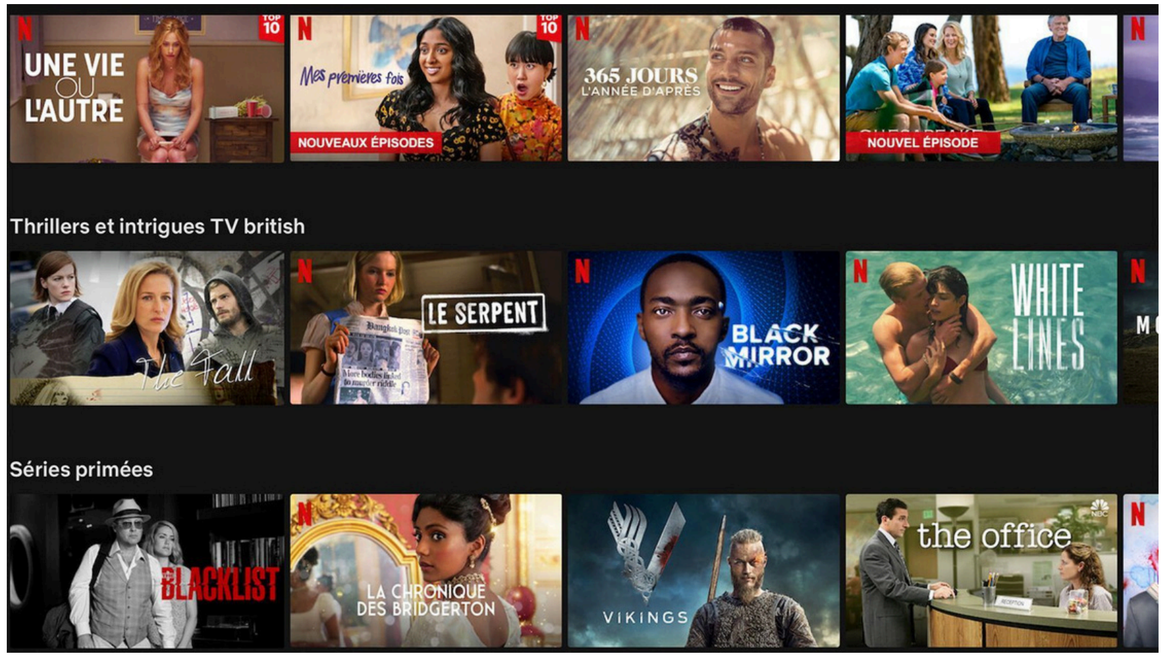
Netfloox

Interface de Streaming Vidéo

Contexte

Objectif : Développer une plateforme de streaming avec un système de recommandation et de prédiction de popularité des films



Résumé des Étapes du Projet

- **Création d'un Trello pour la gestion de projet.**
- **Récupération des données depuis des sources fournies.**
- **Insertion et exploration des données.**
- **Nettoyage, analyse et prétraitement des données.**
- **Développement des modèles :**
  - **Système de recommandation.**
  - **Système de prédiction de popularité.**
- **Conception de l’interface utilisateur.**

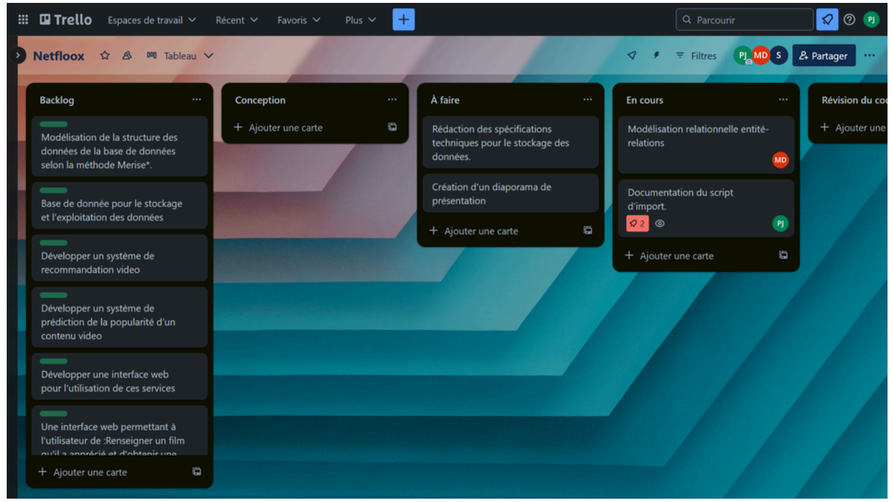
User Stories

Qui sont nos utilisateurs ?

