**République Algérienne Démocratique et Populaire Ministère de l'Enseignement supérieur et de la Recherche scientifique**
**UNIVERSITÉ SAAD DAHLEB DE BLIDA**
**Faculté des sciences**
**Département d'informatique**

---

Mémoire de fin d'études

**Pour l'obtention du diplôme de Master en informatique**

**Option :** Ingénierie de Logiciel

---

# Conception et Réalisation d'un Moteur Intelligent de Recommandation de Programmes de Télévision

---

*Auteurs :*

BELKACEMI NOUR-ELHOUDA & BENTRIOUA AMEL

*Devant le jury composé de :*

Mme DAOUD HAYAT PRÉSIDENTE
Mme. CHIKHI IMENE EXAMINATRICE
Mme Pr. ABED HAFIDA PROMOTRICE
Mme. KHELIFI LILIA ENCADRANTE

02 Juillet 2024

# Remerciements

C'est avec un immense plaisir que nous réservons ces quelques lignes en signe de gratitude et de reconnaissance à toute personne ayant contribué de près ou de loin à l'accomplissement de ce travail.

Nous tenons tout d'abord à remercier **ALLAH** le tout puissant, qui nous a donné la force et la patience d'accomplir ce travail. En second lieu, nous tenons à remercier notre promotrice **Mme. Abed Hafida** et notre encadreur **Mme. Khelifi Lilia** de leurs précieux conseils et leur aide durant toute la période du travail la contribution en stimulant les suggestions et les encouragements, leurs patiences, leurs disponibilités, et surtout leurs sages conseils qui ont contribué à alimenter notre réflexion et à coordonner notre projet.

Nos vifs remerciements s'adressent également aux membres de jury qui ont accepté d'évaluer notre travail.

Nous tenons à exprimer notre gratitude au département d'informatique de l'Université Blida 1, pour sa contribution tout au long de ce programme de master, ainsi qu'à toute l'équipe d'**ENTV** de nous avoir donné la chance de travailler avec eux.

Nous tenons à exprimer notre gratitude à monsieur **Ibari Zahir**, **Khalil Abulokmane**, pour leur aide et pour nous avoir offert la chance de travailler avec AlWatania pour établir notre collecte d'information.

Nous tenons également à remercier le directeur du département ainsi que les enseignants, qui nous ont fourni les outils nécessaires à la réussite de nos études universitaires.

Nos remerciements vont également à nos collegues et amis, notamment **Boularas Sidahmed**, **Bouchiba Amine**, **Boubekri Fayçal** et **Remmide Karim**, pour leur disponibilité, leurs conseils et leur capacité d'écoute et d'échange d'informations.

# Dédicace

C'est avec une profonde gratitude et mots sincères que je dédie ce travail en premier lieu à Mes chères parents, quel que soit l'objet qu'on essayera de leur offrir, il n'atteindra jamais ce que j'ai envie de leur dire et exprimer.

**A ma chère MAMAN**

La perle de mon existence, quelle brave dame que tu es, que de sacrifices consentis à mon égard afin que je progresse dans mes études. Je tombe en admiration devant la bonté de ton cœur à nulle pareille.

Quels que soient mes caprices et mes écarts tu m'as toujours soutenue, trouvant les mots justes pour me ramener sur le bon chemin.

L'évènement que nous célébrons aujourd'hui t'est entièrement dédié. Que dieu te préserve santé et longue vie.

**A mon cher PAPA**

Celui qui est toujours là pour moi, et m'a donné un magnifique modèle de labeur et de persévérance, celui qui m'a encouragé qui m'a toujours protégé, mon modèle qui fait ma fierté.

Que ce travail soit l'expression des vœux que tu n'as cessé de formuler dans tes prières. Mon cher père.

A mes chères sœurs **Fatima**, **Mélissa**, **Loubna** et son époux **Oussama**.

A ma plus grande source de bonheur mon frère **Mohamed Rayan.**

A toute ma famille, mes grands-parents, mes oncles et tantes maternelles et paternelles, cousins et cousines.

A mes meilleurs amis Aziza, Hadil, Rofaida, Fayçal, Sabrina, Ilhem, Mourad, Ikram, Houda, Dalel. A ceux qui m'ont supportée, encouragée.

A tous les membres du CSC Club.

Et spécialement à ma très chère binôme **Amel**, merci pour tes conseils et tes encouragements, mais aussi pour les bons moments qui ont contribué à rendre ces années inoubliables.

**Nour-ElHouda**

# Dédicace

C'est avec une profonde gratitude et sincères mots que je dédie ce travail en premier lieu à Mes chères parents, quel que soit l'objet qu'on essayera de leur offrir, il n'atteindra jamais ce que j'ai envie de leur dire et exprimer.

**A ma chère MAMAN**

La perle de mon existence, quelle brave dame que tu es, que de sacrifices consentis à mon égard afin que je progresse dans mes études. Je tombe en admiration devant la bonté de ton cœur à nulle pareille.

Quels que soient mes caprices et mes écarts tu m'as toujours soutenue, trouvant les mots justes pour me ramener sur le bon chemin.

L'évènement que nous célébrons aujourd'hui t'est entièrement dédié. Que dieu te préserve santé et longue vie.

**A mon cher PAPA**

Celui qui est toujours là pour moi, et m'a donné un magnifique modèle de labeur et de persévérance, celui qui m'a encouragé qui m'a toujours protégé, mon modèle qui fait ma fierté.

Que ce travail soit l'expression des vœux que tu n'as cessé de formuler dans tes prières.

A ma chère sœur **Naziha** et son époux **Sidahmed**.

A mon cher frère **Reda**.

A ma plus grande source de bonheur **Layla**.

A toute ma famille, mes grands-parents, mes oncles et tantes maternelles et paternelles, cousins et cousines.

**À mon oncle et ma tante, qui nous ont quittés**

À ceux qui ont marqué nos vies de leur présence et de leur amour, même s'ils ne sont plus parmi nous physiquement, leur mémoire et leur influence continuent de vivre dans nos cœurs priant pour que Dieu les accueille dans son vaste paradis.

A mes meilleurs amis Iskander, Wassim. A ceux qui m'ont supportée, encouragée.

Et spécialement à ma très chère binôme **Nourhane**, merci pour tes conseils et tes encouragements, mais aussi pour les bons moments qui ont contribué à rendre ces années inoubliables.

**Amel**

# Resumé

Les plateformes de streaming vidéo sont devenues très populaires, cependant, le contenu Algérien est absent de ces plateformes en raison de divergences culturelles avec les contenus étrangers, des préoccupations liées à la préservation du patrimoine, et des inquiétudes quant à la protection des données, malgré l'intérêt croissant des utilisateurs pour le cinéma national.

Notre initiative vise à créer une plateforme de streaming nationale en Algérie, agrégeant les programmes de EPTV, et équipée d'un modèle de recommandation de vidéos basé sur le traitement automatique du langage, notamment le word embedding. Cette plateforme automatisera l'indexation des vidéos et offrira aux utilisateurs la possibilité de visionner leurs émissions locales préférées, accompagnées d'une expérience optimisée grâce à des recommandations personnalisées. En recommandant des films algériens, notre système contribuera à diversifier le contenu culturel accessible aux utilisateurs, tout en soutenant l'industrie cinématographique locale en mettant en avant les œuvres produites en Algérie, et en proposant un contenu unique et distinct de celui des autres plateformes de streaming.

**Mots clés :** Plateforme de streaming, modèle de traitement du langage naturel, word embedding, indexation automatique des vidéos, optimisation des recommandations personnalisées.

# Abstract

Video streaming platforms have become very popular, including in Algeria. However, Algerian content is absent from these platforms due to cultural differences with foreign content, concerns about heritage preservation, and data protection concerns, despite the growing interest of users in national cinema.

Our initiative aims to create a national streaming platform in Algeria, aggregating the programs of ENTV, and equipped with a video recommendation model based on automated language processing, including word embedding. This platform will automate the indexing of videos and offer users the opportunity to watch their favorite local programs, accompanied by an optimized experience through custom recommendations. By recommending Algerian films, our system will help to diversify the cultural content accessible to users, while supporting the local film industry by highlighting works produced in Algeria, and offering unique and distinct content from that of other streaming platforms.



**Keywords :** Streaming platform, natural language processing model, word embedding, automatic video indexing, optimization of custom recommendations.

# ملخص

أصبحت منصات بث الفيديو شديدة الشعبية، بما في ذلك في الجزائر. ومع ذلك، فإن المحتوى الجزائري غائب عن هذه المنصات بسبب الاختلافات الثقافية مع المحتوى الأجنبي، والمخاوف بشأن الحفاظ على التراث، ومخاوف حول حماية البيانات، على الرغم من ازدياد اهتمام المستخدمين بالسينما الوطنية.

تهدف مبادرتنا إلى إنشاء منصة بث وطنية في الجزائر، تجمع برامج الهيئة الوطنية للتلفزيون الجزائري، ومجهزة بنموذج توصية بالفيديو مبني على معالجة اللغة الآلية، بما في ذلك تضمين الكلمات. ستقوم هذه المنصة بتأليف فهرسة الفيديوهات تلقائياً وستقدم للمستخدمين فرصة مشاهدة برامجهم المحلية المفضلة، مصحوبة بتجربة محسنة من خلال توصيات مخصصة. بخصوص توصية أفلام جزائرية، سيساهم نظامنا في تنويع المحتوى الثقافي المتاح للمستخدمين، مع دعم الصناعة السينمائية المحلية من خلال إبراز الأعمال المنتجة في الجزائر، وتقديم محتوى فريد ومميز عن تلك المتوفرة على المنصات الأخرى لبث الفيديو.

**الكلمات المفتاحية** منصة بث، نموذج معالجة لغوية طبيعية، تضمين الكلمات، فهرسة تلقائية للفيديوهات، تحسين التوصيات المخصصة.

