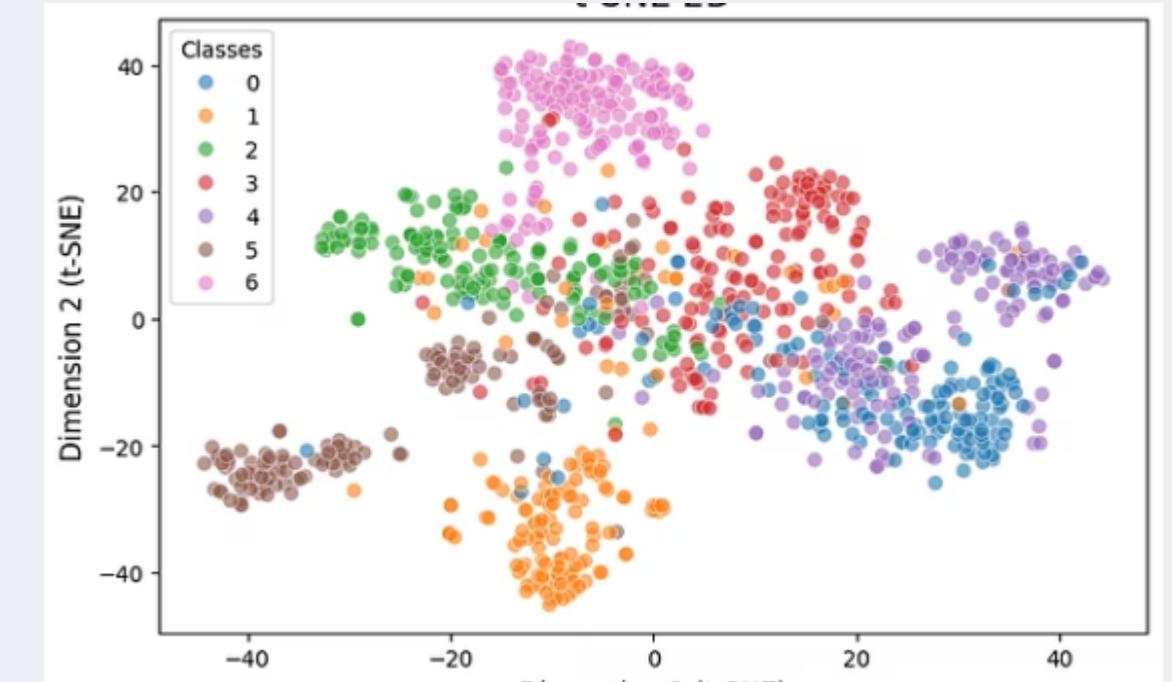
SIFT

0.0314



ORB

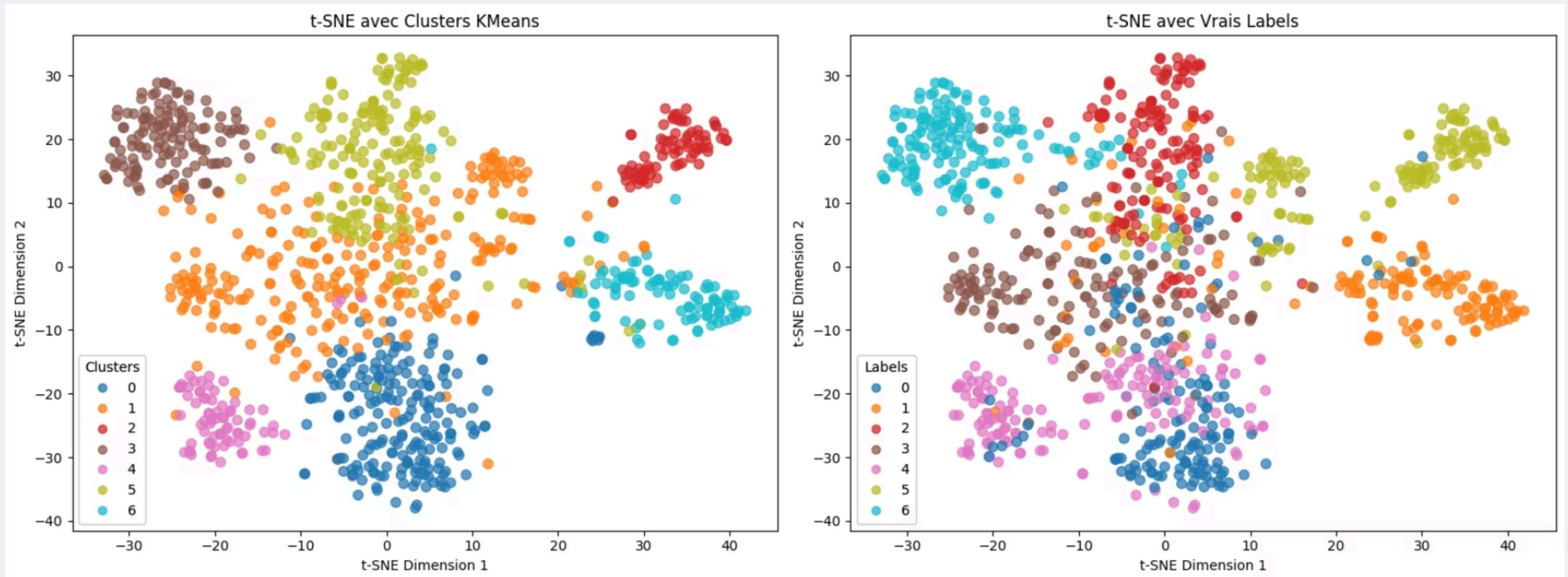
0.0279



VGG16

0.4733

Clustering Kmeans VGG16 vs Real labels



2. Faisabilité de la classification de texte

Étapes clés dans la classification de texte :