- **Utilisateur 1 : Utilisateur Recherchant des films similaires à un film apprécié**
- **Utilisateur 2 : Producteur souhaitant estimer la popularité d’un film futur.**
- **Utilisateur 3 : Utilisateur souhaitant visualiser des données sous forme de graphiques.**

Gestion du Projet

- **Organisation via un tableau Kanban sur Trello.**
  - **Définition des tâches et suivi régulier.**
  - **Mise à jour continue du planning.**

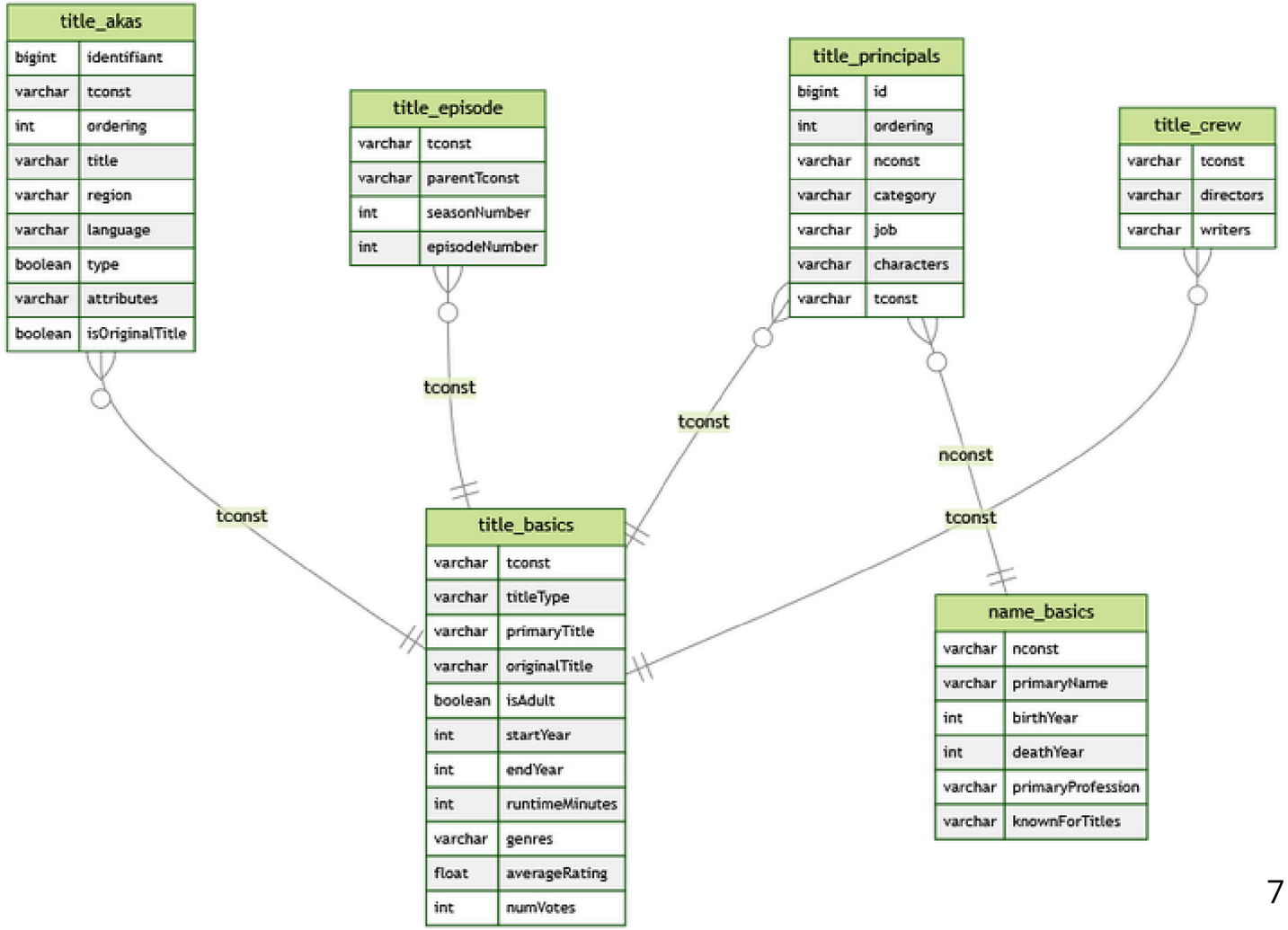


Récupération et Insertion des Données

- **Problème de données manquantes dans l’échantillon de 10 000 films.**
