Membre du groupe :

- BOURENNANI Anis
- Fillali Dhia Eddine

Projet 1 : Topic Modeling des Avis des Produits

Ce projet vise à analyser les avis clients issus de dataset
« Cell_phones_and_accessories »

Objectif :

- Identifier les sujets principaux abordés dans les avis clients (Topic Modeling).
- Mesurer les sentiments exprimés dans les avis pour évaluer la satisfaction des clients.
- Fournir des insights exploitables sur les tendances et les perceptions des produits.

Prétraitement des Avis de Produits

Chargement et exploration des données

- Extraction des colonnes **rating**, **title** et **text** qu'on estime essentielles du fichier « **reviews.jsonl »**
- Première visualisation des données avec **df.head()**, ce qui nous donnent :

Données avec champs sélectionnés :

	rating	title	text
0	4	No white background! It's clear!	I bought this bc I thought it had the nice whi...
1	5	Awesome! Great price! Works well!	Perfect. How pissed am I that I recently paid ...
2	5	Worked but took an hour to install	Overall very happy with the end result. If you...
3	4	Decent	Lasted about 9 months then the lock button bro...
4	5	LOVE IT!	LOVE THIS CASE! Works better than my expensive...

Traitement linguistique

On a choisi d'utiliser « **SapCy** » car c'est une bibliothèque performante pour le traitement du langage naturel (NLP), nous avons préférer à NLPK car elle est idéale pour les documents pas trop volumineux. Dans cette étape, nous avons :

- **Tokeniser** les textes : décomposer en unités linguistiques appelées tokens.
- **Lemmatiser** les mots : extraire leur forme de base.
- **Supprimer :**

- Les **stop words** (des mots sans signification spécifique, comme "the" ou "is").

- Les termes non alphabétiques ou courts (longueur ≤ 2).

La lemmatisation est essentielle pour regrouper les variantes d'un même mot

Exemple : « Running » devient « run »

Le but de ces étapes est de réduire le bruit, notamment la suppression des stop words et des termes non patients

Etape 2 :

Pour transformer les données textuelles en une matrice numérique exploitable, nous avons utilisé **TF-IDF** (Term Frequency-Inverse Document Frequency). Cette méthode attribue un poids élevé aux mots fréquents dans un document mais rares dans l'ensemble des documents.

- **TF-IDF** est une méthode classique et efficace pour les textes non supervisés, capturant l'importance relative des mots sans ajouter de biais liés à leur fréquence globale.
- Limiter le vocabulaire à **5000** caractéristiques améliore l'efficacité tout en conservant les informations essentielles

Clustering des avis avec DBSCAN

Nous avons utilisé **DBSCAN** (Density-Based Spatial Clustering of Applications with Noise) pour regrouper les avis clients en clusters selon leurs similarités.

- **DBSCAN** est adapté pour détecter des clusters de formes arbitraires et ignorer les points de bruit (non pertinents).
- La métrique cosine est pertinente pour les données textuelles représentées par des vecteurs TF-IDF.
- Les hyperparamètres **eps=0.5** (rayon de recherche) et **min_samples=10** (nombre minimum de points dans un cluster) assurent un bon compromis entre sensibilité et robustesse.

Avec cette méthode, le nombre de clusters identifiés : **4**

Nous avons également calculé le **score de silhouette** pour évaluer la qualité du clustering (entre -1 et 1, où une valeur proche de 1 indique un bon clustering) ,on a un score de : **0.360**, ce qui est relativement bon.

Analyse des clusters et extraction des mots-clés

Pour chaque cluster, nous avons extrait les mots les plus fréquents à l'aide de **CountVectorizer**.