Lecture du dataframe et creation des labels

Extraire et lire les données fournies et création des labels

1

Features extraction

CountVectoriser, TF/IDF, Word2vec, USE,
SentenceTransformers

2

Preprocessing

Tokenisation, lemmatisation, stop word ...

3

Analyse Graphique et Similarité des clusters avec les données initiale

Analyse graphique 2D et 3D avec TSNE, Clustering Kmeans et Adjusted Rand Score

4

Preprocessing

- 1. Mise en minuscules
- 1. Lemmatization
- 1. Suppression de la ponctuation
- 1. Tokenization
- 1. Suppression des stopwords

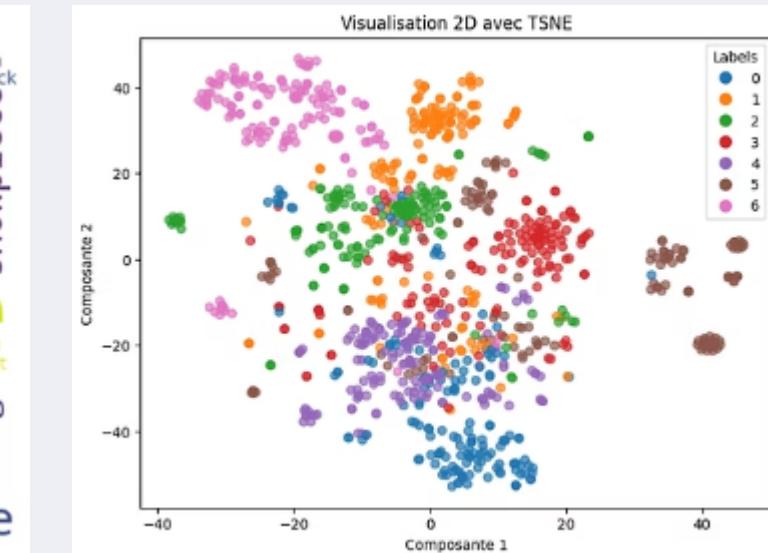
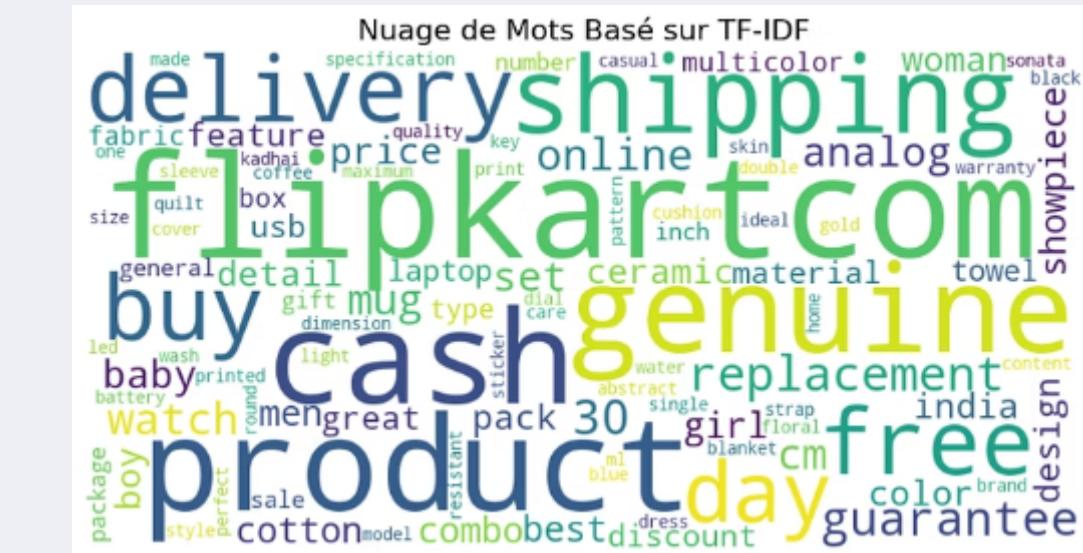


CountVectorizer

Le CountVectorizer est un outil qui transforme du texte en une matrice de nombres. Il **compte le nombre d'occurrences de chaque mot** dans un ensemble de documents. Cela permet de convertir des données textuelles en données numériques pour l'analyse.