# Table des matières

# Table des figures

# Liste des tableaux

# Liste des abréviations

— **FC (Collaborative Filtering)** - Filtrage collaboratif
— **RMSE (Root Mean Square Error)** - Erreur quadratique moyenne
— **MAE (Mean Absolute Error)** - Erreur absolue moyenne
— **SR (Recommender System)** - Système de recommandation
— **IA (Artificial Intelligence)** - Intelligence artificielle
— **NLP (Natural Language Processing)** - Traitement du langage naturel
— **W2V (Word to Vector)** - Mot en vecteur
— **TF-IDF (Term Frequency-Inverse Document Frequency)** - Fréquence de terme-fréquence inverse de document
— **BoW (Bag of Words)** - Sac de mots
— **CBOW (Continuous Bag of Words)** - Sac de mots continu
— **kNN (k-Nearest Neighbors)** - k plus proches voisins
— **SVD (Singular Value Decomposition)** - Décomposition en valeurs singulières
— **PCA (Principal Component Analysis)** - Analyse en composantes principales
— **NMF (Non-negative Matrix Factorization)** - Factorisation en matrices non négatives
— **MVC (Model-View-Controller)** - Modèle-vue-contrôleur
— **API (Application Programming Interface)** - Interface de programmation d'applications
— **ANN (Artificial Neural Network)** - Réseau de neurones artificiels
— **TP (True Positive)** - Vrai Positif
— **TN (True Negatve)** - Vrai Negatif
— **FP (False Positive)** - Faux Positif
— **FN (False Negative)** - Faux Negatif

# Introduction Générale

La télévision publique Algérienne, en tant qu'institution culturelle majeure, détient un trésor audiovisuel composé de plus de 10 000 programmes diversifiés couvrant une multitude de catégories. Ces programmes captivent et reflètent l'identité riche et variée de la population Algérienne. En effet, au fil des décennies, la télévision publique a joué un rôle central dans la préservation et la promotion de la culture, de l'histoire et des traditions du pays. Ces émissions, allant des séries dramatiques aux documentaires en passant par les émissions de divertissement, sont bien plus que de simples programmes télévisés, elles incarnent la mémoire collective et l'expression de la diversité culturelle Algérienne. Dans ce contexte, l'analyse et la gestion de cette vaste bibliothèque audiovisuelle deviennent cruciales pour l'institution et pour le public.

## Contexte

La société publique de télévision Algérienne, connue sous le sigle EPTV, est une entité nationale chargée de superviser toutes les activités télévisuelles, de la production à la diffusion. Ses programmes couvrent une diversité de sujets, qu'ils soient nationaux, régionaux ou internationaux, abordant l'actualité, l'information du public et le divertissement à travers ses neuf chaînes : la terrestre (TV1), la principale chaîne généraliste, Canal Algerie (TV2), à dominante francophone, Al-Ikhbaria (TV3), chaîne d'information en continu, Al-Amazighia (TV4), une chaîne berbérophone, Coranique (TV5), chaîne religieuse, Al-Chababia (TV6), axée sur la jeunesse, Al-Maarifa (TV7), dédiée au savoir et à l'enseignement, Al-Dhakira (TV8), centrée sur l'histoire, et Al-Barlamania (TV9), une chaîne parlementaire. L'EPTV est une entreprise publique à caractère industriel et commercial (EPIC), dont l'État Algérien est l'unique actionnaire. Fondée en 1960 sous le nom de Radiodiffusion-télévision Algérienne (RTA), elle a été transformée en entreprise nationale de la télévision (ENTV) en 1986, puis en 1991 en entreprise publique de la télévision Algérienne (EPTV). Membre actif de l'Union de radiodiffusion des États arabes (ASBU) et de l'Union européenne de radio-télévision (UER).

### Direction des archives audiovisuelles

Cette direction a joué un rôle crucial dans l'acquisition des données essentielles pour la réalisation de notre projet. En son sein, une application locale est déployée et utilisée par les agents de collecte, de documentation et de validation. Leur mission consiste à traiter les programmes de manière exhaustive, en leur attribuant des informations signalétiques et analytiques, avant de les valider. Chaque programme est ainsi pourvu de son titre en arabe et en français, de son sous-titre en arabe et en français, de ses mots-clés, d'un résumé et d'autres détails sur les employés impliqués dans le traitement de ce programme.

## Problématique

La richesse des programmes de la télévision Algérienne demeure largement sous-exploitée, avec une grande partie de ces contenus restant méconnue du public en raison d'un accès restreint. Seuls quelques programmes sont disponibles sur des plateformes telles que YouTube, tandis que la majorité des spectateurs doivent attendre leur diffusion sur l'une des neuf chaînes, ce qui s'avère souvent décevant. Ces programmes sont pourtant disponibles dans les archives, offrant ainsi une opportunité d'exploitation et de valorisation, une pratique de plus en plus adoptée par les groupes de télévision modernes. De plus, le site de l'EPTV se concentre sur les informations et ne propose pas d'autre fonctionnalité telle que le direct. Notre solution, une plate-forme baptisée "NAVision", vise à remédier à cette situation.

Toutefois, avec un vaste catalogue de contenus, le spectateur doit trouver une sélection de programmes qui correspond le plus à ses préférences, ce qui peut s'avérer fastidieux. Pour répondre à cette problématique, l'ajout d'un moteur de recommandation devient indispensable, offrant aux utilisateurs la possibilité de découvrir des contenus similaires à ceux déjà visionnés et suggérant également de nouvelles découvertes pour enrichir leur expérience.

## Objectifs

Face à la fragmentation des contenus télévisuels Algériens et aux difficultés d'accès rencontrées par les utilisateurs, notre projet se distingue en tant que pionnier en visant à créer une plate-forme centralisée. Cette plate-forme offrira une expérience de visionnage unifiée et enrichie, répondant aux besoins et aux attentes variés des spectateurs. Nos principaux objectifs sont :

**Centraliser tous les programmes Algériens sur une seule plateforme :** cette initiative permettra de rassembler la richesse et la diversité des contenus télévisuels Algériens, offrant ainsi aux utilisateurs une destination unique pour découvrir et accéder à une variété de programmes.

**Faciliter l'accès aux programmes et à leurs rediffusions :** en mettant à disposition un catalogue complet de contenus télévisuels, notre plateforme permettra aux utilisateurs de retrouver facilement les

programmes qu'ils souhaitent regarder, y compris leurs rediffusions.

**Comprendre les besoins et les attentes des utilisateurs :** à travers des sondages, des enquêtes et des interactions avec les utilisateurs, nous chercherons à comprendre leurs préférences en termes de contenu, d'interactivité et de fonctionnalités, afin d'adapter notre plateforme à leurs besoins.

**Offrir une multitude de services en ligne :** en plus de la diffusion de programmes, notre plateforme offrira des fonctionnalités telles que la possibilité d'interagir avec d'autres utilisateurs via des commentaires, ainsi que la possibilité de télécharger des contenus pour une visualisation hors ligne.

**Assurer une diffusion télévisuelle en direct :** nous fournirons également un service de diffusion en direct, permettant aux utilisateurs de regarder leurs programmes préférés en temps réel, offrant ainsi une expérience similaire à celle de la télévision traditionnelle.

**Guider l'utilisateur dans ses choix grâce à la recommandation personnalisée :** en utilisant des algorithmes de recommandation avancés, notre plateforme proposera des programmes adaptés aux préférences de chaque utilisateur, facilitant ainsi la découverte de nouveaux contenus.

**Faire découvrir d'anciens programmes à la génération actuelle :** en mettant en avant des programmes historiques et en les intégrant dans notre catalogue, nous visons à sensibiliser la génération actuelle à l'héritage culturel et télévisuel de l'Algérie.

**Collecter de nouvelles données sur le public :** les interactions des utilisateurs avec notre plateforme nous fourniront des données précieuses sur leurs préférences et leurs habitudes de visionnage, qui seront utilisées pour améliorer notre système décisionnel et optimiser l'expérience utilisateur.

# Organisation du mémoire

La structure de notre mémoire se compose de quatre chapitres :

Le premier chapitre explore les systèmes de recommandation, le contexte général de notre travail ainsi que les travaux existants.

Le deuxième chapitre présente l'architecture globale de notre système ainsi que la conception de nos modèles de recommandation, notamment les modèles basés sur le contenu, le filtrage collaboratif et le filtrage hybride.

Le troisième chapitre décrit la conception de notre plateforme ainsi que les diagrammes UML utilisés.

Le quatrième et dernier chapitre illustre l'implémentation de notre système ainsi que les résultats obtenus pour les modèles réalisés dans chaque approche.

Enfin, nous concluons notre manuscrit par une conclusion générale et quelques perspectives.

# Chapitre 1

# Les systèmes de recommandation

## 1.1 Introduction

L'état de l'art sur les systèmes de recommandation est une étape essentielle dans la compréhension de la recherche dans ce domaine dynamique. Cette phase préliminaire offre une perspective globale des techniques et des algorithmes qui ont été largement adoptés et ont déjà prouvé leur efficacité sur de nombreuses plateformes de streaming.

Donc cette étape constitue un socle essentiel pour situer notre propre recherche, en identifiant les pistes prometteuses à explorer et les défis à relever dans ce domaine en constante évolution.

## 1.2 Les systèmes de recommandation :

Un système de recommandation est un ensemble de méthodes et d'algorithmes conçus pour proposer des éléments pertinents aux utilisateurs en fonction de leurs préférences et comportements passés. Utilisés couramment dans le commerce électronique, les services de streaming et les réseaux sociaux, ces systèmes fournissent des suggestions personnalisées de produits, de contenus ou de services. [1]

Le succès d'une plateforme de streaming repose largement sur la qualité et la diversité de ses programmes, ainsi que sur l'efficacité de son système de recommandation. Par exemple, Netflix attribue 80 % des contenus visionnés à ses recommandations. En 2015, Netflix a indiqué que la personnalisation et les recommandations combinées permettent d'économiser plus d'un milliard de dollars par an. Ainsi, un système de recommandation performant est indispensable, nécessitant une connaissance approfondie des techniques et algorithmes modernes. [2]

## 1.3 Les approches existantes des systèmes de recommandations

Pour la conception d'un moteur de recommandation, une diversité d'approches est disponible, chaque approche étant caractérisée par ses propres spécificités, avantages et contraintes. Ces méthodes reposent sur des algorithmes distincts, chacun visant à fournir des recommandations pertinentes et adaptées aux besoins des utilisateurs.

