TP : Analyse de sentiment des critiques de films provenant d'IMDb (Internet Movie Database)

Dans le monde numérique d'aujourd'hui, les critiques de films abondent sur diverses plateformes en ligne, fournissant des opinions variées et souvent passionnées. Ce TP d'analyse de sentiments vise à exploiter le pouvoir de l'apprentissage automatique pour comprendre et classifier ces opinions. En utilisant des techniques avancées de traitement du langage naturel, notre objectif est de développer un modèle capable de discerner automatiquement si une critique de film exprime un sentiment positif ou négatif.

Objectif du TP :

L'objectif de ce TP est de **développer un modèle d'analyse de sentiments** en utilisant ce jeu de données. En utilisant **le texte des critiques comme entrée**, le modèle sera formé pour **prédire automatiquement et précisément si le sentiment exprimé dans une critique est positif ou négatif**.

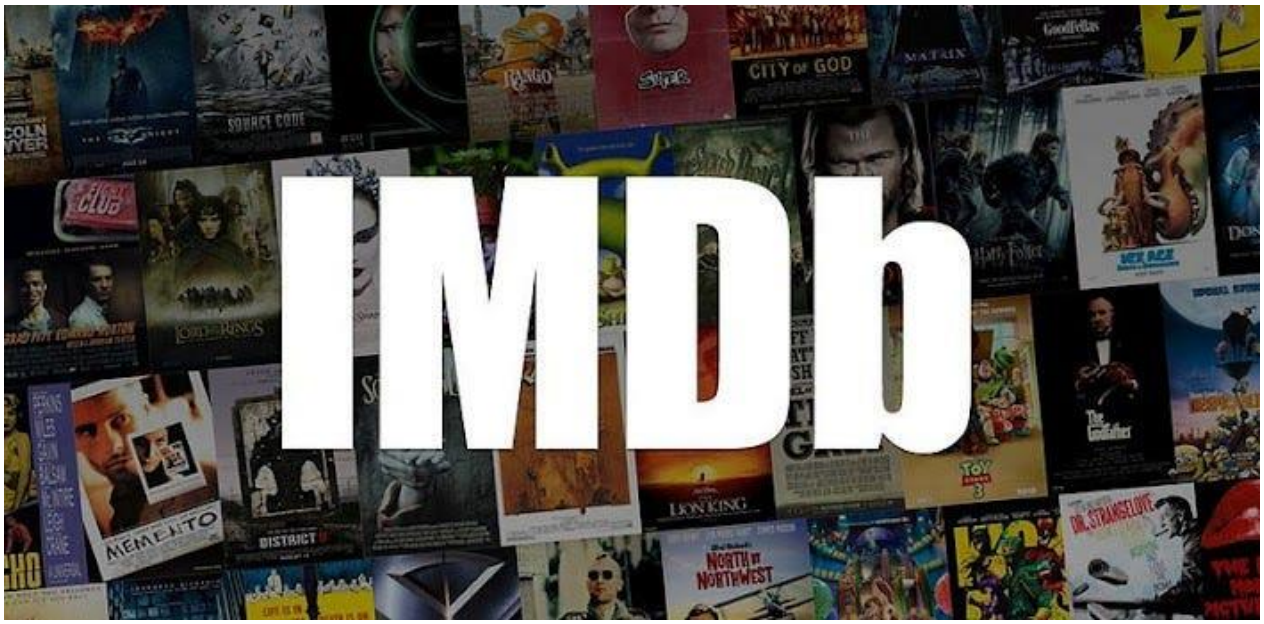
Le jeu de données (Dataset) :

Ce jeu de données provient de la plateforme **Kaggle** et se concentre sur l'analyse de sentiments à partir de **critiques de films provenant d'IMDb (Internet Movie Database)**. Voici une description du dataset :

cette dataset se compose de 50 000 critiques de films, étiquetées avec des sentiments positifs ou négatifs, dans le but de former des modèles d'analyse de sentiments. Chaque critique est associée à deux colonnes principales :

- **Critique (Review) :** Cette colonne contient le texte de la critique de film, servant de variable d'entrée pour le modèle d'analyse de sentiments.
- **Sentiment (Sentiment) :** Cette colonne indique la polarité du sentiment associé à la critique, prenant les valeurs `positive` ou `negative`.

Lien du dataset : https://www.kaggle.com/code/shubhamptrivedi/sentiment-analysis-on-imdb-movie-reviews/input?select=IMDB+Dataset.csv



La réalisation du TP :

- **Exploration des données (EDA) :**

Cette phase vise à comprendre les caractéristiques principales du jeu de données avant d'appliquer les modèles statistiques ou d'apprentissage automatique. Vous devez suivre les étapes suivantes :

1. Importer les bibliothèques nécessaires :

- Commencez par importer toutes les bibliothèques Python indispensables pour l'EDA et l'analyse de sentiment telles que Pandas, NumPy, Matplotlib, Seaborn, et éventuellement NLTK ou SpaCy pour le traitement du langage naturel.

2. Charger le jeu de données :

- Utilisez Pandas pour charger le jeu de données à partir d'un fichier CSV ou d'une autre source. Assurez-vous que le chemin du fichier est correct et que le fichier est bien structuré.

3. Visualiser les informations sur le jeu de données :

- Aperçu initial : Utilisez des commandes comme *data.head()*, *data.info()*, et *data.describe()* pour obtenir un aperçu général des premières lignes, des types de données, et des statistiques descriptives du jeu de données.
- Visualiser la distribution des sentiments càd la répartition des sentiments positifs et négatifs dans le jeu de données.
- Visualiser les mots les plus fréquents dans les critiques positives et négatives.

