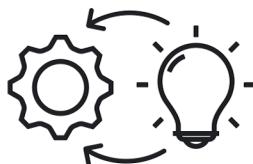


Analyse de Sentiments de films et Système de Recommandations

Fannich Salma

Le projet repose sur le développement d'un système polyvalent alliant *analyse de sentiments et recommandation*, avec une interface utilisateur conviviale. L'objectif principal est de permettre aux utilisateurs d'explorer et de bénéficier d'une expérience personnalisée. Dans ce contexte, l'analyse de sentiments vise à évaluer les commentaires des utilisateurs sur des films, les classifiant en catégories telles que positives, négatives ou neutres. Cette évaluation alimente un système de recommandation qui suggère des films en fonction des préférences sentimentales des utilisateurs. L'interface utilisateur offre une plateforme intuitive pour interagir avec ces fonctionnalités.

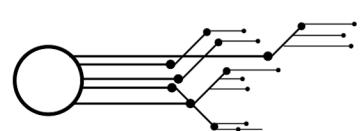
1. Introduction :



Dans cette ère du big data, l'apprentissage automatique émerge comme un domaine de recherche incontournable. Il permet une analyse efficace de données massives, jouant un rôle essentiel dans la classification et la prédiction du contenu du langage, également connu sous le nom de traitement du langage naturel (NLP). L'une des zones les plus remarquables de la NLP est l'analyse de sentiment, qui s'applique généralement à trois niveaux : au niveau de la phrase, au niveau du document et au niveau de l'aspect. Le niveau de la phrase analyse le sentiment de chaque phrase, le niveau du document classe l'ensemble du document en classe binaire ou multiclass, tandis que le niveau de l'aspect identifie d'abord les différents aspects d'un corpus, puis classe chaque document en fonction des aspects observés.

Parallèlement à cette analyse de sentiment, le projet intègre un système de recommandation novateur. Les systèmes de recommandation traditionnels peuvent souvent manquer de personnalisation, ne prenant pas en compte les sentiments et les préférences subjectives des utilisateurs. Ainsi, la problématique sous-jacente à ce projet réside dans la nécessité de rendre les systèmes de recommandation plus personnalisés et pertinents. En intégrant l'analyse de sentiment, le projet vise à résoudre ce problème en offrant des suggestions de films personnalisées en fonction des émotions dégagées par les utilisateurs lors de leurs critiques. Cette approche vise à créer une expérience plus immersive et à répondre de manière proactive aux attentes des utilisateurs.

Le rapport se concentre sur la classification des représentations sentimentales des critiques de la base de données en ligne (IMDb) grâce à la classification basée sur l'apprentissage automatique au niveau du document. Le rapport commence par éliminer les mots vides et normaliser les mots dans les critiques IMDb pour améliorer les performances de la classification. Ensuite, les critiques sont transformées en une matrice de mots, représentant les caractéristiques de la classification. Enfin, plusieurs algorithmes (régression logistique, SVM, Naïve Bayes, Random Forest ,Gradient Boosting) sont utilisés pour former et tester la matrice de mots afin d'évaluer quel algorithme peut mieux fonctionner pour cette classification.



Suite à l'évaluation de ces différents algorithmes, le modèle de machine learning offrant les meilleures performances est sélectionné pour alimenter le système de recommandation. Une fois ce modèle intégré dans le système de recommandation, des techniques de mesure de similarité, telles que le cosme et le Jaccard, sont exploitées pour évaluer la proximité sémantique entre les critiques. Ces métriques de similarité servent de fondement à la recommandation de films similaires, en prenant en compte à la fois les préférences antérieures et les sentiments exprimés par les utilisateurs dans leurs critiques.

En parallèle, une interface utilisateur est développée pour permettre aux utilisateurs d'interagir facilement avec le système. L'interface prend en charge la saisie de critiques, offre des prédictions de sentiment en temps réel, et propose des recommandations de films personnalisées.

I . PARTIE 1: Analyse de Sentiment

1.Méthodologie



Le rapport adopte une méthodologie pour mener l'analyse de l'analyse de sentiment des critiques IMDb, comme le montre la Fig. 1. Tout d'abord, le rapport illustre et alimente les données dans le processus de nettoyage et de prétraitement des données. Ensuite, les mots vides et certains mots non pertinents sont supprimés des données d'origine ; et applique quelques visualisations puis, les techniques de vectorisation sont appliquées pour transformer le texte en une matrice de caractéristiques. Enfin, le rapport applique cinq algorithmes différents pour former et tester la matrice de caractéristiques.