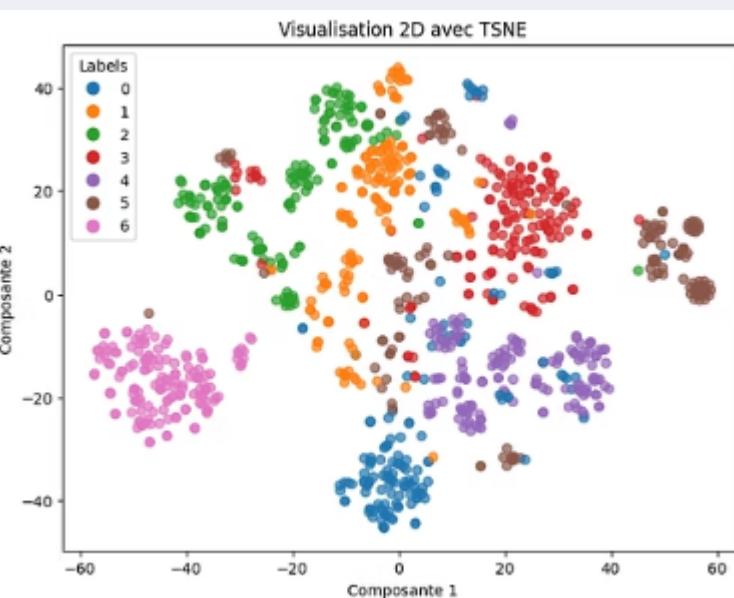
```
Data = ['The', 'quick', 'brown', 'fox', 'jumps', 'over', 'the', 'lazy', 'dog']
```

	The	quick	brown	fox	jumps	over	lazy	dog
Data	2	1	1	1	1	1	1	1



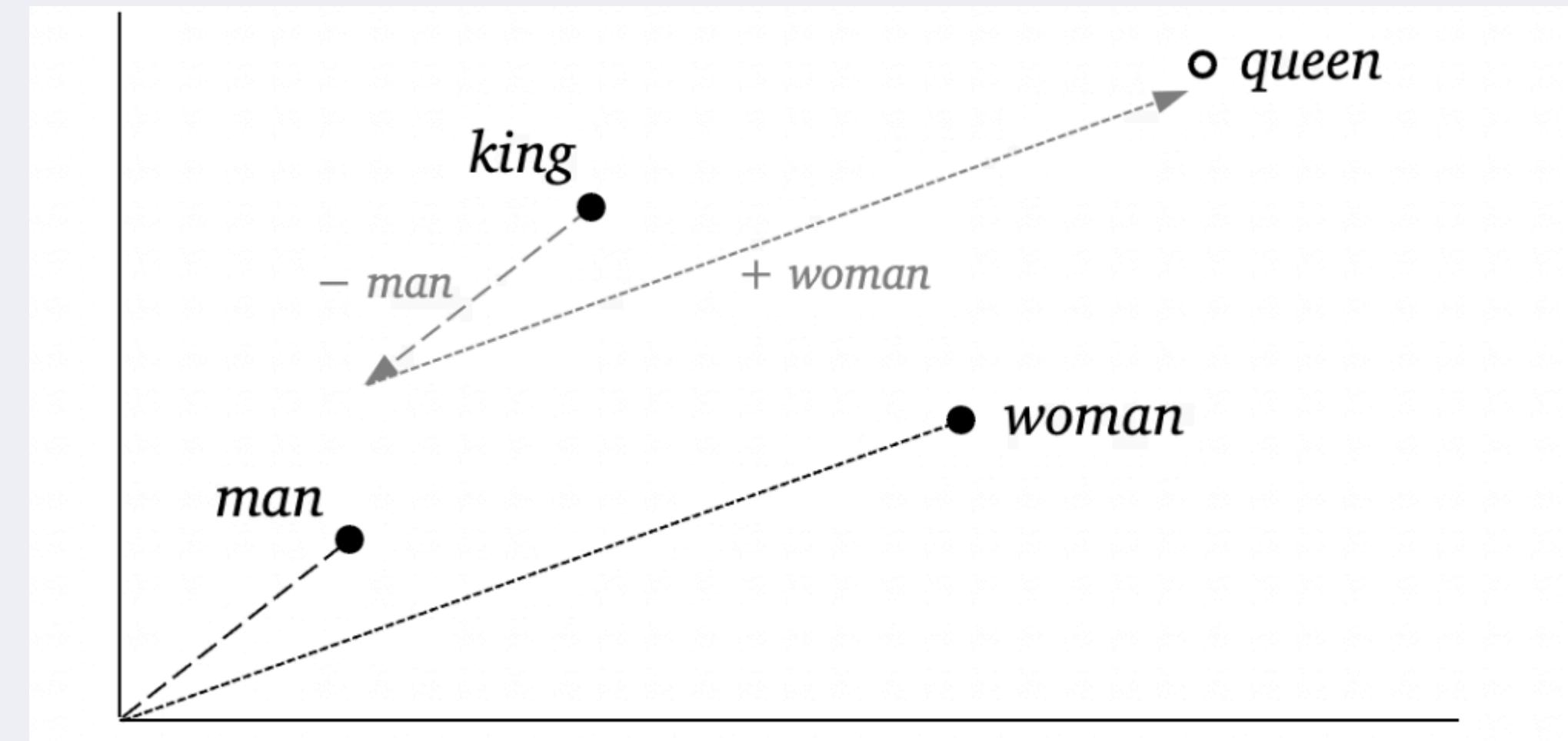
TF/IDF

Le TF-IDF (Term Frequency-Inverse Document Frequency) mesure **l'importance d'un mot dans un texte en fonction de sa fréquence** dans ce texte et dans l'ensemble des documents. Il aide à identifier les mots clés significatifs.