La figure 1.1 que nous avons réalisée représente les différentes approches de recommandation :

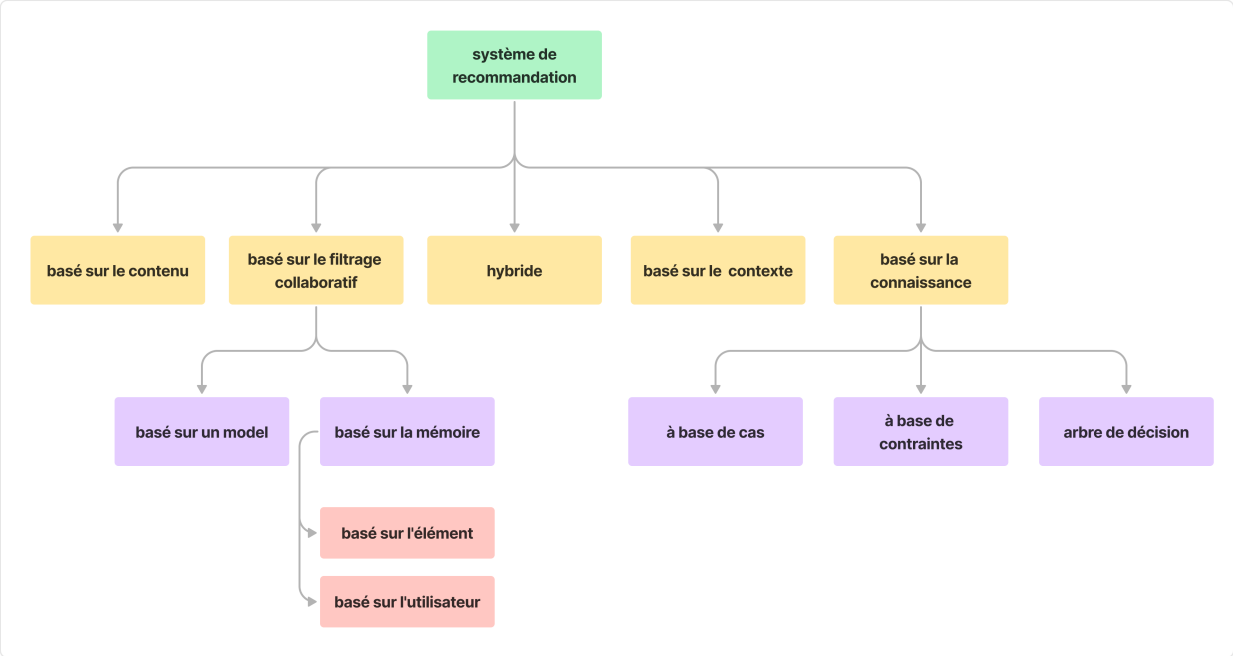


FIGURE 1.1 – Les approches des systèmes de recommandation

Nous définissons chaque approche des systèmes de recommandation comme suit :

### 1.3.1 Approche basée sur le contenu

Cette approche se focalise sur l'extraction des caractéristiques inhérentes à chaque article, telles que ses attributs, ses métadonnées ou son contenu textuel. Ensuite, elle cherche à trouver des articles similaires en comparant ces caractéristiques. L'objectif est de recommander des articles qui partagent des similitudes avec ceux que l'utilisateur a déjà appréciés. Ce processus permet de fournir des recommandations personnalisées en se basant sur les préférences individuelles des utilisateurs, sans recourir à des données d'autres utilisateurs. [3]

### 1.3.2 Approche basée sur le filtrage collaboratif

Les systèmes de recommandation par filtrage collaboratif se fondent sur les préférences d'utilisateurs similaires pour recommander des éléments. Contrairement aux systèmes basés sur le contenu, ils ne se concentrent pas sur le contenu des articles, mais sur les relations entre utilisateurs et articles, en exploitant les données historiques des interactions [4] . Il existe deux types principaux :

#### a. Filtrage collaboratif basé sur la mémoire

Recommande en se basant sur les similitudes entre utilisateurs ou éléments, où l'algorithme KNN peut être appliqué.

— **Plus proche voisin (KNN)** : L'algorithme des k plus proches voisins (KNN) est une technique d'apprentissage supervisé utilisée pour la classification et la régression. Il fonctionne en trouvant les k voisins les plus proches d'une nouvelle observation dans l'ensemble de données d'entraînement, puis en attribuant une étiquette ou une valeur en fonction des étiquettes ou des valeurs des voisins. [5]

#### b. Filtrage collaboratif basé sur les modèles

Utilise des techniques statistiques ou d'apprentissage automatique pour prédire les préférences des utilisateurs, incluent la factorisation de matrice, les réseaux de neurones, les réseaux bayésiens, et les arbres de décision pour le filtrage collaboratif.

— **Matrice de factorisation (MF)** : La matrice de factorisation est une représentation matricielle des interactions entre les utilisateurs et les éléments, souvent sous forme de notes attribuées par les utilisateurs aux éléments. Elle vise à décomposer une grande matrice en plusieurs matrices plus petites, afin de simplifier les calculs. En multipliant ces matrices décomposées, on peut reconstruire la matrice originale ou prédire les interactions manquantes. [6]

— **Décomposition en valeur singulière :** La décomposition en valeur singulière SVD (Singular Value Decomposition) est la technique la plus puissante de réduction de la dimensionnalité[7], Elle est largement utilisée dans les systèmes de recommandation car elle permet de capturer les relations complexes entre les utilisateurs et les éléments dans un espace de dimension réduite.

**Les modèles de la factorisation matricielle :**

| Modèle | Description | Applications | Exemple |
|---|---|---|---|
| SVD (Singular Value Decomposition) | Décompose une matrice en trois matrices : U (matrice unitaire à gauche), (matrice diagonale des valeurs singulières) et V(matrice unitaire à droite). | Réduction de dimension, compression de données, reconstruction d'images, filtrage collaboratif. | Analyse de ratings de films où la matrice représente les utilisateurs et les films, et les valeurs correspondent aux évaluations des utilisateurs pour les films. |
| PCA (Principal Component Analysis) | Utilise une technique linéaire de transformation pour projeter les données dans un nouvel espace de dimension inférieure tout en maximisant la variance des données projetées. | Réduction de dimension, visualisation de données, débruitage de données, analyse des composantes principales. | Analyse de données de marché où la matrice représente les caractéristiques des produits et les valeurs correspondent aux ventes de produits. |
| NMF (Non-Negative Matrix Factorization) | Décompose une matrice en deux matrices non négatives : une matrice de base et une matrice de coefficients. | Extraction de caractéristiques, débruitage de données, analyse de texte, séparation de sources. | Analyse de texte où la matrice représente les mots et les documents, et les valeurs correspondent au nombre d'occurrences de mots dans les documents. |

TABLE 1.1 – Tableau de comparaison entre les modèles de décomposition

### 1.3.3 Approche hybrides

Les approches hybrides sont des approches qui combinent deux ou plusieurs approches de recommandation. Ces combinaisons permettent de bénéficier des avantages des approches utilisées, les méthodes d'hybridation sont couramment utilisées pour des domaines spécifiques afin d'améliorer la précision des recommandations et de surmonter les limitations et les problèmes des techniques de

filtrage de base. [8]

Burke [8] a recensé les différentes manières d'hybrider des approches de recommandation, décrites ci-dessous :

| Stratégie | Définition |
|---|---|
| Pondéré (Weighted) | Les valeurs de deux ou plusieurs systèmes de recommandation (RS) sont collectées pour générer une seule recommandation. |
| Commutation (Switching) | Le système navigue entre les systèmes de recommandation hybrides en tenant compte du cas en cours. |
| Mixte (Mixed) | Les résultats de deux ou plusieurs techniques de recommandation sont générés simultanément. Par exemple, le rang CF (3) + le rang CB (2) -> rang combiné (5). |
| Combinaison de caractéristiques (Feature combination) | Les caractéristiques provenant de différentes sources sont intégrées dans une seule technique de recommandation. |
| Cascade | La technique de recommandation en cours affine les résultats d'un deuxième système de recommandation. |
| Augmentation des caractéristiques (Feature augmentation) | Les résultats d'une technique de recommandation sont intégrés comme attributs d'entrée dans un deuxième système de recommandation. |
| Niveau méta (Meta-level) | Le modèle appris par un système de recommandation est intégré comme entrée dans un autre système de recommandation. |

TABLE 1.2 – Les stratégies de la recommandation Hybride

### 1.3.4 Approche basée sur la connaissance

Dans cette approche, les recommandations sont générées en se basant sur une modélisation explicite des préférences des utilisateurs à l'aide de règles, de taxonomies ou d'autres formes de connaissances expertes. Les systèmes de recommandation basés sur la connaissance sont particulièrement utiles dans le cas d'articles qui sont rarement achetés, tels que des maisons, des automobiles, des services financiers, voire des objets de luxe. Dans ces situations, le processus de recommandation est souvent entravé par un manque d'évaluations de produits. Ces systèmes ne se basent pas sur les évaluations

des utilisateurs, mais plutôt sur des similitudes entre les exigences des clients et les descriptions des articles, ou en utilisant des contraintes spécifiant les exigences des utilisateurs. Cette approche rend ces systèmes uniques, car elle permet aux utilisateurs de spécifier explicitement ce qu'ils veulent.[9]

### 1.3.5 Approche basée sur le contexte

Cette approche prend en compte le contexte dans lequel les recommandations sont faites, tel que le moment, le lieu et l'activité actuelle de l'utilisateur. Les systèmes de recommandation ont jusqu'à présent sous-exploité les données contextuelles. Pourtant, des informations telles que le temps, la localisation et la compagnie d'autres personnes pourraient considérablement améliorer le processus de recommandation dans divers domaines.
Exemples :

— Une application de recommandation de restaurants utilise des informations contextuelles telles que la localisation, l'heure et les préférences alimentaires de l'utilisateur.

— Une application de recommandation de musique intègre des informations contextuelles telles que la localisation, l'heure, l'activité de l'utilisateur (sport, détente, etc.) et les conditions météorologiques.

— Une application de tourisme utilise des informations contextuelles telles que la saison et la localisation (lieux touristiques) .[10]

.

## 1.4 Les défis associés à la conception d'un système de recommandation

La création d'un système de recommandation peut être une tâche difficile pour les entreprises, nécessitant une attention minutieuse à de nombreux détails, en raison de trois principaux défis.
Nous identifions les trois principaux enjeux ci-dessous :

### 1.4.1 La Sérendipité

La sérendipité désigne la capacité à faire des découvertes inattendues et bénéfiques tout en cherchant autre chose. Nous sommes constamment exposés au hasard dans notre quotidien, que ce soit en retrouvant un objet perdu en cherchant autre chose chez nous, ou en rencontrant quelqu'un d'inattendu lors de nos recherches d'amis. Bien que le hasard puisse parfois mener à des résultats décevants, il peut également nous ouvrir à de nouvelles perspectives et découvertes enrichissantes. Par conséquent, un algorithme de recommandation efficace devrait non seulement anticiper nos préférences, mais aussi

nous exposer à des choix aléatoires et objectifs, nous permettant ainsi de rester ouverts à de nouvelles expériences.[11]

### 1.4.2 La Sparsité

La sparsité se réfère à des ensembles de données ou matrices où la majorité des valeurs sont nulles ou proches de zéro, fréquente dans des domaines comme l'apprentissage automatique et la recommandation. Par exemple, Spotify propose 60 millions de chansons, mais les utilisateurs n'interagissent qu'avec une fraction infime.