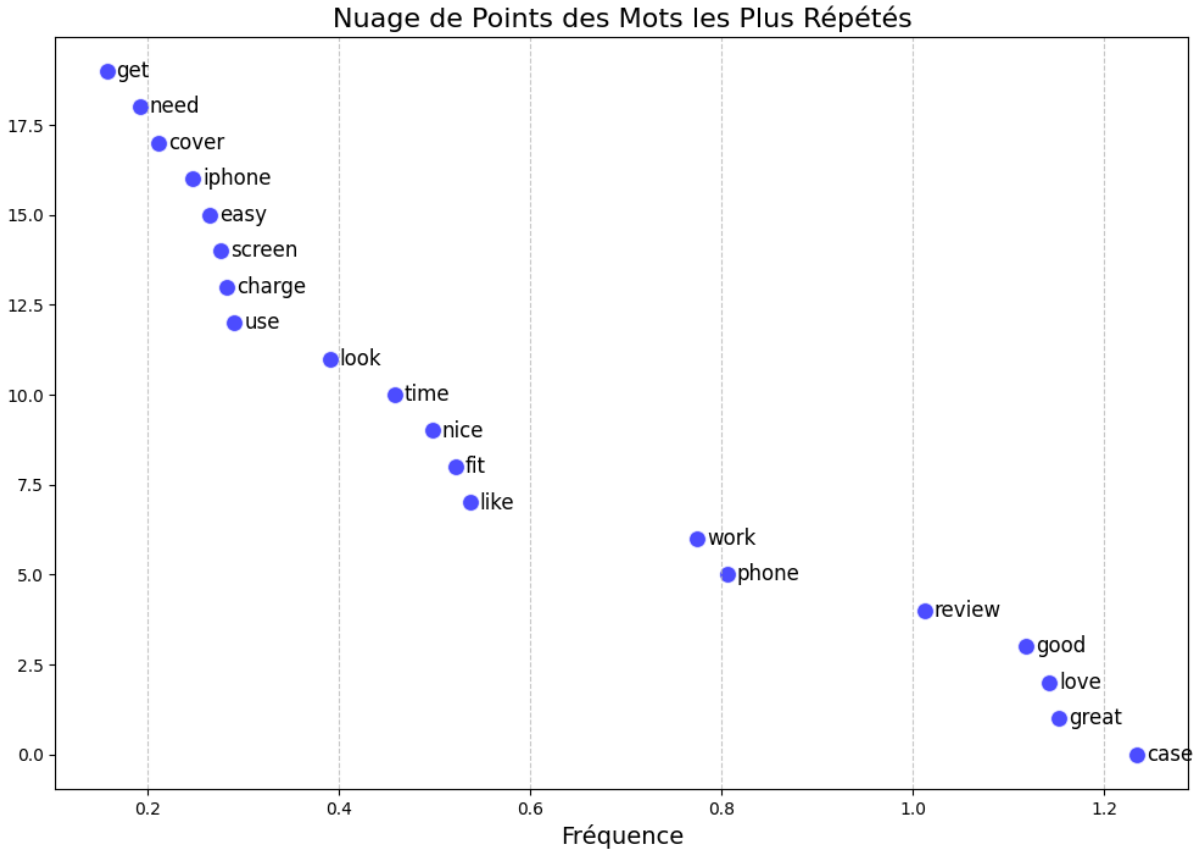
Les fréquences simples fournissent une première idée des thèmes dominants dans chaque cluster.

On a obtenu le résultat suivant :