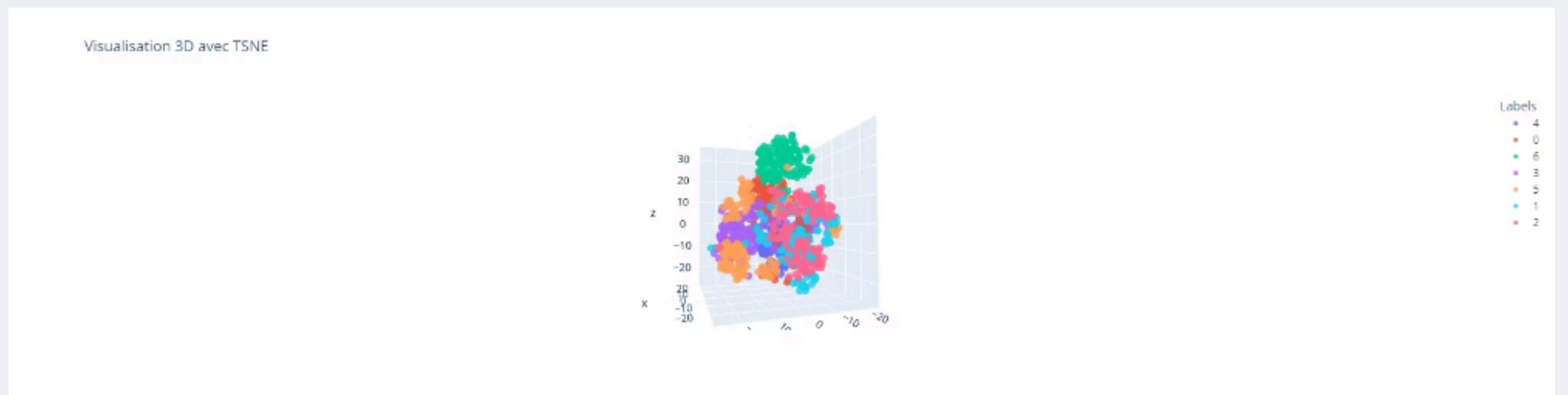
WORD2VEC

Word2Vec transforme les mots en vecteurs numériques en fonction de leur contexte. Il capture les relations sémantiques entre les mots, permettant de trouver des mots similaires ou de comprendre le sens des phrases.