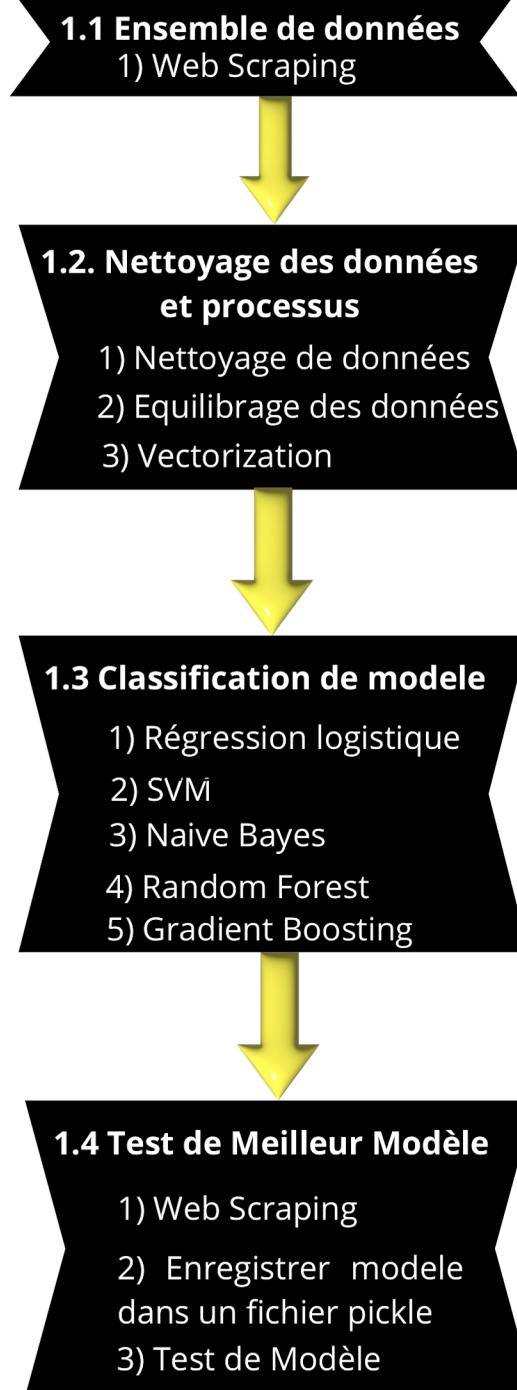


Fig. 1. Méthodologie

1.1 Ensemble de données



1) Web Scraping

Dans le cadre de ce projet, j'ai réalisé le web scraping des 250 meilleurs films du site IMDb [1] en utilisant la bibliothèque Scrapy. Chaque film a été extrait avec ses commentaires associés. Pour analyser le sentiment de ces commentaires, j'ai utilisé la bibliothèque TextBlob, qui permet de classifier les textes en sentiments positifs, négatifs ou neutres.

70% de ces données pour l'ensemble d'entraînement et 30% pour ensemble de test . Cette division est essentielle pour évaluer la performance des modèles d'analyse de sentiment sur des données non vues, assurant ainsi une évaluation fiable de la généralisation des algorithmes.

1.2. Nettoyage des données et processus

1) Nettoyage de données

Dans le but de faciliter l'interprétation des données dans les travaux ultérieurs, les textes bruts obtenus de la section précédente sont prétraités. Tout d'abord, des éléments tels que la ponctuation, les sauts de ligne, les chiffres et les mots vides tels que « a », « the » et « of » sont supprimés, car ils fournissent peu d'informations sur l'impression de l'utilisateur envers un film.