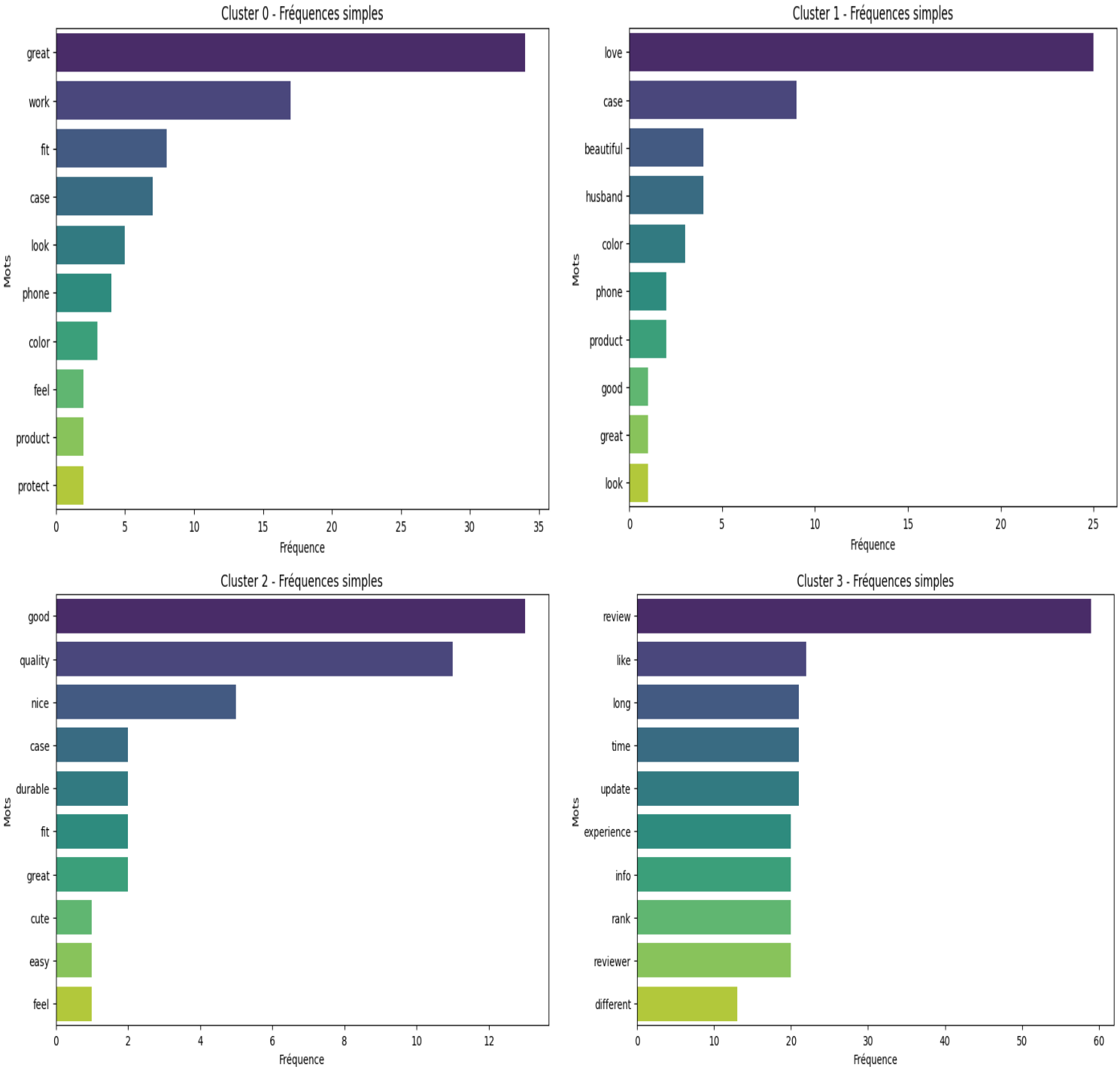
```
Cluster -1 :
Mots-clés les plus fréquents : [('phone', 757), ('case', 576), ('work', 259), ('great', 222), ('like', 220), ('good', 2

Cluster 0 :
Mots-clés les plus fréquents : [('great', 34), ('work', 17), ('fit', 8), ('case', 7), ('look', 5), ('phone', 4), ('colo

Cluster 1 :
Mots-clés les plus fréquents : [('love', 25), ('case', 9), ('beautiful', 4), ('husband', 4), ('color', 3), ('phone', 2)

Cluster 2 :
Mots-clés les plus fréquents : [('good', 13), ('quality', 11), ('nice', 5), ('case', 2), ('durable', 2), ('fit', 2), ('

Cluster 3 :
Mots-clés les plus fréquents : [('review', 59), ('like', 22), ('long', 21), ('time', 21), ('update', 21), ('experience',
```

On note que le **cluster -1** représente les bruits !

On remarque la redondance de mots comme « Phone », « Case » qui ne sont pas très intéressant pour nos analyses !