- **Importation du dataset complet pour garantir la cohérence des relations.**

Exploration et Analyse des Données

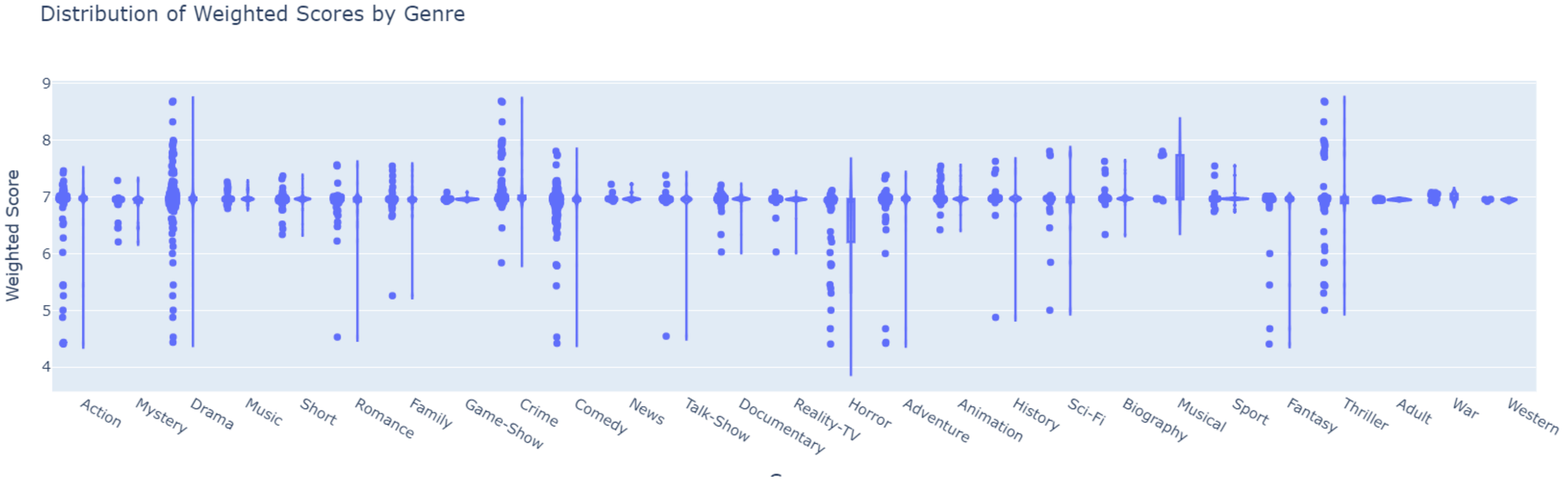
- **Compréhension des tables et de leurs infos**
- **Relations entre les tables**
- **Présence de table de jointure**
- **Conception d’un Modèle Conceptuel de Données.**
- **Identification des colonnes pertinentes et gestion des données manquantes.**



Analyse des features

$$\left(\frac{\text{votes film}}{\text{votes film} + \text{min votes}} \times \text{note film}\right) + \left(\frac{\text{min votes}}{\text{votes film} + \text{min votes}} \times \text{note moyenne}\right)$$