Cette réalité crée des "matrices creuses", où la plupart des interactions (utilisateur × éléments) sont nulles. Les systèmes de recommandation opèrent sur ces matrices, nécessitant des algorithmes adaptés pour gérer efficacement ces données. Malgré la lenteur des opérations, la sparsité permet des économies de mémoire via des structures de données comme le format "Yale Sparse Matrix.[11]

### 1.4.3 Démarrage à froid

Le démarrage à froid représente un défi majeur dans les systèmes de recommandation, où le manque d'interactions passées entre utilisateurs et éléments peut entraver les recommandations. Les entreprises abordent ce problème de deux manières : le démarrage à froid par l'utilisateur et par l'élément.

Lorsqu'un nouvel utilisateur rejoint des plateformes comme Netflix, le manque de données sur ses préférences est un obstacle. Pour maintenir l'engagement, des stratégies telles que les essais gratuits sont souvent utilisées. Les sélections d'intérêts lors de la création du profil, comme sur Netflix, en impliquant activement l'utilisateur dans le processus.

Pour le démarrage à froid des éléments, les entreprises doivent recommander de nouveaux articles ou contenus ajoutés au catalogue. Des approches comme celle d'Airbnb, qui utilise des listes similaires géographiquement et de même gamme de prix pour créer des vecteurs moyens, sont utilisées pour résoudre ce problème et garantir une expérience utilisateur optimale. [11]

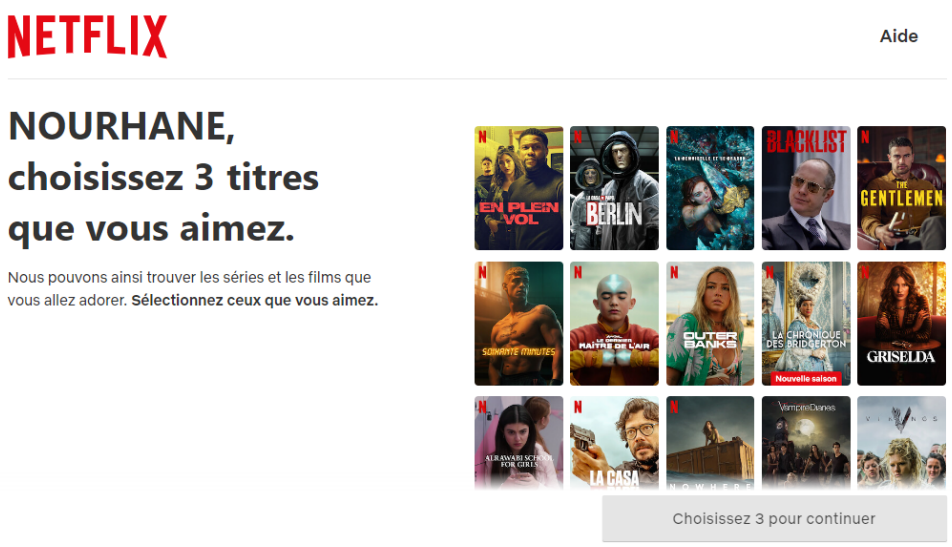


FIGURE 1.2 – Solution démarage à froid proposée par Netflix

Nous allons maintenant discuter de la représentation des mots, une étape cruciale pour transformer le langage textuel en formes numériques compréhensibles par les algorithmes de traitement automatique du langage naturel.

## 1.5 Représentation des mots

La vectorisation est une méthode d'extraction de fonctionnalités à partir du texte afin que nous puissions saisir ces fonctionnalités dans un modèle d'apprentissage automatique pour travailler avec des données textuelles. Ils essaient de préserver les informations syntaxiques et sémantiques. Les méthodes telles que Bag of Words (BOW) , CountVectorizer et TFIDF s'appuient sur le nombre de mots dans une phrase mais n'enregistrent aucune information syntaxique ou sémantique. [12]

Elle permet de transformer un mot ou ensemble de mot sous forme de vecteur. Il existe de nombreux moyens de représenter les données. Le Machine Learning s'intéresse notamment à exploiter et analyser toutes les données possibles. Les données tabulaires sont les plus simples à représenter et à analyser informatiquement. Pour traiter les données textuelles il nous faut donc une représentation mathématique plus simple. [13]

Voici un aperçu de ces techniques de représentation des mots :

### 1.5.1 Bag of Words

Une représentation bag-of-words classique sera donc celle dans laquelle on représente chaque document par un vecteur de la taille du vocabulaire |V| et on utilisera la matrice composée de l'ensemble de ces N documents qui forment le corpus comme entrée de nos algorithmes. [14]

### 1.5.2 TF-IDF

**TF « Term Frequency » :** Fréquence d'un terme dans un document. Plus un terme est fréquent dans un document plus il est important dans la description de ce document.

$$TF(i) = \frac{\log_2(freq(i,j)+1)}{\log_2(Lj)} \tag{1.1}$$

i : Terme dont le Term Frequency dans le document doit être déterminé.

j : Document analysé.

Lj : Nombre total de mots dans le document « j ».

freq(i,j) : Fréquence d'un mot « i » dans le document « j ».

log2 : Logarithme du nombre x en base 2.

**IDF « Inverse Document Frequency » :** Fréquence inverse d'un terme. Mesure l'importance d'un terme dans le corpus de documents. [15]

$$IDF(i) = log(\frac{Nd}{Fi}+1) \tag{1.2}$$

i : Terme dont l'Inverse Document Frequency doit être déterminée.

log : Logarithme du nombre x en base 10 ou en toute autre base b.

Nd : Nombre de tous les documents dans le corps du document (qui contiennent les termes pertinents).

Fi : Nombre de tous les documents dans lesquels le terme « i » apparaît.

**TF-IDF :** Etant donné que le Term Frequency représente la pertinence d'un terme dans un document donné et que l'Inverse Document Frequency peut refléter le rôle d'un terme par rapport à tous les documents d'un corpus, la combinaison des deux valeurs permet de bien comprendre la fréquence réelle des termes et le potentiel de chaque terme pour optimiser le texte existant. Pour ce faire, il suffit de multiplier les deux valeurs entre elles, ce qui donne la formule générale suivante pour l'analyse TF-IDF et la détermination d'une fréquence de termes aussi exacte et utilisable que possible [16] :

$$TF-IDF(i,j) = TF(i,j) \times IDF(i) \tag{1.3}$$

### 1.5.3 Les Word Embeddings

Le word embedding (ou plongement lexical en français) est un ensemble de techniques permettant les représentations numériques de mots dans un espace de dimension inférieure, capturant des informations sémantiques et syntaxiques. Ils jouent un rôle essentiel dans les tâches de traitement du langage naturel (NLP).

Cette ressource introduit des méthodes de représentation des mots dans l'informatique au travers d'un premier exemple simple utilisant l'occurrence des mots dans un corpus de textes puis d'un exemple utilisant Word2Vec qui regroupe un ensemble de réseaux de neurones pour l'apprentissage de vectorisation des mots afin d'opérer sur ceux-ci. Une attention particulière sera faite quant aux biais introduits par rapport aux données d'apprentissage.

### a. Word2Vec

Une technique/modèle pour produire l'intégration de mots pour une meilleure représentation des mots. Il s'agit d'une méthode de traitement du langage naturel qui capture un grand nombre de relations syntaxiques et sémantiques précises entre les mots. Il s'agit d'un réseau neuronal peu profond à deux couches qui peut détecter des mots synonymes et suggérer des mots supplémentaires pour des phrases partielles une fois formé. [17]

Maintenant que nous savons que la représentation numérique des objets aide à l'analyse en quantifiant une certaine qualité, la question est : quelle qualité des mots voulons-nous quantifier ?

La réponse à cette question est que nous voulons quantifier la sémantique. Nous voulons représenter les mots de manière à ce qu'ils capturent leur sens de la même manière que les humains. ils captent un sens contextuel du mot mais pas le sens exact. [18]

Afin de pouvoir optimiser le résultat, et bien capturer le contexte, nous devons utiliser une des deux architectures "Skip-Gram" (Sauter le gramme) ou bien "Continuous Bag of Words Model (CBOW)" (Sac continu de mots) en appliquant les réseaux de neurones.

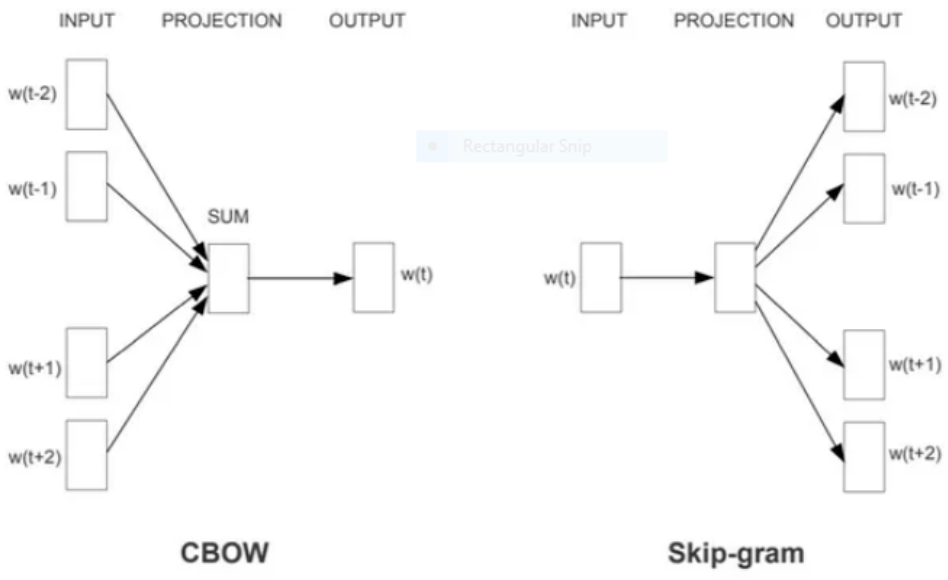


FIGURE 1.3 – Représentation graphique des deux modèles CBOW et Skip-gram [19]

Les deux architectures sont expliquées ci-dessous :

### a.1 Sac continu de mots

CBOW est un type d'architecture de réseau neuronal utilisé dans le modèle Word2Vec. L'objectif principal de CBOW est de prédire un mot cible en fonction de son contexte, constitué des mots environnants dans une fenêtre donnée. Étant donné une séquence de mots dans une fenêtre contextuelle, le modèle est entraîné à prédire le mot cible au centre de la fenêtre.