En effet, comme on peut le voir sur le nuage de point ci-dessus, on trouve certains mots comme « case », « phone » ou « review »

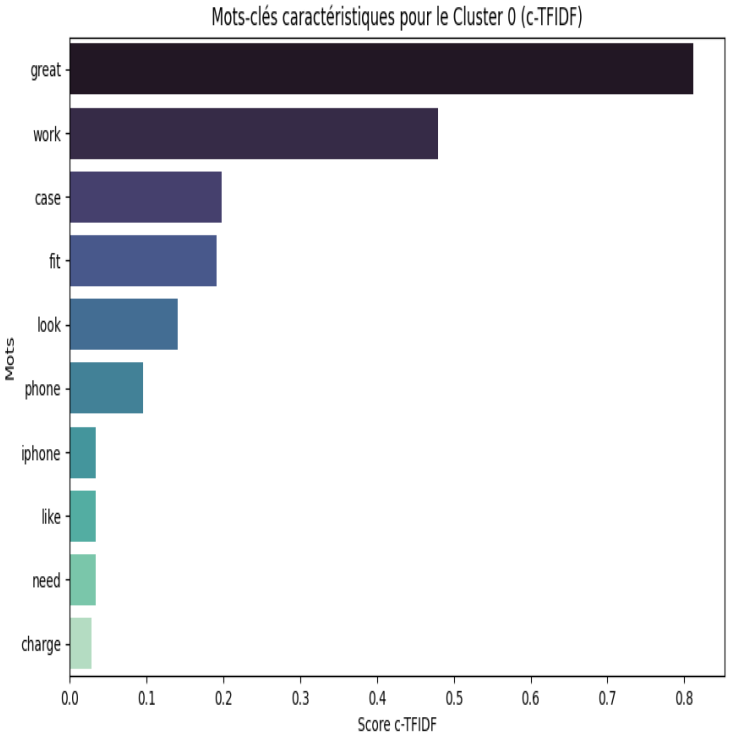


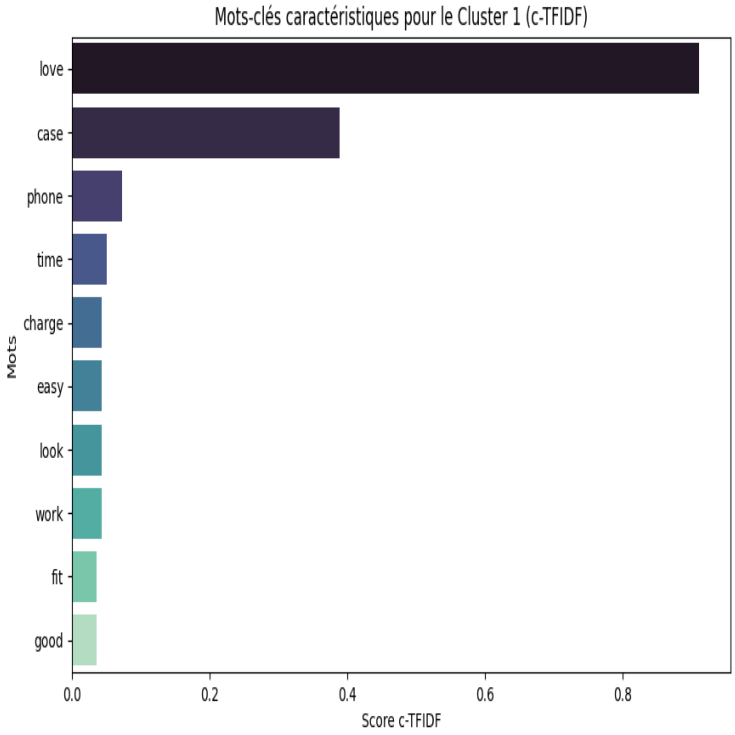
On a ci-dessous une représentation graphique des mots les plus répétés pour nos 4 clusters.