Exemple de graphique d'analyse de relation score pondéré - Genre



Extraction des features

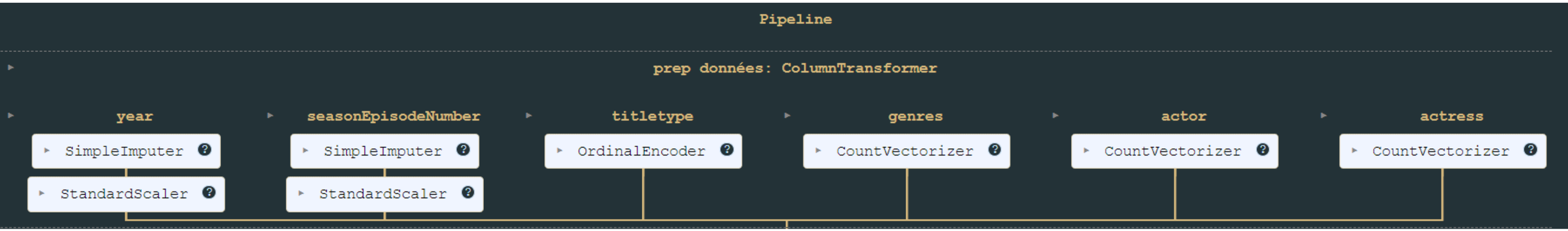
- **Extraction des 10000 films les plus récents de la table principale (*title_basics*) ayant *numvotes* et *averagerating***
- **Extraction de la table *title_episode* avec SQL et fusion avec *title_basics* avec pandas pour gérer les séries liés via *tconst* ou *parenttconst***
- **Extraction des données liés sur *title_akas* via SQL avec agrégation des régions sous 2 forme (liste et nombre) puis fusion avec pandas**
- **Extraction des données liés sur *title_principals* puis gestion de *category* et *primaryname* pour créer une colonne pour les catégories principales**

	tconst	titletype	primarytitle	isadult	startyear	genres	averagerating	numvotes	seasonnumber	episodenumber	regionnumber	regionlist	actor	self	producer	actress	director
4	tt12605348	movie	A Father's Fight	False	2021	Action,Drama,Family	4.7	3056	NaN	NaN	17.0	['\\N', 'SE', 'SG', 'TR', 'US', 'ZA', 'CA', 'I...	['Jude Mowery', 'John French', 'Chelcie Gibson...	[]	['Brandon Feller', 'Tyler Sansom', 'Ty Carter'...	['Noella Smith', 'Lindsay Rawert', 'Sarah Clev...	['Tyler Sansom']

Nettoyage et Prétraitement

- **Analyse des colonnes : *title_type*, *primary_title*, *genres*, ainsi que les acteurs, actrices et self liés aux films**
- **Transformations appliquées adaptées au contexte et à la colonnes :**
- **Pondération de la note**
- **Nettoyage des listes pour gérer les “\N”**

	tconst	titletype	primarytitle	isadult	startyear	genres	averagerating	numvotes	seasonnumber	episodenumber	regionnumber	regionlist	actor	self	producer	actress	director	weighted_score
4	tt12605348	movie	A Father's Fight	False	2021	Action,Drama,Family	4.7	3056	NaN	NaN	17.0	[SE, SG, TR, US, ZA, CA, IN, PL, US, AE, AU, D...	[]	[]	[]	[]	[]	5.303972



Système de Recommandation

- **Utilisation de *cosine_similarity* de Scikit-learn.**
- **Transformation des données via un pipeline.**
- **Construction d’une matrice de similarité entre films.**
- **Suggestion des 5 films les plus similaires.**

Système de Prédiction de Popularité

- **Évaluation initiale avec PyCaret.**
- **Modèle final : KNeighborsRegressor (meilleur R^2, MSE, MAE), sans utilisation de *PrimaryTitle***
- **Intégration du modèle dans la pipeline après prétraitement.**

Performance de notre modèle final:

- **MSE: 0.0676**
- **MAE: 0.1096**
- **R^2: 0.2463**

Conception de l’application Streamlit

- **Intégration de ces 2 systèmes dans une application Streamlit selon nos user stories**
- **Extraction des données au moment du lancement et mise en cache des données et du modèle**

Propositions d’améliorations

- **Extraire les fréquences des acteurs dans tout le dataset pour récupérer leur “popularité” et utiliser cette feature de popularité totale des acteurs pour chaque film dans les systèmes**
- **Ajout d’un input pour entrer un titletype pour le système de popularité de l’application**
- **Affinement du score pondéré en intégrant une correction logarithmique.**
- **Optimisation des performances en exploitant des techniques de vectorisation avancée. (BERT/SBERT utilisé pour rechercher de la similarité entre texte)**

Est-ce que vous avez des questions ?