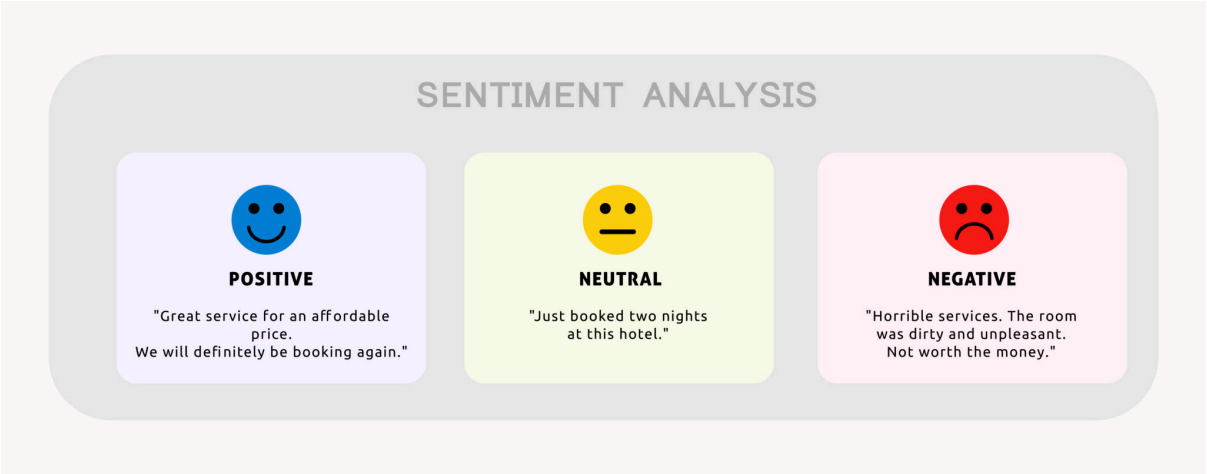
Rapport du Sentiment analysis

TP1 par CHALHAJ SALIM



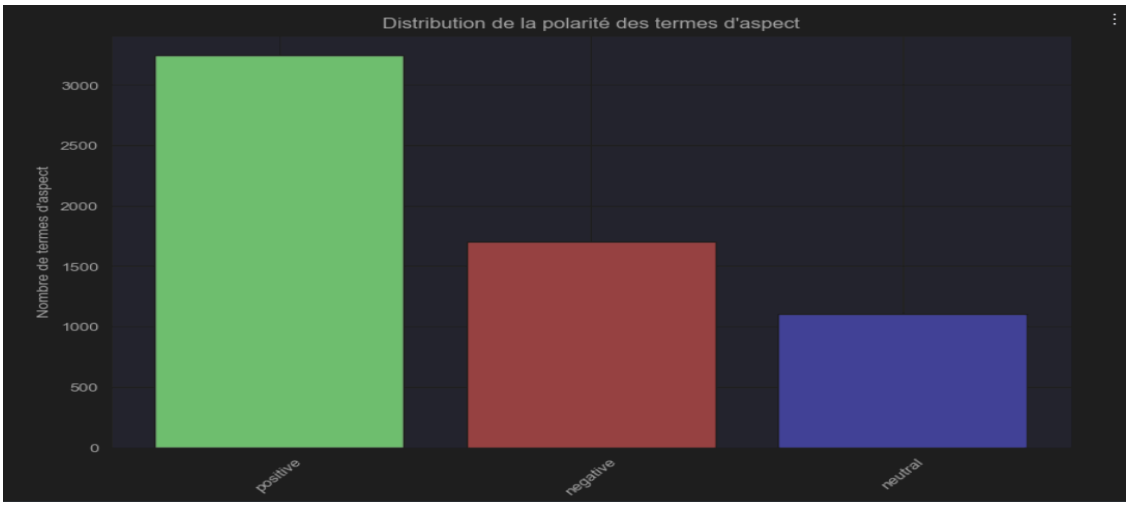
Dans le code, le fichier nommé "partie1" se consacre à l'accomplissement du premier objectif, tandis que le fichier "analysis2" aborde et réalise le second objectif.

Objectif 1 : calculer la polarité des mots dans les deux jeux de données à l’aide d’un lexicon de sentiment

Méthodologie

Extraction des Données: Utilisation de **XML.etree.ElementTree** de Python pour l'extraction efficace des données structurées XML, permettant l'accès aux avis clients, aux aspects mentionnés et à leurs polarités associées.

Analyse des Données: : spaCy a été employé pour le traitement du langage naturel, couplant tokenisation, lemmatisation, et identification des parties du discours, avec SentiWordNet pour une analyse de sentiment précise, facilitant ainsi une compréhension profonde des sentiments des clients.



Résultats Clés

Distribution des Polarités:

Une dominance de polarités positives avec des signes d'amélioration requis dans le domaine technologique.

Termes et Catégories d'Aspect les Plus Fréquents:

“ Service "qualité de la nourriture" pour la restauration; "fonctionnalités" et "performance" pour la technologie, avec des critiques négatives ciblant la "batterie" et le "support client".