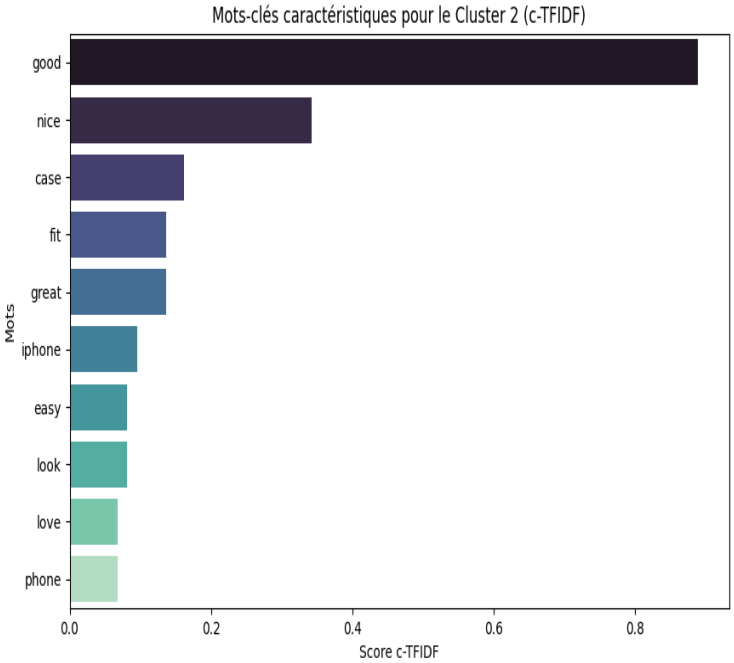
Mots-clés avec c-TFIDF

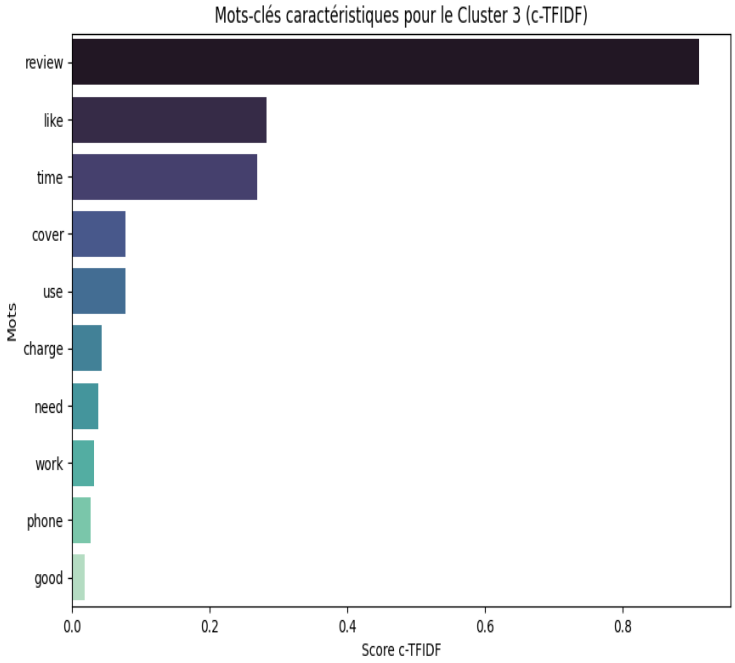
Nous avons utilisé le **c-TFIDF** (class-based TF-IDF), une variante de TF-IDF, pour identifier les **mots-clés** les plus caractéristiques des clusters.

- Le c-TFIDF est particulièrement utile pour identifier les différences spécifiques entre les clusters.









Ça nous donne le résultat suivant :

Analyse :

On remarque que certains mots sont moins récurent après utilisation de c-TFIDF, donc une nette amélioration pour faire des analyses, on remarque aussi que pour les 4 cluster, ce sont des avis positifs (avec les mots « good », « nice » ou encore « great »)

Analyse des sentiments

Dans cette partie, Nous avons utilisé le modèle pré-entraîné **nlptown/bert-base-multilingual-uncased-sentiment** pour effectuer une analyse de sentiments des avis.

- Ce modèle multilingue est adapté aux textes en anglais et permet une analyse rapide et fiable des sentiments.

Evaluer les performances :

Nous avons calculé deux métriques principales pour évaluer les performances,

- Corrélation de Pearson : pour évaluer la relation entre notes réelles et prédites.