CBOW est un réseau neuronal à action directe avec une seule couche cachée. La couche d'entrée représente les mots de contexte et la couche de sortie représente le mot cible. La couche cachée contient les représentations vectorielles continues apprises (intégrations de mots) des mots d'entrée. [12]

### a.2 Saut-Gramme

Le modèle Skip-Gram apprend les représentations distribuées de mots dans un espace vectoriel continu. L'objectif principal de Skip-Gram est de prédire les mots contextuels (mots entourant un mot cible) à partir d'un mot cible. C'est l'opposé du modèle CBOW (Continuous Bag of Words), où l'objectif est de prédire le mot cible en fonction de son contexte. Il est montré que cette méthode produit des plongements plus significatifs. [12]

**Comparaison entre Skip-gram et CBOW**

Voici un tableau de comparaison entre ces deux modèles :

| | Saut-Gramme | Sac continu de mots |
|---|---|---|
| Avantages | - Skip-gram fonctionne bien pour les mots moins fréquents.<br>- Il s'agit d'un apprentissage non supervisé et peut donc fonctionner sur n'importe quel texte brut.<br>- Il nécessite moins de mémoire par rapport à d'autres algorithmes de vectorisation. | - CBOW est relativement plus rapide à entraîner.<br>- CBOW est meilleur pour les mots qui apparaissent fréquemment.<br>- CBOW est plus simple. |
| Inconvénients | - Skip-gram est plus lent.<br>- Skip-gram est plus compliqué. | - CBOW ne fonctionne pas bien pour les mots qui apparaissent rarement. |

TABLE 1.3 – Comparaison entre Skip-gram et CBOW

**b. GloVe**

le terme GloVe signifie vecteurs globaux pour la représentation des mots (Global Vectors for Word Representation) fonctionne de la même manière que Word2Vec. Alors que nous pouvons voir ci-dessus que Word2Vec est un modèle «prédictif» qui prédit un mot donné par le contexte, GLOVE apprend en construisant une matrice de cooccurrence (mots X contexte) qui compte essentiellement la fréquence d'apparition d'un mot dans un contexte. Comme il s'agira d'une matrice gigantesque, nous factorisons cette matrice pour obtenir une représentation de dimension inférieure. [20]

## 1.6 Évaluation des systèmes de recommandation

L'évaluation d'un système de recommandation implique plusieurs étapes essentielles, notamment la collecte de données d'évaluation, le choix de mesures d'évaluation appropriées, la division des données en ensembles d'entraînement et de test, et enfin l'analyse des résultats pour améliorer le système. Il existe diverses méthodes pour évaluer un système de recommandation, telles que :

### 1.6.1 Évaluation basée sur les prédictions

Cette approche évalue la précision des scores prédits par le système de recommandation en les comparant aux évaluations réelles des utilisateurs. Les mesures couramment utilisées :

**a. RMSE** : l'erreur quadratique moyenne qui est un indicateur de vérification de la fiabilité d'un modèle. Cet outil étudie les écarts entre les valeurs réellement observées et les valeurs prédites par le modèle. [21]

**b. MAE** : L'erreur moyenne absolue donne la différence absolue moyenne entre les prévisions du modèle et les valeurs cible dans Watson OpenScale. [22]

### 1.6.2 Évaluation basée sur la liste des top-N recommandations

Cette approche évalue la qualité des recommandations fournies par le système en comparant les éléments réellement recommandés aux éléments jugés pertinents pour chaque utilisateur dans l'ensemble de données de test. Les mesures couramment utilisées incluent :

**Rappel** permet de savoir le pourcentage de positifs bien prédit par notre modèle. En d'autres termes c'est le nombre de positifs bien prédit (Vrai Positif) divisé par l'ensemble des positifs (Vrai Positif + Faux Négatif). [23] Sous forme mathématique, on a :

$$Rappel = \frac{TP}{TP+FN} \tag{1.4}$$

**Précision** permet de connaître le nombre de prédictions positifs bien effectuées. En d'autres termes c'est le nombre de positifs bien prédit (Vrai Positif) divisé par l'ensemble des positifs prédit (Vrai

Positif + Faux Positif). [23] Cela nous donne sous forme mathématique :

$$Precision = \frac{TP}{TP+FP} \tag{1.5}$$

**Accuracy** permet de décrire la performance du modèle sur les individus positifs et négatifs de façon symétrique. [24] Elle mesure le taux de prédictions correctes sur l'ensemble des individus :

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1.6}$$

**F1-Score [24]** est une métrique pour évaluer la performance des modèles de classification à 2 classes ou plus. Il est particulièrement utilisé pour les problèmes utilisant des données déséquilibrées comme la détection de fraudes ou la prédiction d'incidents graves.

Le F1-score permet de résumer les valeurs de la Précision et du rappel en une seule métrique. Mathématiquement, le F1-score est défini comme étant la moyenne harmonique de la Précision et du rappel, ce qui se traduit par l'équation suivante :

$$F1-score = \frac{TP}{TP+\frac{1}{2}(FN+FP)} \tag{1.7}$$

## 1.7 Travaux Connexes

Netflix, Youtube et Amazon Prime sont les plateformes de streaming les plus connues en raison de leur architecture, de leur contenu et de leurs systèmes de recommandation. Selon les études, 80% des utilisateurs passent beaucoup de temps sur les réseaux sociaux grâce aux systèmes de recommandation. De plus, l'intelligence artificielle a récemment réussi à contrôler les êtres humains, et à les attacher à l'Internet afin d'obtenir davantage de données, de comprendre leurs intérêts et de les orienter vers les publications qui attirent leur attention.

D'après nos recherches, nous avons regroupé les plateformes repérées dans un tableau, et effectué une comparaison entre elles selon differentes critères.

Voici une brève comparaison de ces trois plateformes :

| Plate-forme | Netflix | Youtube | Amazon Prime |
|---|---|---|---|
| Objectif principal | Maximiser le temps de visionnage | Augmenter le temps passé sur la plateforme | Maximiser les achats |
| Algorithmes et méthodes utilisés | filtrage collaboratif, basés sur le contenu, la popularité et le filtrage hybrides. | basé sur le taux de visionnage des utilisateurs et les techniques de traitement du langage naturel. | filtrage collaboratif, filtrage basées sur les habitudes d'achat des utilisateurs, filtrage hybrides et des techniques d'apprentissage automatique avancées. |
| Impact financier et commercial | 80% du temps de visionnage total est attribué à son système de recommandation. | 70% du temps de visionnage sur YouTube était passé à regarder des vidéos recommandées par l'algorithme. | 35% du temps de visionnage total est attribué à son système de recommandation. |
| Contenu disponible | Propose une vaste sélection de films, séries TV, documentaires et contenus originaux produits par Netflix. | Plateforme principalement axée sur les vidéos générées par les utilisateurs, mais propose également des contenus professionnels, des émissions originales et des films moyennant un abonnement. | Offre une bibliothèque de films, séries TV et contenus originaux, ainsi que la possibilité de louer ou d'acheter des films et des émissions à la carte. |

| | | | |
|---|---|---|---|
| Modèle économique | Abonnement mensuel sans publicité offrant un accès illimité au contenu. | Disponible gratuitement avec des annonces publicitaires, offre également un abonnement payant (YouTube Premium) pour un accès sans publicité et des fonctionnalités supplémentaires. | Inclus dans l'abonnement Amazon Prime, qui offre également la livraison gratuite, la musique en streaming et d'autres avantages, ou disponible en tant qu'abonnement vidéo seul. |
| Disponibilité géographique | Disponible dans la plupart des pays du monde, bien que la bibliothèque de contenu puisse varier selon la région. | Accessible dans le monde entier, bien que certains contenus puissent être restreints dans certaines régions en raison de restrictions de licence ou de censure. | Disponible dans un nombre limité de pays, avec une disponibilité limitée de certains contenus originaux en dehors des États-Unis. |

TABLE 1.4 – Comparaison entre les plateformes de streaming

### 1.7.1 Inconvénients

Ces plateformes, bien qu'elles offrent une variété de contenus et de services, ne sont pas adaptées à l'environnement Algérien pour plusieurs raisons :

• Elles ne proposent pas de contenu spécifique à l'Algérie, ce qui limite leur pertinence pour les utilisateurs algériens.

• Les préférences et les sensibilités culturelles du public algérien peuvent différer de celles des utilisateurs typiques de ces plateformes, en particulier en ce qui concerne les contenus religieux et les langues disponibles qui complique la tâche de la recommandation.

• L'Algérie protège son patrimoine culturel et ses données, et elle veille à ne pas être associée à des contenus ou plateformes inappropriés. Par conséquent, ces plateformes peuvent ne pas répondre aux besoins et aux attentes spécifiques des utilisateurs en Algérie d'où la nécessité d'investir dans des solutions et des plateformes qui prennent en compte ces spécificités culturelles et linguistiques, tout en offrant un contenu pertinent et adapté à la population locale.

## 1.8 Conclusion

Dans ce chapitre, nous avons abordé la définition d'un système de recommandation, explorant ses concepts de base, des exemples de plateformes qui l'utilisent, ainsi que ses approches et méthodes d'évaluations.

Leurs objectifs premier est d'assister les utilisateurs dans la découverte de vidéos correspondant à leurs intérêts et préférences. Le but n'est pas seulement de proposer les contenus les plus populaires, mais plutôt d'identifier ceux qui répondent le mieux aux attentes des utilisateurs et qui sont susceptibles de les satisfaire. L'objectif est de garantir une expérience utilisateur agréable et de les inciter à passer le plus de temps possible sur la plateforme.

Nous présenterons dans le chapitre suivant la conception de cette Plate-forme ainsi que l'architecture de notre système de recommandation.

# Chapitre 2

# Concéption du moteur de recommandation

## 2.1 Introduction

L'étude effectuée dans le chapitre précèdent nous a permit d'avoir une compréhension approfondie sur les approches théoriques et pratiques nécessaires que nous devrons adopter pour développer un moteur de recommandation répondant aux besoins des utilisateurs.

Au cours de ce chapitre, nous exposerons le schéma global de notre moteur, comme nous allons expliquer toutes les phases et les étapes de notre conception.

## 2.2 Architecture du système

Nous avons déduit que notre système est basé sur une structure globale comprenant trois processus essentiels comme la figure 2.2 l'indique.