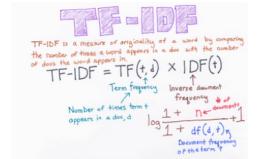
2) Equilibrage des données

Dans la section d'équilibrage des données, l'un des défis majeurs rencontrés concerne le déséquilibre significatif entre les classes d'opinions. Au départ, l'ensemble de données présente un déséquilibre avec

16 657 commentaires positifs, 3510 commentaires négatifs et 1809 commentaires neutres. Afin de remédier à cette disparité, deux techniques d'équilibrage, à savoir **le suréchantillonnage** (oversampling) et **le sous-échantillonnage** (undersampling),[2] sont appliquées. Le suréchantillonnage vise à augmenter la taille de la classe minoritaire, tandis que le sous-échantillonnage vise à réduire la taille de la classe majoritaire. Suite à l'application de ces méthodes, le nouvel ensemble de données équilibré présente une distribution de 5925 commentaires positifs, 6102 commentaires négatifs et 3255 commentaires neutres des données d'entraînements .

3) Vectorisation

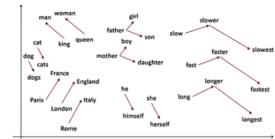
La vectorisation est le processus de transformation des données textuelles en représentations numériques, permettant ainsi aux algorithmes d'apprentissage automatique de comprendre ces données. Dans ce projet, on utilise 2 méthodes différentes de vectorisation.



• Vectorisation de TF-IDF :

Le terme fréquence de document à fréquence inverse(tf-idf) [3] est une mesure de la façon dont un mot donné est concentrés dans relativement peu de documents. Ce méthode est basée sur l'idée que les termes qui apparaissent plus fréquemment et de manière concentrée dans moins de les documents sont plus représentatifs du contenu dans les documents.

• Vectorisation par Word2Vec :

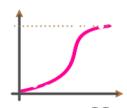


La vectorisation Word2Vec [4] est une méthode puissante dans le domaine de l'analyse de texte qui capture les relations sémantiques entre les mots. Contrairement à d'autres méthodes de vectorisation, Word2Vec ne se contente pas de représenter les mots en fonction de leur fréquence, mais plutôt en fonction de leur contexte d'utilisation. Il repose sur un réseau de neurones pour apprendre des représentations continues et multidimensionnelles des mots, où des mots similaires ont des vecteurs similaires.

1.3 Classification de modèle

Le rapport met en œuvre cinq modèles de classification analyser le sentiment du contexte, y compris régression logistique, machine à vecteurs de support, Naïve Classificateur Bayes, classificateur de forêt aléatoire et Gradient Boosting.

1) Régression logistique



La régression logistique [5] effectue le binaire classification en utilisant une fonction sigmoïde comme hypothèse, qui est donnée par :

$$P(y=1|x; \theta) = h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T x}}$$

Le modèle de régression logistique est formé en ajustant le paramètre θ via le maximum de vraisemblance, où le log de vraisemblance la fonction peut être représentée comme suit :

$$\ell(\theta) = \sum_{i=1}^n y^{(i)} \log h(x^{(i)}) + (1 - y^{(i)}) \log(1 - h(x^{(i)}))$$

Ensuite, θ peut être mis à jour en utilisant un gradient stochastique règle de remontée

$$\begin{aligned}\theta_j &:= \theta_j + \alpha \frac{\partial}{\partial \theta_j} \ell(\theta) \\ &:= \theta_j + \alpha (y^{(i)} - h_{\theta}(x^{(i)})) x_j^{(i)}\end{aligned}$$

[Les paramètres de régression logistique dans ce rapport :](#)

- Regularisation C : 10 .
- Penalty : l2 .
- max_iterations : 100 .
- dual : False, True .
- fit_intercept : False, True .
- solver : lbfgs ,liblinear,sag .

2) SVM :



La machine à vecteurs de support (SVM) [6] est considérée comme l'un des meilleurs algorithmes pour l'apprentissage supervisé. L'idée principale de cet algorithme est de cartographier les données d'un espace dimensionnel relativement faible

à un espace dimensionnel relativement élevé afin que les données de dimension supérieure peuvent être séparées en deux classes par un hyperplan.

Le hyperplan qui sépare les données avec un maximum la marge est appelée le classificateur de vecteurs de support, qui peut être déterminé à l'aide des fonctions du noyau afin d'éviter des calculs coûteux pour transformer les données explicitement

[Les paramètres de SVM dans ce rapport :](#)

- Regularisation C : 0.1,1,10 .
 - kernel : linear, rbf .
- On choisit meilleur paramètre :
- régularisation : 10 et kernel = 'rbf' .