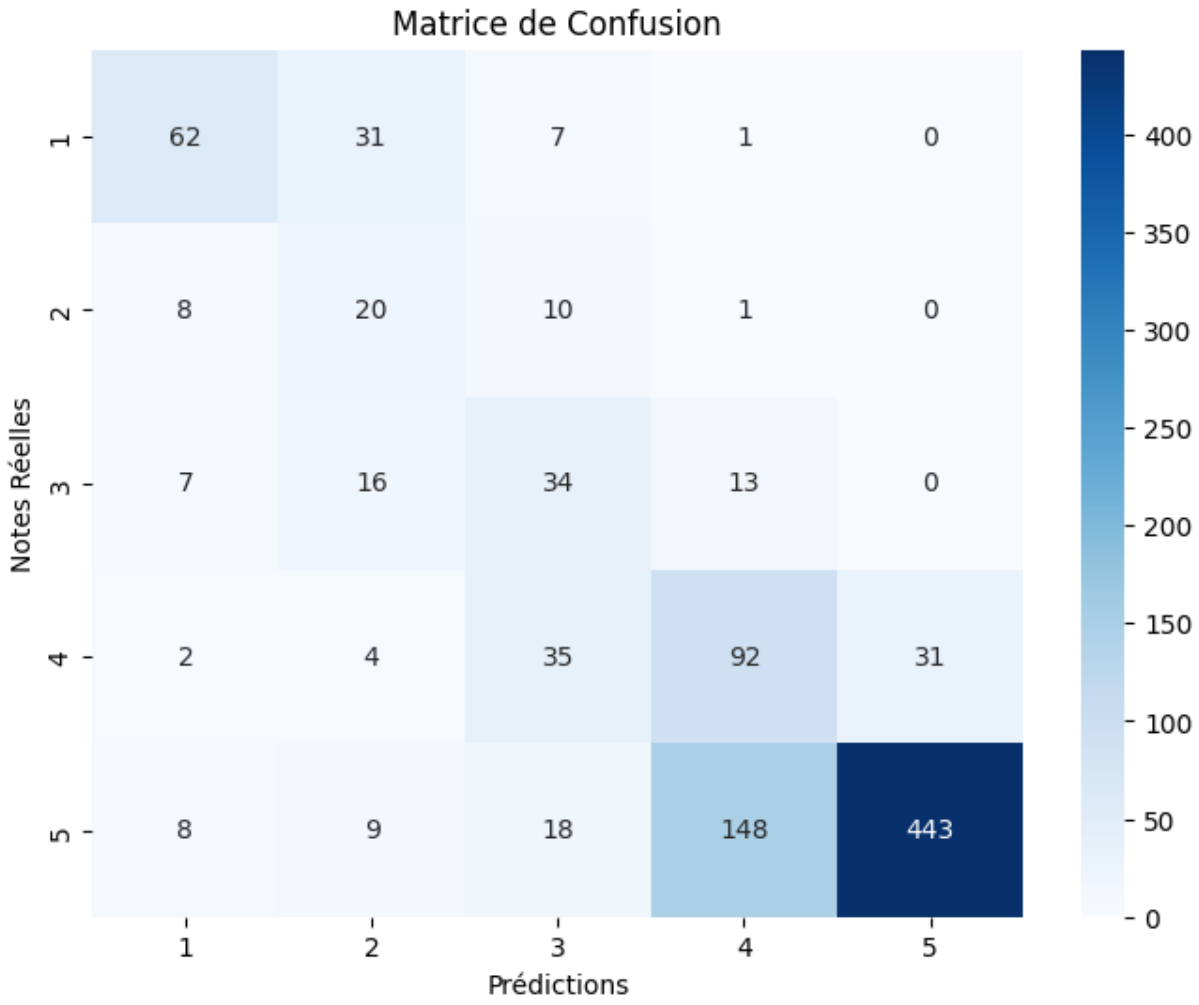
Ça nous a donné un coefficient de **0.817** (ce qui est proche de 1, donc il y a une **relation positive forte** entre les note réelle et prédites)

- Rapport de classification : accuracy, precision, recall, F1-score, et matrice de confusion.