La figure 2.1 montre les entrées et les sorties du système, donc nous avons deux ensembles de données comme entrées, une base de données contient des informations sur les utilisateurs et l'autre contient des informations concernant les programmes d'EPTV, la sortie de notre système est la recommandation pour chaque utilisateur selon ses choix de visionnage.

-La première phase consiste à extraire et pré-traiter nos ensembles de données.

-La deuxième phase vise à calculer la similarité et à générer les profils similaires ainsi que les préférences des programmes de chaque utilisateur.

-Lors de la troisième et derniere phase, nous formulons la recommandation.

La figure 2.1 décrit les entées et les sorties de notre système de recommandation :


FIGURE 2.1 – Les Entrées/ Sorties de notre système

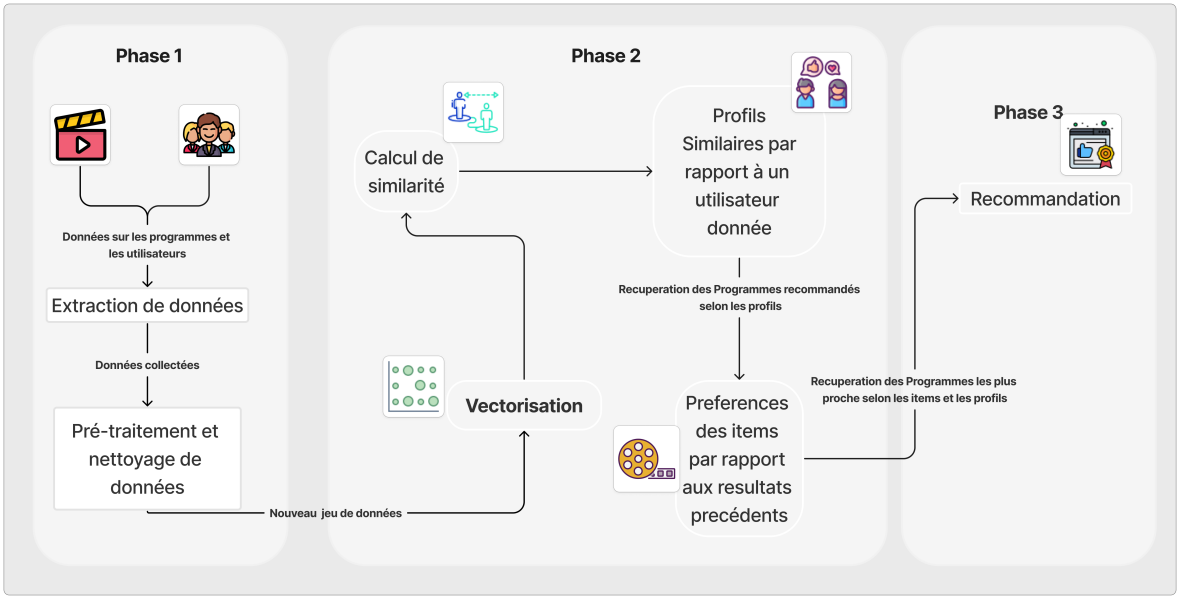


FIGURE 2.2 – Schema global de notre système de recommandation

Nous approfondissons dans chaque phase en ce qui suit :

### 2.2.1 Phase 01 : Pré-traitement

Le principal obstacle était de trouver une solution pour nettoyer nos ensembles de données, étant donné que la base de données des programmes contient plus de 80 000 lignes. Et comme nous le savons, les employés Algériens ne sont pas habitués à la gestion des données ou à l'intelligence artificielle. En général, les personnes qui remplissent les données ne sont pas des experts du domaine informatique, mais plutôt des documentalistes qui ne comprennent pas l'importance des données car leur travail c'est l'indexation, pareillement avec le deuxieme jeu de données, qui a été rempli par des gens non spécialists. C'est pourquoi nous trouvons des données non pertinentes, et c'est la raison pour laquelle l'étape de pré-traitement est l'une des étapes les plus complexes de notre projet. Le pré-traitement consiste donc à nettoyer les données, transformer les variables catégorielles en variables indicatrices, tokéniser le texte, supprimer les mots vides, lemmatiser et raciniser les tokens.

### a. Nettoyage des données

Au cours de notre processus de nettoyage des données, nous avons suivi plusieurs étapes. L'étape initiale consiste à traiter les données manquantes, en commençant par la suppression des colonnes non essentielles qui pourraient donner un faux résultat, telles que ( _"proposition_validation", "diffusé", "date_début_traitement", "date_fin_traitement", "ref_etablissement", "ref_employer_archive", "ref_validateur", "date_insertion", "date_non_validation" _...).

Par la suite, nous procéderons à ce que l'on appelle le mappage. Cela peut englober le nettoyage des données en modifiant les types de données, en supprimant les données invalides ou les doublons, en enrichissant les données ou en effectuant d'autres transformations. Dans notre situation, nous remplacerons _"ref_theme", "ref_langue", "ref_classe" par "theme", "langue" et "classe", et "duree_prog" _ par "short movie" si la durée est inférieure à une heure ou "long movie" si la durée est supérieure à une heure.

La troisième étape vise à retirer la ponctuation ainsi que le NaN qui indiquent que cette information est manquante.

Après cela, nous passons à éliminer les cases non traitées et les cases en cours de traitement.

Puis, nous avons procédé au "lowercasing" qui consiste à substituer les lettres majuscules par des lettres minuscules.

### b. Transformation des variables catégorielles en variables indicatrices

C'est une étape de prétraitement qui prend en entrée un DataFrame ou une série et crée de nouvelles colonnes binaires, une pour chaque catégorie unique dans la variable d'origine.

### c. Suppression des valeur erronées

Les valeurs erronées sont des données entrées au clavier par l'utilisateur lors de la saisie de ses préférences de programme de télévision, incluant des programmes non produits par l'EPTV.

### d. Tokénisation

Aprés avoir nettoyé les données textuelles en éliminant les informations indésirables, les erreurs et le bruit, la tokénisation est une étape essentielle du prétraitement. Elle est le processus de découpage d'un texte en unités plus petites, appelées tokens. Ces tokens sont ensuite utilisés pour l'analyse syntaxique et sémantique, permettant aux modèles d'IA de comprendre la structure et le sens du texte. La tokenization doit être précise et tenir compte des nuances de la langue, comme la ponctuation, les espaces et les caractères spéciaux, pour assurer une analyse correcte.

La tokenization est utilisée dans une variété d'applications de NLP, de la traduction automatique à l'analyse de sentiment, en passant par la reconnaissance vocale.
En transformant le texte en tokens, les modèles d'IA peuvent traiter et analyser de grandes quantités

de données textuelles de manière efficace et précise. Cela conduit à une meilleure compréhension du langage humain et à des réponses plus pertinentes et précises de la part des systèmes basés sur l'IA. [25]

Exemple a partir du ensemble de données :

La phrase en entré : "activité du président Abdelmadjid Taboune".

La phrase en sortie : ['activité', 'du', 'président', 'Abdelmadjid', 'Taboune'].

### e. Suppression des mots vides

Une fois que les données ont été nettoyées et tokenisées, Nous avons retiré les mots vides "stop-words", qui sont une liste de mots insignifiants du ensemble de données(de, des, le, les, et...).

Exemple : ['l', 'environnement', 'et', 'l', 'homme'].

Cela devient : ['environnement'. 'homme'].

### f. Lemmatisation ou Racinisation

Le but de cette étape est de ramener les tokens à leur forme de base, en utilisant soit la lemmatisation (en trouvant le lemme, c'est-à-dire la forme de base du mot), soit la racinisation (en réduisant les mots à leur racine). Dans notre exemple, "cuisine" reste inchangé car il s'agit déjà de la forme de base. [26]

Après ce processus de prétraitement, la phrase "activites president abdelmadjid Taboune" est prête à être utilisée dans le modèle de langage pour la génération de texte en langage naturel. Exemple :

Après la lemmatisation :['activité', 'président', 'Abdelmadjid', 'Taboune'].

Après la racinisation : ['activit', 'presid', 'abdelmadj', 'taboun'].

Nous prenons un autre exemple pour bien comprendre le principe de la lemmatisation et la racinisation, "environnement homme" :

Après la lemmatisation :['environnement', 'homme'].

Après la racinisation : ['environ', ' hom'].

Un autre exemple pour bien comprendre le principe de la lemmatisation et la racinisation :

La phrase en entré : "renards bruns rapides sautaient"

La phrase Après la lemmatisation : ['renard', 'brun', 'rapide', 'sauter']

La phrase Après la racinisation : ['renard', 'brun', 'rapid', 'saut']

Puisque nous avons deux jeux de données, le traitement est différent pour chacun, car ces deux bases de données ont leurs propres étapes de nettoyage. Une fois que les données ont été nettoyées, tokenisées, lemmatisées et racinées, nous passons ensuite à la deuxième phase où nous allons expliquer le processus ainsi que la réalisation de notre modèle.

### 2.2.2 Phase 02 : Application des techniques et calcul de la similarité

Notre projet consiste à sélectionner la meilleure approche dans le domaine de la recommandation. Nous avons développé trois systèmes en suivant trois approches, dont le premier est (BC), le deuxième est basé sur le (FC) et le dernier est (FH).

La figure 2.3 présente les techniques utilisés dans chaque approches des systèmes de recommandation. Nous allons expliquer chacun de ces techniques pour chaque approche ci-dessous.