3) Naïve Bayes :



L'algorithme Multinomial Naïve Bayes est utile dans le cas où les fonctionnalités x_j sont à valeurs discrètes en raison de sa simplicité et de sa facilité de mise en œuvre. Cet algorithme est basé sur l'hypothèse forte que x_i est conditionnellement indépendant étant donné y , qui est également connue sous le nom d'hypothèse Naïve Bayes (NB) . Le modèle est paramétré par $\phi_{j|y=1}$, $\phi_{j|y=0}$, and ϕ_y , ces paramètres peuvent être calculés comme suit :

$$\begin{aligned}\phi_{j|y=1} &= p(x_j = 1 | y = 1) \\ &= \frac{\sum_{i=1}^n 1\{x_j^{(i)} = 1 \wedge y^{(i)} = 1\}}{\sum_{i=1}^n 1\{y^{(i)} = 1\}}\end{aligned}$$

$$\begin{aligned}\phi_{j|y=0} &= p(x_j = 1 | y = 0) \\ &= \frac{\sum_{i=1}^n 1\{x_j^{(i)} = 1 \wedge y^{(i)} = 0\}}{\sum_{i=1}^n 1\{y^{(i)} = 0\}}\end{aligned}$$

$$\phi_y = p(y = 1) = \frac{\sum_{i=1}^n 1\{y^{(i)} = 1\}}{n}$$

Après ajustement des paramètres, la prédiction sur un nouvel échantillon avec des fonctionnalités peut être obtenu sous la forme (notons que ces calculs dans cas de deux classes positifs et négatives) :

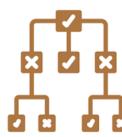
$$p(y = 1|x) = \frac{(\prod_{j=1}^d p(x_j|y = 1))p(y = 1)}{(\prod_{j=1}^d p(x_j|y = 1))p(y = 1) + (\prod_{j=1}^d p(x_j|y = 0))p(y = 0)}$$

Les paramètres de Naive Bayes dans ce rapport :

- alpha : [0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10, 100, 1000] .

→ On choisit meilleur paramètre :

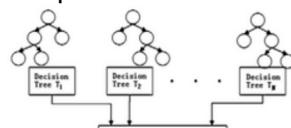
- alpha : 10 .



4) Random Forest :

La classification des arbres est très puissante pour classer les ensemble de données non linéaires, comme la PNL. Le classement comprend l'arbre en sac, la forêt aléatoire et le boosting.

La forêt aléatoire offre une amélioration par rapport à arbres ensachés. Les arbres en sac prennent en compte tous les prédicteurs (p prédicteurs) dans chaque division de l'arbre, alors que



La forêt aléatoire restreint la sélection des prédicteurs à m prédicteurs. Le nombre de prédicteurs pris en compte dans la répartition en forêt aléatoire est égal à la racine carrée du nombre total de prédicteurs, $m = \sqrt{P}$. Autrement dit, la forêt aléatoire décortille les arbres en considérant moins de prédicteurs. Contrairement aux arbres en sac hautement corrélés, la variance dans la forêt aléatoire est sensiblement réduite.

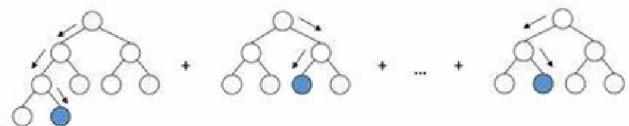
Les paramètres de Random Forest dans ce rapport :

- n_estimators : [50, 100, 200] .
- max_depth : [None, 5, 10] .

→ On choisit meilleur paramètre :

- depth : None ,
- n_estimators : 200 .

5) Gradient Boosting :



Le classificateur Boosting est une autre approche de l'arbre classification. Le boosting devient aussi une méthode pour améliorer les prédictions sur les arbres en sac. Booster les arbres poussent de manière séquentielle. Chaque arbre est cultivé en fonction sur les informations provenant d'arbres déjà cultivés, donc robuste au surapprentissage. Notamment, le boosting ne implique pas un échantillonnage bootstrap ; à la place chaque arbre s'adapter collectivement à l'arbre d'origine

Les paramètres de Gradient Boosting dans ce rapport :

- n_estimators : 100 .
- learning_rate : 0.1 .
- max_depth : 5 .
- min_sample_split : 2 .
- min_sample_leaf : 1 .
- subsample : 1 .