Rapport de Classification :

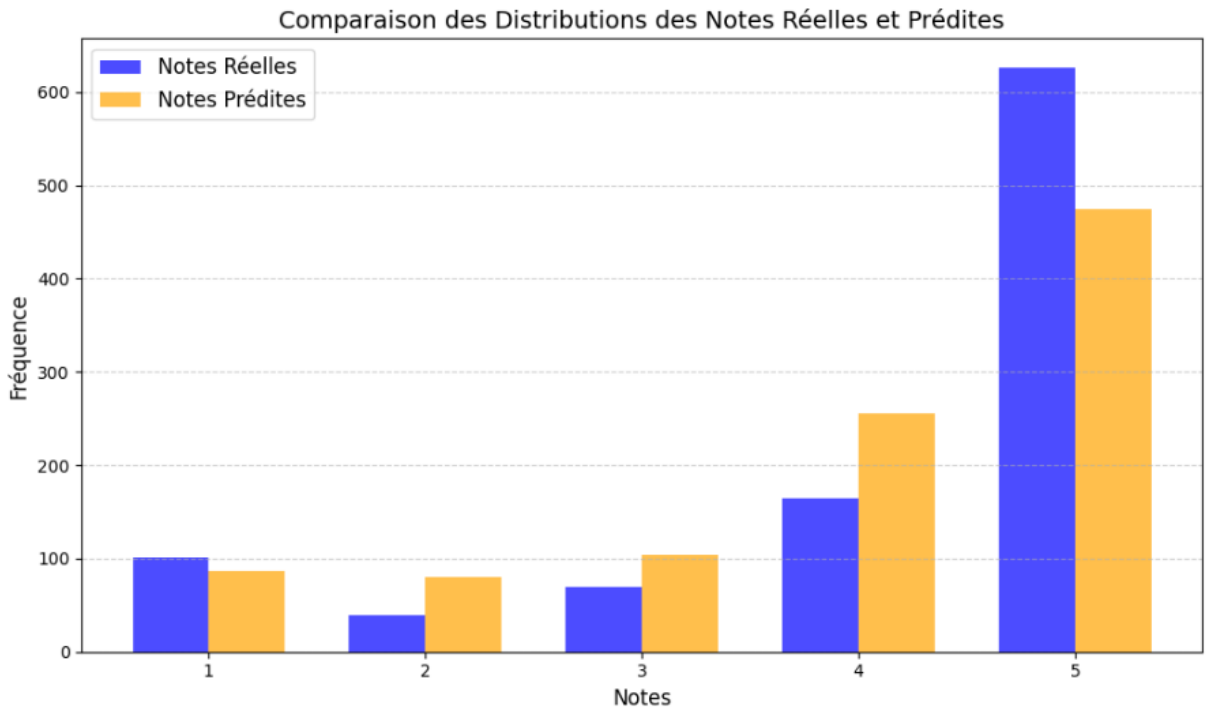
	precision	recall	f1-score	support
1	0.71	0.61	0.66	101
2	0.25	0.51	0.34	39
3	0.33	0.49	0.39	70
4	0.36	0.56	0.44	164
5	0.93	0.71	0.81	626
accuracy			0.65	1000
macro avg	0.52	0.58	0.53	1000
weighted avg	0.75	0.65	0.68	1000

On remarque une accuracy de **65%**



On remarque que les prédictions pour le **rating 5** sont le plus souvent correctes avec **93%** de précision

Comparaison des distributions des notes réelles et prédites avec le graphique ci-dessous :



Conclusion :

Chaque étape et technique choisie, de **TF-IDF** à **DBSCAN** et **c-TFIDF**, a été motivée par le besoin de traiter des données textuelles de manière efficace et robuste. Cette combinaison a permis d'identifier les thèmes sous-jacents, de regrouper les avis en clusters significatifs, et d'effectuer une analyse fine des sentiments.

En résumé, nous avons essayé de faire un pipeline complet qui couvre les principales étapes de traitement NLP pour l’analyse des avis produits Amazon.