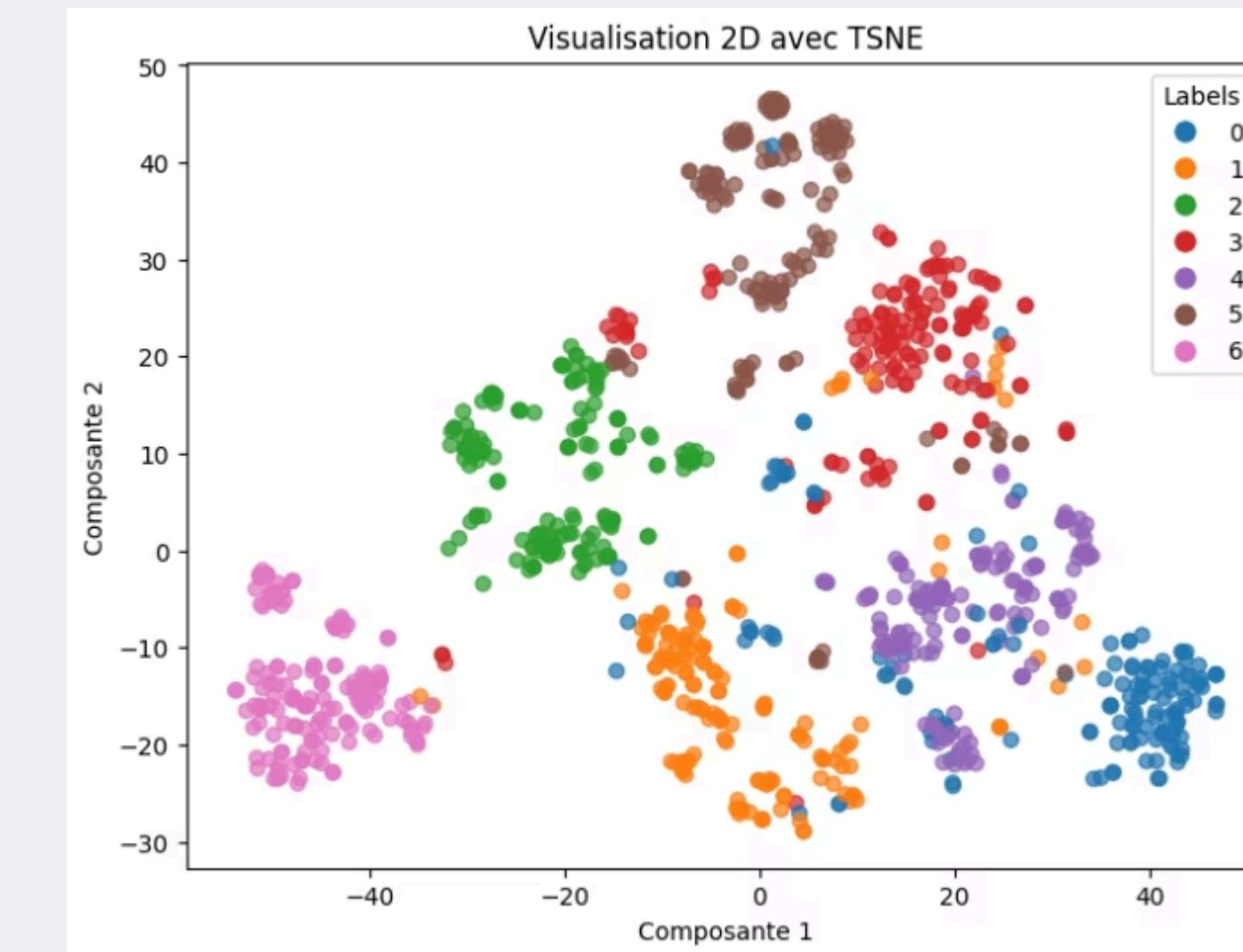
USE

L'Universal Sentence Encoder (USE) **transforme des phrases en vecteurs numériques**. Il capture la signification des phrases, permettant de comparer et regrouper des textes similaires.



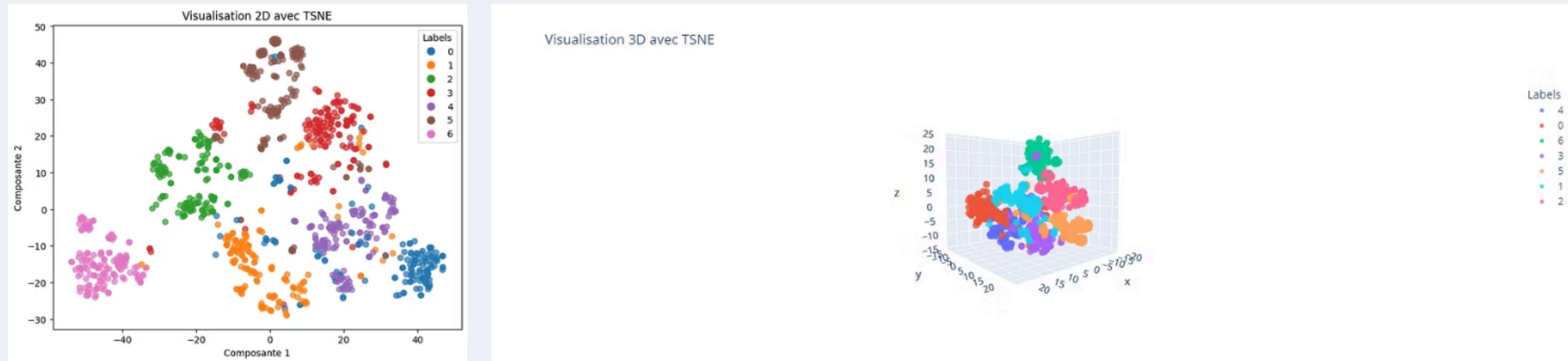
SENTENCE TRANSFORMERS

Sentence Transformers **converti des phrases en vecteurs numériques, capturant leur signification et le contexte de la phrase**



Clustering Kmeans / Sentence Transformers

IMAGE des labels initiaux



Score ARI avec les données initiale: 0.653

3. Creation d'un modèle de classification d'images supervisé

1

Choix du modèle

VGG16, RESNET50, EFFICIENTNET

2

Data-Augmentations

Albumentations et keras

3

FineTuning du modèle

Definition de l'architecture, compilation,

4

Résultat Obtenu et Visualisation des perfs

Metrics, Matrice de confusion ...

Comparatifs des modèles

Modèle	Description	Points Forts	Points Faibles	Score Top-1 sur ImageNet
VGG19	Réseau avec 19 couches, simple et classique.	Facile à comprendre et à utiliser. Bon pour des tâches de base.	Moins précis que les modèles plus récents.	~73%
ResNet50	Réseau avec 50 couches, utilise des blocs résiduels.	Équilibre entre performance et complexité. Bon pour le fine-tuning.	Peut être plus lent que EfficientNet.	~76%
EfficientNet-B0	Modèle optimisé, version de base de la famille EfficientNet.	Bon équilibre entre précision et efficacité. Moins gourmand en ressources.	Moins précis que les versions plus grandes (B1 à B7).	~77%