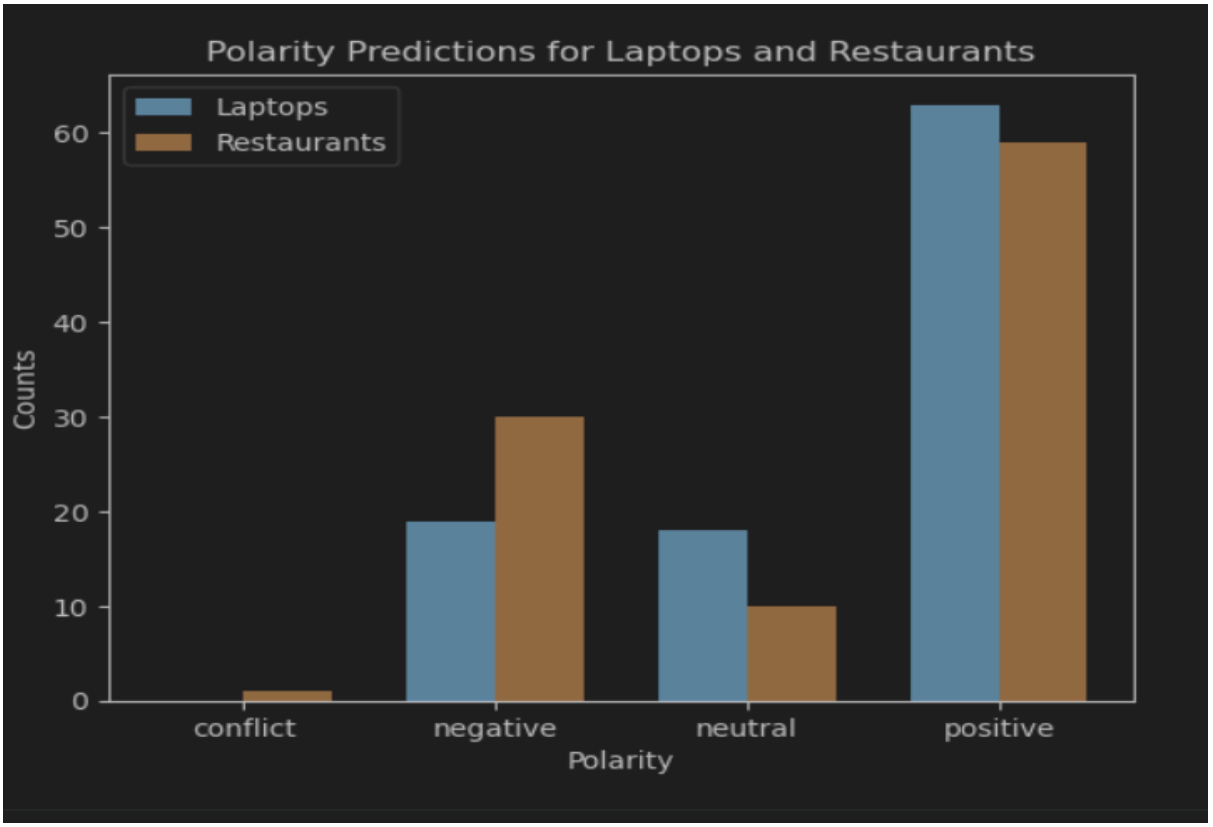
Référence :

- SentiWordNet pour l'évaluation des polarités.
- Python's XML.etree.ElementTree pour une extraction efficace et structurée des données à partir de fichiers XML.
- Matplotlib pour la création de visualisations graphiques, illustrant la distribution des polarités des termes d'aspect et fournissant une représentation visuelle des données analysées.

Objectif 2 : Déterminer la polarité des termes des aspects

Extraction des Données :
Les avis clients ont été extraits de fichiers XML en utilisant xml.etree.ElementTree, se concentrant sur les termes d'aspect et leurs polarités associées.

Traitement des Données :
Les données textuelles ont été transformées en vecteurs TF-IDF à l'aide de TfidfVectorizer de Scikit-learn, fournissant une représentation numérique pour l'entraînement du modèle.
Modélisation Initiale : Un modèle de forêt aléatoire (RandomForestClassifier) a été entraîné sur les données, atteignant une précision initiale de 59% sur l'ensemble de test.
Optimisation : Une recherche des hyperparamètres (GridSearchCV) a été effectuée, affinant les paramètres tels que le nombre d'arbres (n_estimators) et la profondeur maximale (max_depth), aboutissant à une amélioration significative de la précision à 69%.
Résultats
L'analyse des résultats montre que l'optimisation des hyperparamètres du modèle a eu un impact significatif sur la performance, soulignant l'importance d'un réglage fin dans les tâches de classification de texte. Les prédictions améliorées fournissent des insights plus précis.
Quelques visualisation au niveau de la prédiction :



Conclusion

En conclusion, ce tp a souligné la puissance du traitement du langage naturel et de l'apprentissage automatique pour tirer des insights de textes complexes, et a montré qu'un ajustement soigné peut nettement booster l'exactitude de l'analyse des sentiments.