1.4 Test de Meilleur Modèle



On choisit le meilleur modèle qui est SVM avec $C=10$ et kernel 'rbf' ce résultat est interpréter d'après résultats dans le tableau

1) Web Scraping :

Dans le but d'améliorer la robustesse du modèle d'analyse de sentiments, On entrepris

le web scraping de nouveaux commentaires provenant de films récemment ajoutés.

Cette démarche vise à élargir la diversité des données d'entraînement en incluant des critiques plus récentes et variées. J'ai ciblé **2500** commentaires pour constituer un ensemble de test distinct, permettant ainsi d'évaluer la performance du modèle sur des données plus récentes et potentiellement changeantes.

2) Enregistrer modèle dans un fichier pickle



Afin de préserver les modèles de classification SVM et TF-IDF entraînés, j'ai opté pour l'enregistrement dans des fichiers pickle. **Pickle** est un module Python qui permet de sérialiser et désérialiser des objets, convertissant ainsi les structures de données complexes en une représentation binaire. Cette démarche offre la possibilité de sauvegarder les modèles dans un format facilement réutilisable. Les modèles entraînés ont été enregistrés dans des fichiers distincts, préservant ainsi leur structure interne, les poids et les paramètres. L'utilisation de ces fichiers pickle permet une récupération rapide et efficace des modèles, évitant la nécessité de réentraîner le modèle à chaque nouvelle utilisation.

3) Test de Modèle :



Après l'enregistrement des deux modèles dans des fichiers pickle, les modèles SVM et TF-IDF ont été testés sur un nouvel ensemble de commentaires extrait récemment. Le test a consisté à évaluer la performance de ces modèles sur des données inédites, non utilisées lors de la phase d'entraînement. Les résultats ont démontré une performance satisfaisante, avec une précision élevée dans la classification des nouveaux commentaires en termes de sentiments (positif, négatif ou neutre).

2. Résultats et Discussion:

Le rapport évalue les performances des algorithmes par la matrice de confusion. Une matrice de confusion montre le relation entre les prédictions correctes et fausses, comme indiqué dans le tableau. 1.

TABLEAU. 1. MATRICE DE CONFUSION

Etiquettes prédictives	Positive	Neutre	Négative
	Vrai Positive (TPP)	Faux Positive (FNeP)	Faux Positive (FNP)
Neutre	Faux Neutre (FPNe)	Vrai Neutre (TNeNe)	Faux Neutre (FNNe)
Negative	Faux Négative (FPN)	Faux Négative (FNeN)	Vrai Négative (TNN)

La matrice fournit plusieurs paramètres d'évaluation, y compris:

- **Précision positive** : la précision du positif prédiction.

TPP

$$\text{Précision positive} = \frac{\text{TPP}}{\text{TPP} + \text{FNeP} + \text{FNP}}$$

- la notation de Ne pour neutre et N
- signe Negative pour faire difference entre eux .

- **Précision négative** : la précision de la prédiction négative.

TNN

$$\text{Précision négative} = \frac{\text{TNN}}{\text{TNN} + \text{FPN} + \text{FNeN}}$$

- meme principe sur précision neutre

- **Précision** : le rapport des prédictions correctes, qui est la moyenne des valeurs négatives , positives, et negatives précision.