Modèle choisi: **RESNET50**

Data Augmentation

Redimensionnement

Crop Aléatoire

**Retournement
Horizontal**

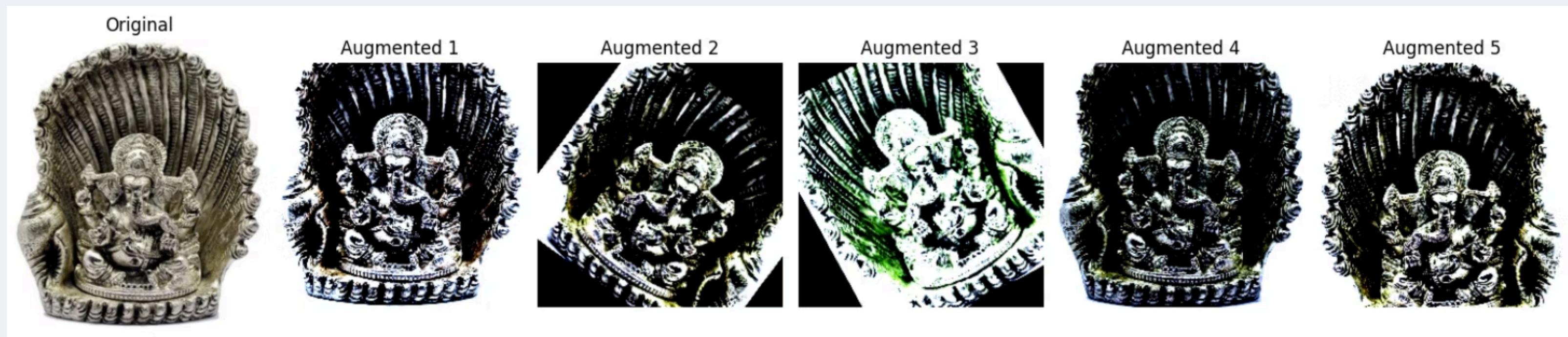
**Luminosité et
Contraste**

Rotation et Echelle

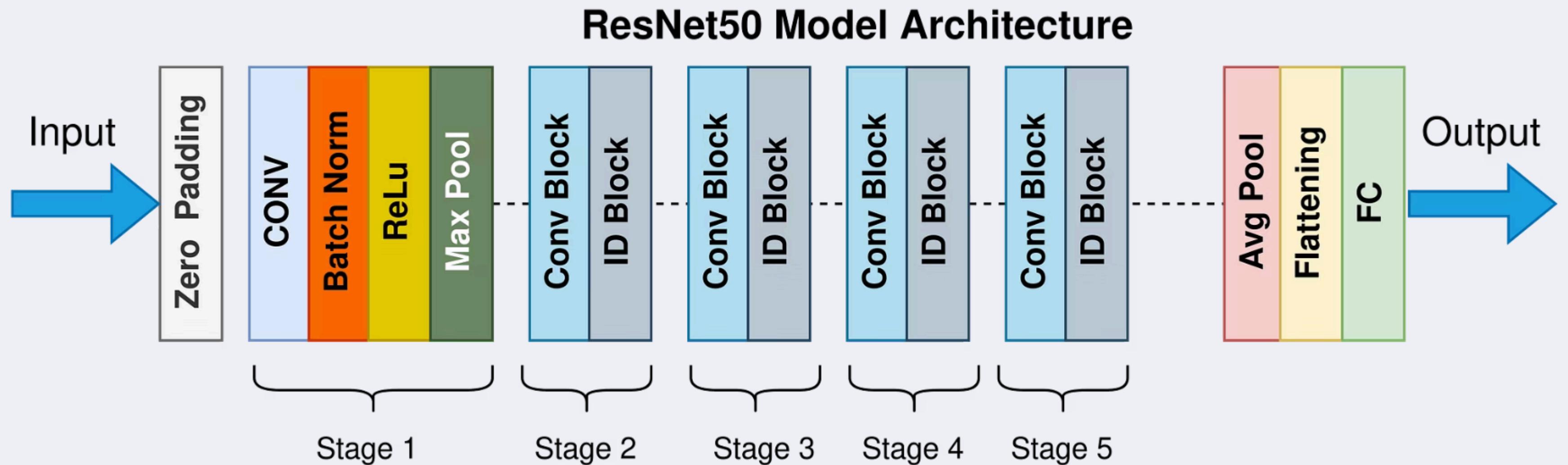
**Teinte, Saturation,
Valeur**

Dropout Coarse

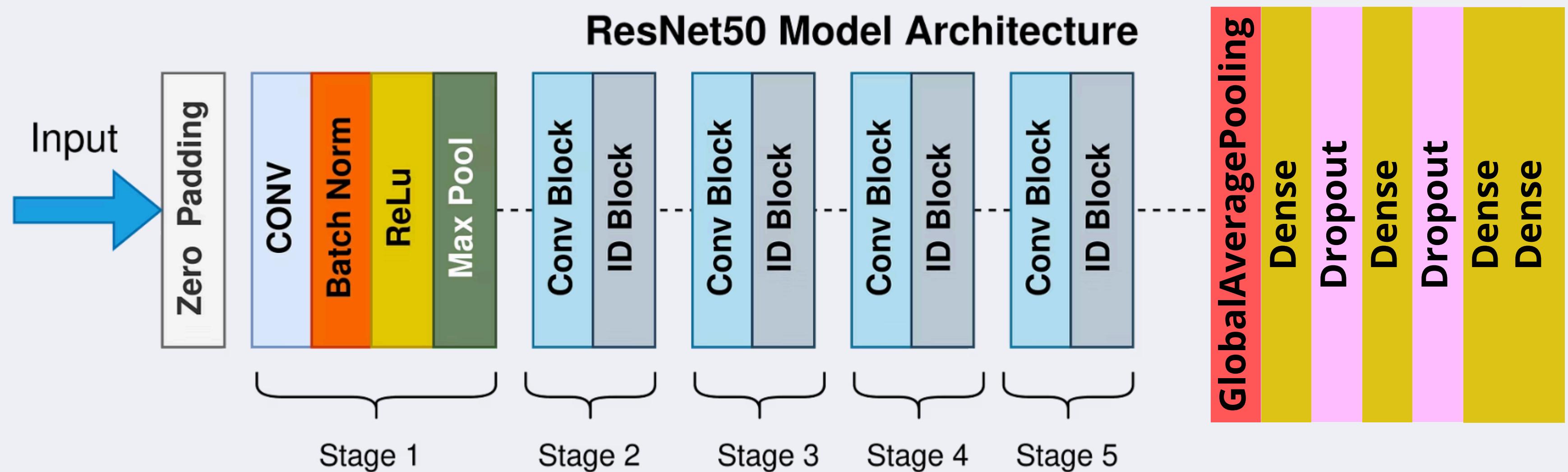
Normalisation



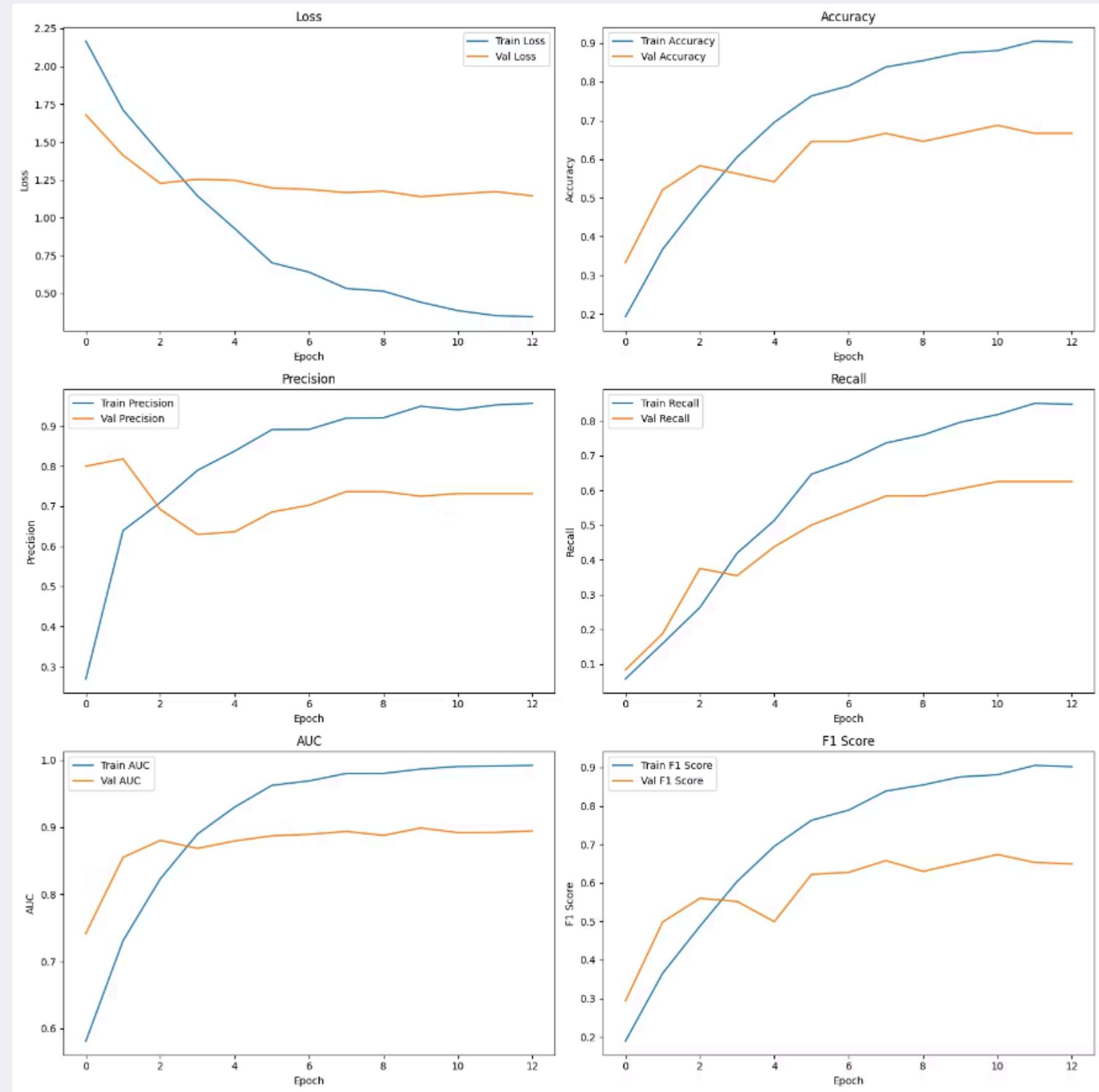
Architecture Resnet50

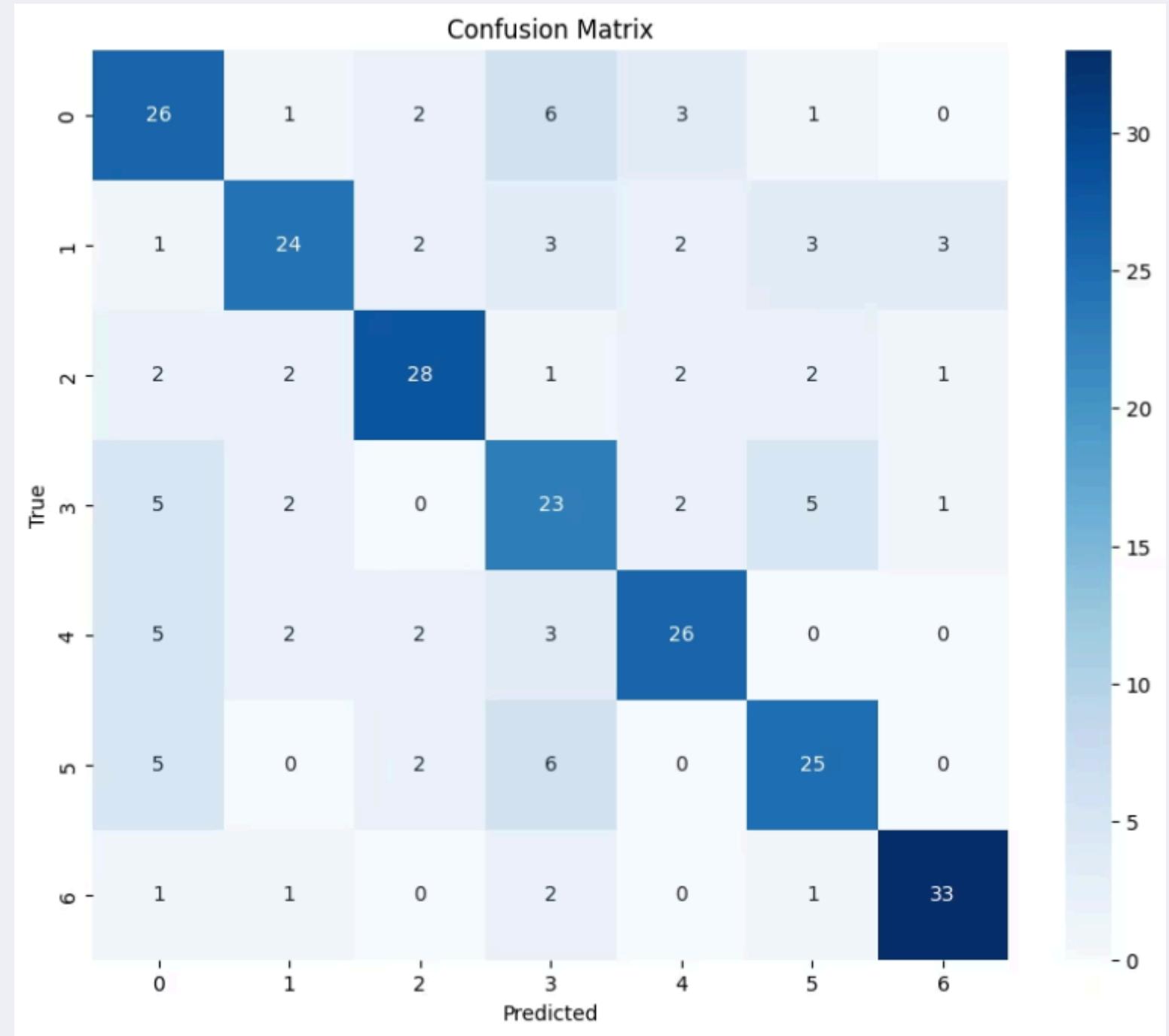


Couches ajoutées



Performance du







Conclusion

Il est donc possible et judicieux de faire un modèle de classification supervisé dans le cadre de classification visuelle ou textuelle de bien de consommation, cepeandant il serait judicieux de pouvoir avoir plus de données global afin de garantir la fiabilité et la performance du/des modèle(s)

Axe d'amélioration

Augmenter le nombre d'images et de descriptions

Pertinent pour une POC mais insuffisant, voire dangereux pour l'utilisation en production

Monitoring des modèles, du datadrift au cours du temps

Permet de s'assurer de la pertinence du modèle au cours du temps et des tendances du marché

Création d'une API d'inference

Pour faciliter l'utilisation du modèle dans une application tierce

Tester une approche avec un Modèle VL

LLM capable de traiter à la fois le Texte et les images, peut être plus performant mais est beaucoup plus coûteux en ressources computationnelles