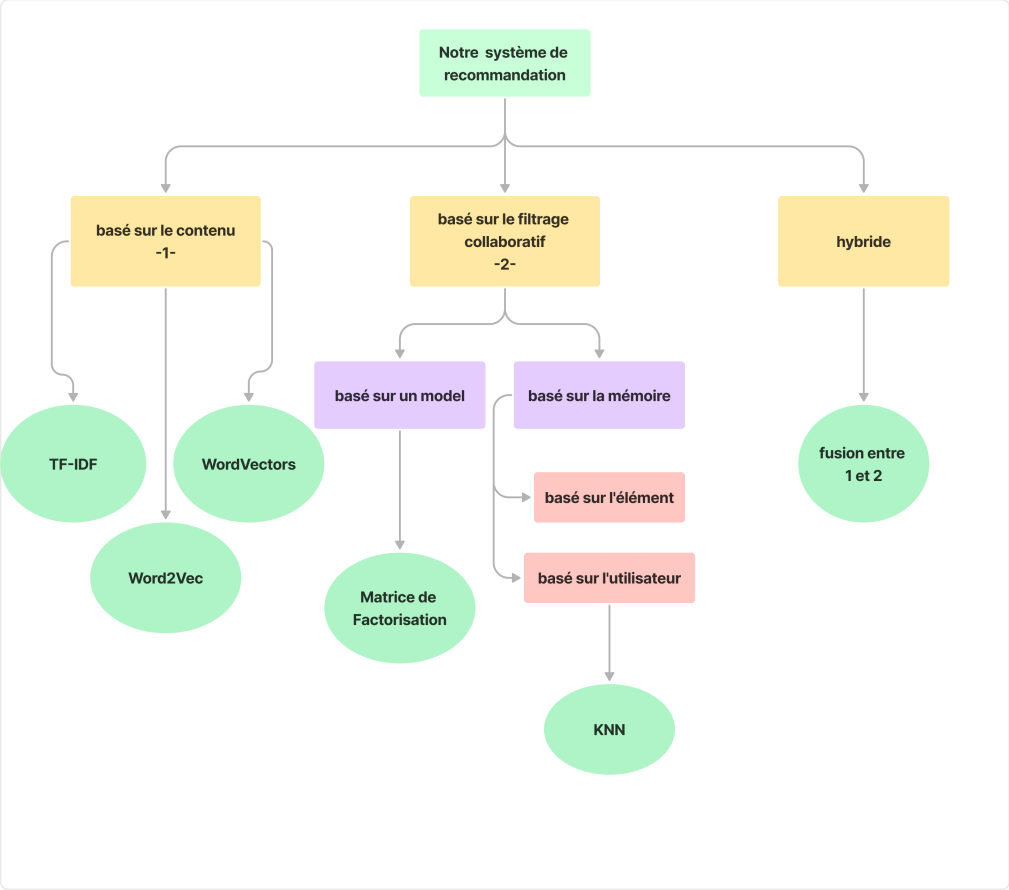


FIGURE 2.3 – Les techniques utilisées dans chaque approche

Les carrés jaunes définissent les approches existantes mentionnées dans le chapitre 1.

Les carrés mauves indiquent les types d'approches.

Les carrés roses représentent les types de la sous-approche filtrage basée sur la mémoire.

Les bulles vertes représentent les algorithmes utilisés (comme KNN) ainsi que les techniques dans chaque approche.

### a. Approche basée sur le contenu

La figure suivante exprime les techniques de vectorisation utilisés (Word2vec, TF-IDF et GloVe) dans l'approche (BC). Dont chacune a son propre nettoyage sachant que ces trois techniques sont appliqués à la même base de données, celle de l'EPTV (Programmes).

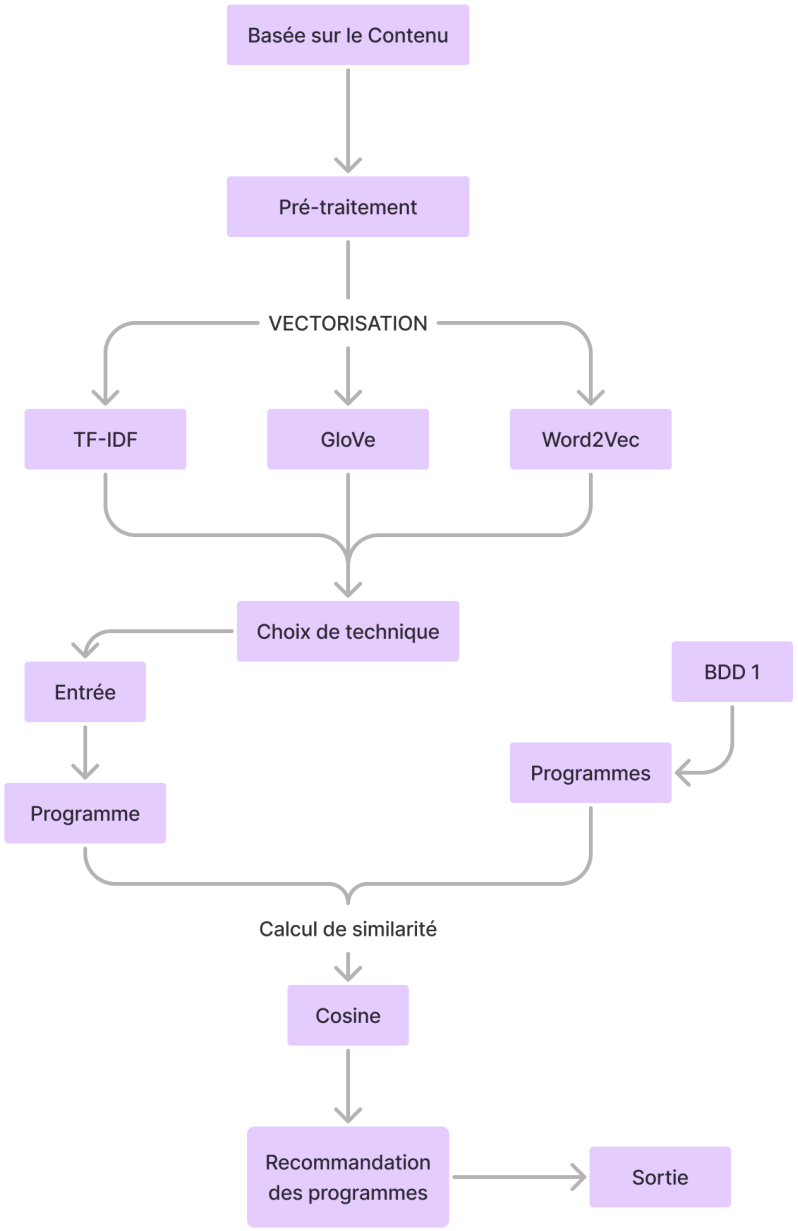


FIGURE 2.4 – Schéma global du système basé sur le contenu (BC)

## a.1 La vectorisation

La vectorisation est une étape cruciale qui permet de représenter des données textuelles sous forme de vecteurs numériques.

Dans notre cas, nous avons utilisé ces trois techniques de vectorisation (TF-IDF, W2V et GloVe) et nous allons expliquer chacune étape par étape dans les prochaines sections :

### a.1.1 Technique 1 : (TF-IDF)

**la vectorisation** se fait en construisant une matrice TF-IDF qui est une représentation numérique des documents textuels et qui représente le poids d'un terme spécifique dans un document spécifique. Par conséquent, si un terme n'apparaît pas dans un document, son poids TF-IDF sera généralement égal à zéro pour ce document.

Dans notre cas, les lignes représentent chaque un texte combiné ('combined_text' = titre + classe+ langue + thème + lieu de production + durée de programme), tandis que les colonnes représentent chaque terme unique qui apparaît dans l'ensemble de ce texte combiné « combined_text ». Cette représentation permet de mesurer la similitude entre les programmes et de les comparer dans l'espace vectoriel défini par les termes.

### a.1.2 Technique 2 : (W2V)

Word2Vec est une technique puissante pour capturer des relations sémantiques entre les mots en les représentant comme des vecteurs dans un espace vectoriel de haute dimension, où des mots similaires sont placés près les uns des autres dans l'espace vectoriel.

Dans notre cas, nous avons configuré la taille des vecteurs de mots à 100 et défini la taille de la fenêtre de contexte à 5. De plus, nous avons spécifié min_count= 5 pour exclure les mots avec une fréquence totale inférieure à ce nombre, et workers=4 pour définir le nombre de threads CPU à utiliser pendant l'entraînement. Enfin, en paramétrant sg=0, nous avons opté pour la méthode CBOW (Continuous Bag of Words) plutôt que Skip-gram.

En pratique, le choix entre CBOW et Skip-gram dépend souvent des caractéristiques spécifiques des données et de la tâche à accomplir. CBOW peut être préféré lorsque les ressources de formation sont limitées et que la capture d'informations syntaxiques est importante. Skip-gram, en revanche, pourrait être choisi lorsque les relations sémantiques et la représentation de mots rares sont cruciales. C'est pour cette raison que nous avons choisi le sac continu de mots.

Une condition préalable à tout réseau neuronal ou à toute technique d'entraînement supervisé est d'avoir des données d'entraînement étiquetées. Comment entraîner un réseau neuronal à prédire l'intégration de mots lorsque vous n'avez pas de données étiquetées, c'est-à-dire des mots et leur intégration de mots correspondants ?

Pour ce faire, nous allons créer une « fausse » tâche que le réseau neuronal doit entraîner. Nous ne nous intéresserons pas aux entrées et aux sorties de ce réseau, mais plutôt à l'objectif d'apprendre les poids de la couche cachée qui sont en fait les «vecteurs de mots» que nous essayons d'apprendre.

La fausse tâche pour le modèle CBOW serait, étant donné un mot, nous essaierons de prédire ses mots voisins. Nous allons définir un mot voisin par la taille de la fenêtre "Windows size". [18]

Le tableau suivant illustre l'enchainement de CBOW en initialisant window-size= 2 :

| Text | Entraînement de CBOW |
|---|---|
| "**concert** lounis ait manguelet" | ((lounis, ait), concert) |
| "concert **lounis** ait manguelet" | ((concert, ait, manguelet), lounis) |
| "concert lounis **ait** manguelet" | ((concert, lounis, manguelet), ait) |
| "concert lounis ait **manguelet**" | ((lounis, ait), manguelet) |

TABLE 2.1 – L'enchaînement de sac continu de mots

Compte tenu de la phrase : «concert lounis ait manguelet» et une taille de fenêtre de 2, si la phrase et les mots voisins sont ( concert, ait , manguelet) donc selon CBOW le mot cible serait "lounis".

Une fois que le modèle CBOW est entraîné, chaque mot du vocabulaire est représenté par un vecteur numérique dans un espace vectoriel. Ces vecteurs capturent les relations sémantiques et syntaxiques entre les mots en fonction de leur contexte dans les phrases.

Exemple :

Pour illustrer l'encodage, prenons le titre suivant : "concert lounis ait manguelet" .

Le principe est d'attribuer un indice à chaque mot du vocabulaire. Le vocabulaire se présente alors sous forme d'un dictionnaire : 'concert' : 0, 'lounis' : 1, 'ait' : 2, 'manguelet' : 3.

### a.1.3 Technique 3 : (GloVe)

Word2Vec et GloVe sont tous deux des algorithmes utilisés pour la création de représentations vectorielles de mots à partir de données textuelles, mais ils sont implémentés de manière légèrement différente,

GloVe utilise des mots pré-entraînées. Dans notre cas, nous utilisons la bibliothèque Gensim pour charger le modèle GloVe nommé 'glove-twitter-50'.