Algorithme	Paramètres	Vectorisation	Positive Précision	Négative Précision	Neutre Précision	Précision
Regression Logistique	Max_iter :100	Tf-idf	0.95	0.55	0.68	0.838
	C=10 ,penalty:l2	Word2Vec	0.92	0.33	0.57	0.686
	Penalty :l1	Tf-idf	0.95	0.61	0.72	0.858
	Class :balanced	Word2Vec	0.92	0.32	0.59	0.682
	Dual:F ,fit_inter:T	Tf-idf	0.93	0.65	0.73	0.868
	Solver :liblinear	Tf-idf	0.94	0.61	0.67	0.856
	'Solver:'sag'	Tf-idf	0.95	0.55	0.68	0.838
		Word2Vec	0.95	0.53	0.66	0.82
SVM	Kernel:'linear'	Tf-idf	0.94	0.59	0.69	0.849
	Linear,C=10	Word2Vec	0.92	0.31	0.60	0.675
	Rbf,C=10	Tf-idf	0.93	0.61	0.66	0.851
		Tf-idf	0.90	0.71	0.79	0.870
Naive Bayes	Simple	Tf-idf	0.88	0.48	0.90	0.798
	Meilleur alpha:10	Tf-idf				0.816
Random Forest	Simple	Tf-idf	0.86	0.58	0.65	0.86
		Word2Vec	0.84	0.41	0.69	0.782
	Bestparametre:	Tf-idf	0.86	0.62	0.65	0.82
	Max_depth:None	Word2Vec	0.84	0.43	0.70	0.78
Gradient Boosting	Simple	Tf-idf	0.95	0.38	0.56	0.732
		Word2Vec	0.90	0.32	0.60	0.697
	N_estimators:100	Tf-idf	0.95	0.42	0.60	0.765
	Learning_rate:0.1	Word2Vec	0.89	0.35	0.62	0.732
	Max_depth=5					
	Min_sample_split:2					
	Min_sample_leaf:1					
	Subsample:1					

TABLEAU. 2. RESULTATS DE MODELES AVEC DIFFERENTS METRIQUES

- On choisit **le meilleur modèle** qui est SVM avec C=10 et kernel 'rbf' ce resultat est interpréter d'après résultats dans le tableau.
- On remarque que les résultats avec la vectorisation **TF-IDF** est mieux que la vectorisation Word2Vec dans notre projet .

II. PARTIE 2: Systeme de Recommandations

2.1 Ensemble de données

1) Web Scraping

2.2 Nettoyage et processus

- Nettoyage
- Vectorisation

2.4 Analyse de Sentiment avec Systeme de Recommandation

2.3 Systeme de Recommandation

- Cosine
- Jacard

2.5 Deploiement d'un site web

2.1 Ensemble de données :

1) Web Scraping

Dans le cadre de cette étude, on entrepris le scraping des informations relatives aux 1000 meilleurs films [7], une liste qui englobe également les films ultérieurement utilisés pour l'analyse de sentiment. Pour cette tâche de récupération de données, j'ai opté pour l'utilisation de Beautiful Soup, une bibliothèque Python spécialisée dans l'extraction d'informations à partir de pages HTML. Cette approche a permis de collecter des détails essentiels sur chaque film, tels que les titres, les réalisateurs, les acteurs principaux, les dates de sortie et les genres.

2.1. Nettoyage et processus :

1) Nettoyage :



On fait même principe dans analyse de sentiments en plus de quelques analyse de notre données, en ajoutant quelques visualisations .

2) Vectorisation

Utilisant Tf-IDF sur la description des films

Fig. 2. Méthodologie

2.3 Système de Recommandation

1) Cosine



La similarité cosinus [8] est une mesure évaluant la similarité entre deux vecteurs en calculant le cosinus de l'angle formé par ces vecteurs dans un espace multidimensionnel. Cette méthode est couramment utilisée pour évaluer la similitude entre des éléments d'un ensemble de données en se basant sur des mots-clés ou d'autres métriques. La formule de calcul de la similarité cosinus implique le produit scalaire des vecteurs A et B, divisé par le produit de leurs magnitudes. En résumé, le score de similarité cosinus entre deux vecteurs augmente à mesure que l'angle entre eux diminue.

$$\cos(\theta) = \frac{A \cdot B}{\|A\| \cdot \|B\|} = \frac{(\sum_{i=1}^n A_i \cdot B_i)}{\sqrt{\sum_{i=1}^n (A_i)^2} \sqrt{\sum_{i=1}^n (B_i)^2}}$$

2) Jaccard :

La similarité de Jaccard [9] dans les systèmes de recommandation de films est une mesure qui évalue le degré de similarité entre deux ensembles d'articles, tels que les préférences d'utilisateurs pour des films. Elle se calcule en divisant la taille de l'intersection des ensembles par la taille de leur union. En d'autres termes, elle quantifie le nombre d'éléments communs entre deux ensembles par rapport à l'ensemble total d'articles qu'ils comprennent. Une similarité de Jaccard plus élevée suggère une plus grande similitude entre les préférences d'utilisateurs en matière de films.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

