

# Analyse de Sentiments pour les Critiques de Films

## 1. Traitement du Langage Naturel (NLP)

### Chargement des données

Nous avons commencé par charger les données de critiques de films. Ces données comprennent le texte de la critique et une étiquette indiquant si la critique est positive ou négative.

### Prétraitement du texte et vectorisation

Nous avons ensuite effectué un prétraitement sur le texte des critiques. Cela comprenait la suppression des caractères non alphabétiques, la conversion du texte en minuscules, et la tokenisation du texte en mots individuels.

Nous avons utilisé une méthode de vectorisation pour convertir les mots en vecteurs numériques que notre modèle peut comprendre. Pour cela, nous avons utilisé la méthode de l'encodage one-hot.

### Entraînement du modèle de classification de texte

Nous avons entraîné un modèle de classification de texte pour prédire si une critique est positive ou négative. Pour cela, nous avons utilisé un réseau de neurones convolutifs (CNN). Nous avons choisi un CNN car ils sont efficaces pour détecter les caractéristiques locales et spatiales dans le texte.

## 2. Déploiement sur une Interface Web

### Création de l'interface web

Nous avons créé une interface web simple qui permet aux utilisateurs de soumettre une critique de film et d'obtenir une prédiction sur son caractère positif ou négatif. L'interface web a été construite en utilisant HTML et JavaScript.

### Création de l'API

Nous avons créé une API en utilisant Flask. Cette API accepte les requêtes POST contenant une critique de film, utilise le modèle pour prédire le sentiment de la critique, et renvoie la prédiction.

### Processus de déploiement

Pour déployer notre application, nous avons d'abord sauvegardé notre modèle entraîné et notre tokenizer. Nous avons ensuite utilisé ces fichiers dans notre application Flask pour effectuer les prédictions. Enfin, nous avons lancé notre serveur Flask et notre interface utilisateur.