- **Prétraitement :**

Cette phase est une étape cruciale avant d'appliquer les modèles de machine learning ou d'apprentissage automatique, surtout dans notre cas car il s'agit de données textuelles pour l'analyse de sentiment. Voici les étapes à suivre pour la phase de prétraitement des données :

1. Nettoyage des Données :

- Supprimer les doublons : Éliminez les observations dupliquées dans le jeu de données.
- Suppression des caractères spéciaux et ponctuations : Enlevez les caractères non pertinents comme les ponctuations, les chiffres, et les symboles spéciaux.
- Suppression des balises HTML.

2. Tokenisation :

- Divisez le texte en mots individuels ou tokens.

3. Normalisation et Suppression des Stopwords :

- Convertissez les mots en minuscules pour assurer la cohérence et supprimez les stopwords (mots courants qui n'apportent pas beaucoup d'information) exemples `a, the, in ....`

4. Stemming ou Lemmatisation :

- Réduisez les mots à leur racine ou forme de base pour simplifier le traitement.

5. Division de données en ensembles d'entraînement et de test:

- Diviser les données en ensembles d'entraînement et de test, pour évaluer la performance du modèle sur des données qu'il n'a pas vues pendant l'entraînement.

6. La Vectorisation:

- Convertissez le texte en une représentation numérique que les modèles de machine learning peuvent utiliser. Les techniques courantes incluent Bag of Words (BoW) ou TF-IDF, Voici une brève présentation de ces deux techniques :

Modèle Bag of Words (BoW) :

Objectif : Le modèle BoW a pour objectif de représenter un document sous forme d'un **vecteur de fréquence d'occurrence des mots, en ignorant l'ordre et la structure grammaticale.**

Fonctionnement :

- Création du vocabulaire : Les mots uniques du corpus sont extraits pour former un vocabulaire.

- Création du vecteur : Chaque document est représenté par un vecteur où chaque composant correspond à la fréquence d'occurrence du mot correspondant dans le vocabulaire.

Avantages :

- Simplicité et facilité de mise en œuvre.
- Donne une représentation numérique aux données textuelles.

Inconvénients :

- Ne tient pas compte de l'ordre des mots.
- Ignore la sémantique et la structure grammaticale.
- Peut conduire à des vecteurs de grande dimension et clairsemés.

Exemple :

Corpus :

"It was the best of times best",

"it was the worst of times",

"it was the age of wisdom",

"it was the age of foolishness"

Vocabulaire :

['age' 'best' 'foolishness' 'it' 'of' 'the' 'times' 'was' 'wisdom' 'worst']

Bag of Words Matrix :

$$\begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

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

Objectif :

TF-IDF vise à **mesurer l'importance d'un mot dans un document au sein d'un corpus**, en prenant en compte à la fois la fréquence du terme et son importance relative dans l'ensemble des documents.

Fonctionnement :

- Calcul de la fréquence du terme (TF) : Représente la fréquence d'un terme dans un document.$\mathrm{TF(t,d)} = \frac{\mathrm{f_{t,d}}}{\sum_{\mathrm{t'}\in \mathrm{d}} \mathrm{f_{t',d}}}$
- Calcul de la fréquence inverse du document (IDF) : Mesure l'inverse de la proportion de documents contenant le terme. $\mathrm{IDF}(t, D) = \log\left(\frac{N}{|\{d \in D : t \in d\}|}\right)$
- Calcul du produit final : TF-IDF est obtenu en multipliant TF par IDF. $\text{TF-IDF}(t, d, D) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \times \log\left(\frac{N}{|\{d \in D : t \in d\}|}\right)$

Avantages :

- Donne plus d'importance aux termes rares et informatifs.
- Réduit l'impact des mots très fréquents.

Inconvénients :

- Nécessite un prétraitement minutieux des données.
- Sensible aux valeurs aberrantes.

exemple:

Document 1

Term	Term Count
this	1
is	1
a	2
sample	1

Document 2

Term	Term Count
this	1
is	1
another	2
example	3

$$tf(\text{example}, d_1) = \frac{0}{5} = 0$$
$$tf(\text{example}, d_2) = \frac{3}{7} \approx 0.429$$
$$idf(\text{example}, D) = \log\left(\frac{2}{1}\right) = 0.301$$

$$tf\text{-}idf(\text{example}, d_1, D) = tf(\text{example}, d_1) \times idf(\text{example}, D) = 0 \times 0.301 = 0$$
$$tf\text{-}idf(\text{example}, d_2, D) = tf(\text{example}, d_2) \times idf(\text{example}, D) = 0.429 \times 0.301 \approx 0.129$$

- **Entraînement et evaluation du modèle :**

L'entraînement et l'évaluation d'un modèle de machine learning sont des étapes cruciales dans le processus de développement. Dans ce TP, nous allons utiliser les modèles `Naive Bayes` et `la régression logistique`.

- **Interprétation des Résultats :**

Naive Bayes :

- Accuracy :
- Classification Report :
- Matrice de Confusion :

Régression Logistique :

- Accuracy :
- Classification Report :
- Matrice de Confusion :

Choix du Modèle :

- choisir le modèle qui a les meilleures performances globales.

Testé le modèles sur des exemples réels :

Example de positifs/négatifs critique (Review) :

- *Positive Reviews:*

The movie was absolutely fantastic! The storyline, characters, and visuals were all top-notch. I highly recommend it to everyone.

- *Negative Reviews:*

I was very disappointed with this movie. The plot was confusing, and the characters were poorly developed. I wouldn't recommend it to anyone.