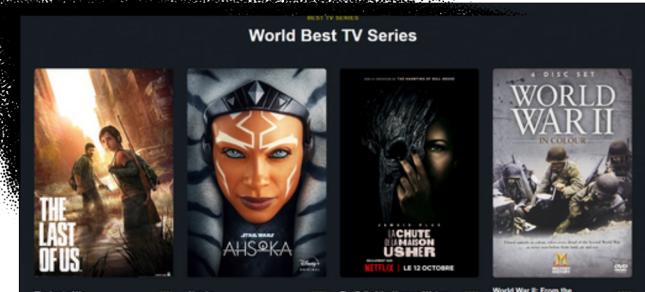
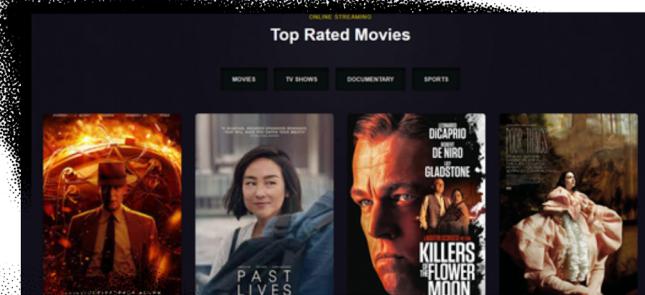
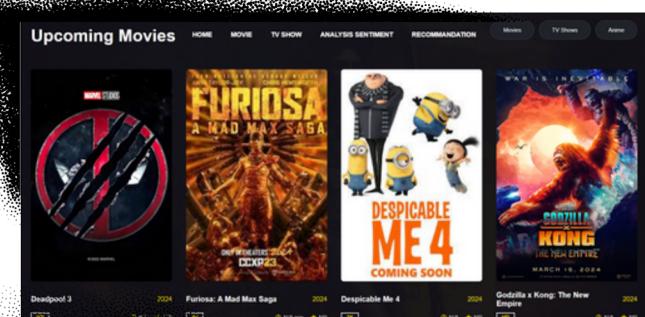
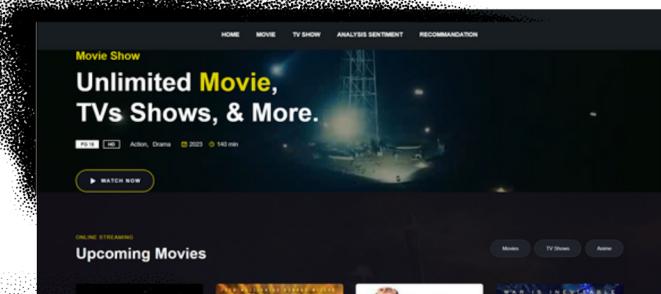
2.4 Analyse de Sentiment avec Système de Recommandation

Intégrer un système de recommandation avec une analyse de sentiments des films permet d'améliorer la pertinence des recommandations en tenant compte des commentaires des utilisateurs. En évitant de recommander des films ayant reçu des commentaires négatifs, le système s'adapte aux préférences des utilisateurs en excluant les options mal évaluées.

