CineMatch: Système de Recommandation Personnalisé de Films

# 1. Introduction

# Dans l'ère numérique actuelle, la surabondance de choix, notamment dans le domaine du divertissement, pose un défi unique : le paradoxe de choix. Avec des milliers de films disponibles sur diverses plateformes, les utilisateurs se retrouvent souvent dépassés et incapables de décider quoi regarder. C'est ici que le besoin d'un système de recommandation personnalisé devient crucial, non seulement pour améliorer l'expérience utilisateur mais aussi pour aider les utilisateurs à découvrir des films qui correspondent vraiment à leurs goûts et préférences.

# 2. Description du Problème

# Le problème principal réside dans la capacité limitée des systèmes de recommandation actuels à fournir des suggestions hautement personnalisées. Ces systèmes ont tendance à se concentrer sur les préférences générales, ignorant souvent les critères spécifiques comme le genre, la durée, ou même les préférences culturelles. Cette approche générique conduit à des recommandations peu pertinentes, parfois avec une intention humaine, laissant les utilisateurs insatisfaits et parfois submergés par des choix peu adaptés, ou du matraquage d’idéologie.

# 3. Solution Proposée: CineMatch

# CineMatch se propose d'être une solution révolutionnaire dans le monde des recommandations de films. En combinant l'apprentissage automatique (ML) pour comprendre et apprendre les préférences des utilisateurs, et la programmation par contraintes (CSP) pour filtrer les films selon des critères spécifiques, CineMatch promet des recommandations sur mesure. L'objectif est d'offrir une expérience utilisateur enrichie et personnalisée, où chaque suggestion de film est en harmonie avec les désirs et contraintes de l'utilisateur.

# 4. Fonctionnalités Clés de CineMatch

## 4.1 Collecte et Traitement des Données

Sources des données. Méthodes de traitement.

## 4.2 Modélisation par Apprentissage Automatique

Techniques de filtrage collaboratif. Algorithmes ML utilisés.

## 4.3 Programmation par Contraintes

Types de contraintes gérées. Impact sur les recommandations.

## 4.4 Interface Utilisateur

Conception et fonctionnalités. Expérience utilisateur visée.

## 4.5 Test et Évaluation

Méthodologie de test. Métriques d'évaluation.

# 5. Technologies Utilisées

5.1 Python  
5.2 Scikit-learn/TensorFlow/Keras  
5.3 Python Constraint/Google OR-Tools  
5.4 Flask/Django

# 6. Conclusion

Résumé des contributions de CineMatch. Perspectives futures et améliorations possibles.

# 7. Références

Liste des sources bibliographiques et des références utilisées.