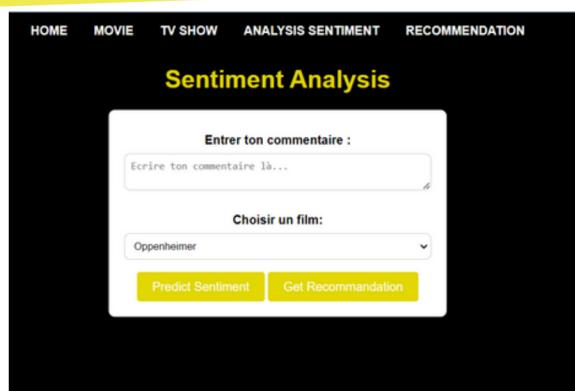
2.5 Déploiement d'un site web

<http://>

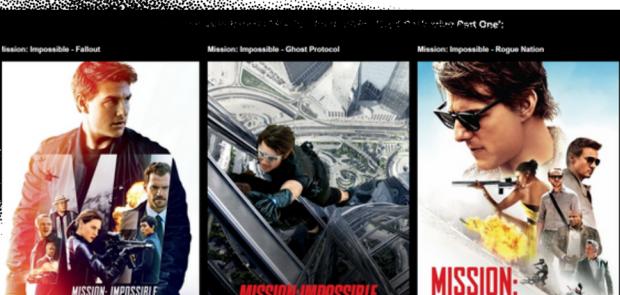
Le déploiement d'un site web dédié aux films offre une expérience enrichie aux utilisateurs. Ce site présente une sélection des meilleurs films, accompagnée de descriptions détaillées pour chaque titre. En intégrant une fonctionnalité d'analyse de sentiments, les utilisateurs peuvent explorer les commentaires et évaluations associés à chaque film, offrant ainsi un aperçu des réactions de la communauté. De plus, le site propose un système de recommandation qui s'adapte aux préférences des utilisateurs, suggérant de choisir un des 1000 films de la liste pour afficher la recommandation.



En cliquant sur Analysis Sentiment ou bien Recommandations dans menu il va nous afficher cette interface qui est intégré avec notre projet de Machine Learning



En entrant notre commentaire et en affichant le type de sentiment ou en choisissant le film depuis la liste et en ayant une recommandation comme suit (10 films similaires au film entré):



- Outils utilisés :

Dans la conception de l'interface utilisateur, nous employons des technologies telles que HTML, CSS et JavaScript pour le développement côté client. En ce qui concerne le côté serveur, Flask est choisi comme framework back-end pour établir la connexion entre le modèle et l'interface utilisateur.

III. Limitations :

Limitations du projet :



Bien que le projet intègre avec succès l'analyse de sentiments et la recommandation de films, quelques limitations subsistent,

notamment la dépendance à la qualité des données d'entraînement, la sensibilité aux nuances linguistiques, et la nécessité continue d'optimisation des modèles pour rester à jour avec l'évolution des préférences cinématographiques.

Extensions possibles :



- Intégration d'algorithmes d'apprentissage continu pour une mise à jour automatique des modèles.
- Exploration de méthodes avancées d'analyse de sentiments pour capturer des nuances linguistiques plus subtiles.
- Personnalisation accrue du système de recommandation en tenant compte des retours utilisateur et des préférences changeantes.

Conclusion :



En conclusion, notre projet a réussi à créer un modèle d'analyse de sentiments précis et un système de recommandation cinématographique efficace basé sur la similarité. Les résultats obtenus ont démontré la capacité du modèle à évaluer avec précision les réactions des utilisateurs aux critiques de films, améliorant ainsi la prédiction des sentiments associés. Malgré ces succès, des limitations subsistent, soulignant la nécessité continue d'optimisation en fonction des évolutions du langage. Les extensions possibles suggèrent des opportunités prometteuses, notamment l'intégration de l'apprentissage continu et l'exploration d'approches avancées d'analyse de sentiments. En résumé, notre projet offre une solution intégrée, conviviale et prometteuse pour l'analyse de sentiments et la recommandation cinématographique.

Références

- [1] <https://www.imdb.com/chart/top/>
- [2] **MULTILABEL OVER-SAMPLING AND UNDER-SAMPLING WITH CLASS ALIGNMENT FOR...**
Simultaneous multiple labelling of documents, also known as multilabel text classification, will not perform optimally if the class is highly imbalanced....
- [3] **A new neutrosophic TF-IDF term weighting for text mining tasks: text classification u...**
This paper aims to present a new term weighting approach for text classification as a text mining task. The original method, neutrosophic term frequency –...
- [4]

	battle	horse	king	man	queen	...	woman
authority	0	0.01	1	0.2	1	...	0.2
event	1	0	0	0	0	...	0
has tall?	0	1	0	0	0	...	0
rich	0	0.1	1	0.8	1	...	0.2
gender	0	1	1	1	1	...	1

Classification using Word2vec
In this tutorial we are going to learn how to prepare a Binary classification model using word2vec mechanism to classify the data. Also you...

[5] **A Logistic Regression Approach for Generating Movies Reputation Based on...**
IEEE Xplore®
ieeexplore.ieee.org

[6] **Analysis Sentiment Based on IMDB Aspects from Movie Reviews using SVM**
A movie is a spectacle that can be done at a relaxed time. Currently, there are many movies that can be watched via the internet or cinema. Movies that are...

[7] 
Top 1000 Highest-Grossing Movies of All Time
imdb.com