Ce modèle contient des vecteurs de mots de 50 dimensions pré-entraînés sur des tweets. ensuite en appliquant **la vectorisation**, nous l'avons fait en utilisant une fonction qui prend une phrase tokenisée en entrée et renvoie un vecteur de taille fixe représentant la phrase. Elle calcule le vecteur moyen de tous les vecteurs de mots présents dans la phrase en utilisant les embeddings GloVe. Dans cet algorithme, nous utilisons **les embeddings** de mots obtenus à partir de GloVe pour représenter les programmes sous forme de vecteurs numériques, puis nous utilisons ces représentations vectorielles pour effectuer une tâche de classification.

Dans notre cas, nous avons utilisé un modèle pré-entraîné à l'aide de la bibliothèque Gensim, qui occupe de la construction de la matrice de co-occurrence et de l'intégrer dans le processus d'entraînement du modèle.

Lorsque nous chargons un modèle GloVe pré-entraîné, la bibliothèque Gensim utilise les statistiques de co-occurrence incluses dans le modèle pour générer les embeddings de mots.

### a.2 Calcul de similarité

Nous avons calculé la similarité en utilisant **"cosine similarity"**, cette méthode est employée pour évaluer la similitude entre les documents. D'un point de vue mathématique, elle calcule le cosinus de l'angle formé par deux vecteurs (élément1, élément2) projetés dans un espace vectoriel de dimension N. L'avantage de la similarité cosinus est qu'elle prédit la similitude entre les documents même si la distance euclidienne est grande. [27]

$$\text{Cosine similarity}(\mathbf{X}, \mathbf{Y}) = \frac{\mathbf{X} \cdot \mathbf{Y}}{\|\mathbf{X}\| \cdot \|\mathbf{Y}\|}$$

En d'autres termes, "plus l'angle est petit, plus la similitude est grande" - c'est le principe fondamental de la similarité cosinus.

Dans ce qui suit, nous abordons l'explication de la recommandation basée sur filtrage collaboratif.

### b. Recommandation basée sur filtrage collaboratif

Dans cette approche, nous ferons appel à la deuxième base de données, renseignée via un formulaire par différents spectateurs. Le filtrage collaboratif se segmente en deux sous-approches principales : la première, basée sur la mémoire, exploite l'historique des interactions des utilisateurs pour établir des similitudes entre eux. La seconde, basée sur un modèle, recourt à des techniques algorithmiques pour anticiper les préférences des utilisateurs.

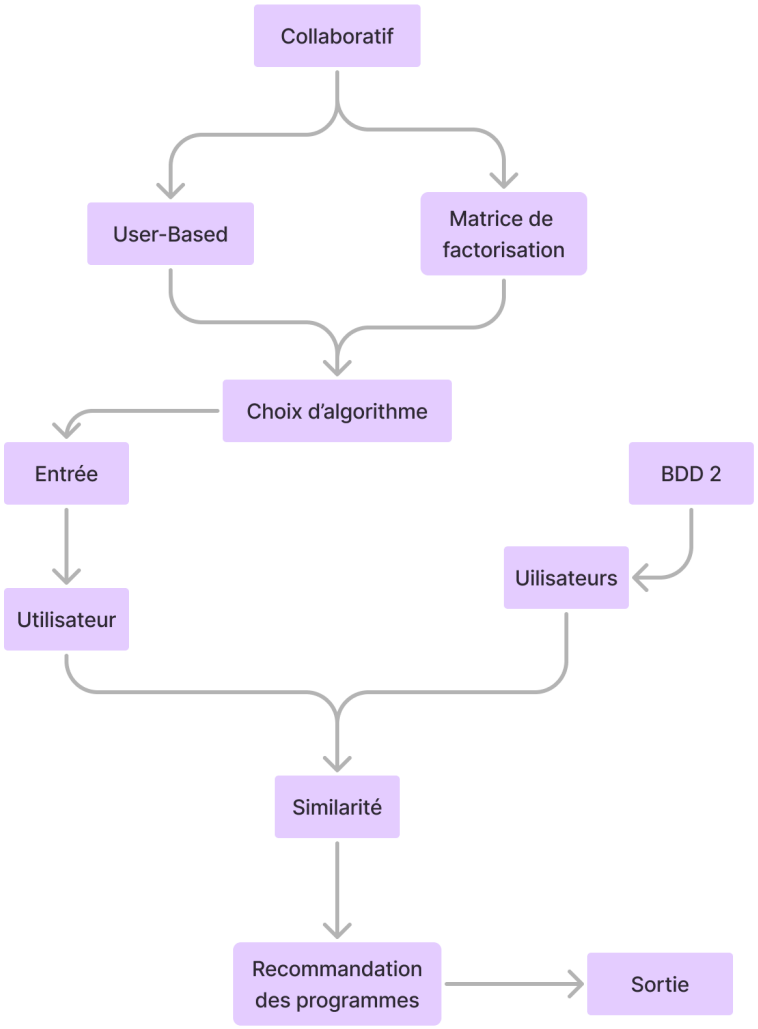


FIGURE 2.5 – Schema global du filtrage collaboratif (FC)

Puisque nous disposons de deux techniques, la première, fondée sur la mémoire, sera expliquée dans la section une où nous utilisons l'algorithme KNN. Quant à la deuxième technique, basée sur le modèle, elle sera présentée dans la deuxième section, mettant en œuvre la matrice de factorisation.

Les deux sous-approches ont été soumises à un processus de nettoyage de données identique, car cette base de données présentait moins d'imperfections par rapport à la première.

On applique la suppression des valeurs erronées entrées par l'utilisateur, la conversion des types de données, la mise en minuscules et la transformation des variables catégorielles en variables indicatrices sont effectuées.

### b.1 Technique 01 : FC basé sur la mémoire / basé sur l'utilisateur (KNN)

Afin de réaliser notre modèle basé sur la mémoire en utilisant l'algorithme (KNN), nous sommes passées par plusieurs étapes :

1. Choisir la cible à laquelle recommander de nouveaux programmes parmi tous les utilisateurs.
2. Calculer la similarité utilisateur-cible : Utiliser une mesure de similarité (par exemple, la similarité cosinus) pour évaluer la proximité entre les profils d'utilisateurs en fonction de leurs interactions passées.
3. Sélectionner les K utilisateurs les plus similaires : Identifier les K utilisateurs les plus similaires au profil de l'utilisateur cible en fonction de leur similarité.
4. Collecter les éléments appréciés par les voisins : Récupérer les éléments appréciés par les K utilisateurs les plus similaires et les classer par ordre de fréquence.
5. Soustraire les éléments vus par la cible des éléments appréciés par les voisins.
6. Recommander des éléments populaires parmi les voisins : Parmi les éléments appréciés par les voisins, recommander ceux qui sont les plus populaires ou qui correspondent le mieux aux préférences de l'utilisateur cible.
7. Fournir des recommandations personnalisées : Présenter à l'utilisateur cible une liste d'éléments recommandés basée sur les préférences de ses voisins similaires.

**Exemple**

Approche basée sur la similarité des utilisateurs : nous élaborons la matrice suivante :

| U/p | Drame | Comédie | Sportif | Religieux | P1 | P2 | P3 | P4 | P5 | P6 | P7 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Sarah | + | + | + | | + | + | | | + | | |
| Amine | | + | + | + | + | | + | | + | + | |
| Imane | + | + | | + | + | | + | | | + | |

TABLE 2.2 – Matrice (utilisateur $\times$ préférences)

En appliquant l'algorithme KNN sur ce tableau, nous pouvons utiliser les informations sur les programmes regardés par chaque utilisateur pour recommander de nouveaux programmes à un utilisateur donné.

Pour recommander des programmes à Sarah, nous calculerions les distances entre Sarah et les autres utilisateurs en fonction de leur catégorie préférée et du type de programme choisi. Ensuite, nous sélectionnerions les k utilisateurs les plus similaires à Sarah (ceux avec les distances les plus courtes) et recommanderions les programmes qu'ils ont regardés mais que Sarah n'a pas encore vus.

Dans notre exemple, les programmes potentiels qui seront recommandés sont **P3 et P6.**

### b.2 Technique 02 : FC basé sur le modèle (matrice de factorisation)

Dans un système de recommandation, nous avons souvent une matrice R où chaque ligne représente un utilisateur et chaque colonne représente un élément (programme). Les valeurs dans la matrice représentent les notes ou préférences des utilisateurs pour ces éléments. Dans notre cas, nous utilisons un système binaire où 0 signifie que l'utilisateur n'a pas regardé le programme et 1 signifie qu'il l'a regardé et éventuellement apprécié.

D'après la comparaison entre les modèles de décomposition effectuée dans le premier chapitre, nous avons choisi la décomposition en valeurs singulières (SVD).

La décomposition en valeurs singulières (SVD) d'une matrice $A$ de taille $m \times n$ est de la forme :

$$A = U\Sigma V^T$$

où $U$ et $V$ sont deux matrices orthogonales de taille $m \times m$ et $n \times n$ respectivement, et $\Sigma$ est une matrice diagonale de taille $m \times n$ qui contient les valeurs singulières de $A$ comme ses entrées diagonales. Pour simplifier, nous pouvons réduire ces matrices en considérant seulement les $k$ plus grandes valeurs singulières, où $k$ est le rang de la matrice $A$.

Dans cette forme réduite :

- $U$ devient une matrice de taille $m \times k$, - $\Sigma$ devient une matrice diagonale de taille $k \times k$, - $V^T$ devient une matrice de taille $k \times n$.

Ainsi, la SVD tronquée se concentre sur les $k$ dimensions les plus importantes, réduisant la dimensionnalité de la matrice tout en préservant l'essentiel de l'information.

**Étapes de la décomposition SVD :**

1- Création de la matrice d'interactions utilisateur-élément : Construction d'une matrice qui représente les interactions entre les utilisateurs et les éléments du système de recommandation.

2- Décomposition de la matrice par SVD : Décomposition de la matrice d'interactions en trois matrices : une matrice utilisateur-latent, une matrice élément-latent et une matrice de valeurs singulières.

3- Choix du nombre de composants latents : Spécification du nombre de dimensions à extraire pour la représentation latente des utilisateurs et des éléments.

4- Prédiction des évaluations : Calcul des évaluations prédites pour les éléments non évalués en multipliant les matrices utilisateur-latent, élément-latent et de valeurs singulières.

5- Recommandation personnalisée : Recommandation des éléments avec les évaluations prédites les plus élevées pour chaque utilisateur, en tant que suggestions les plus susceptibles d'être appréciées.

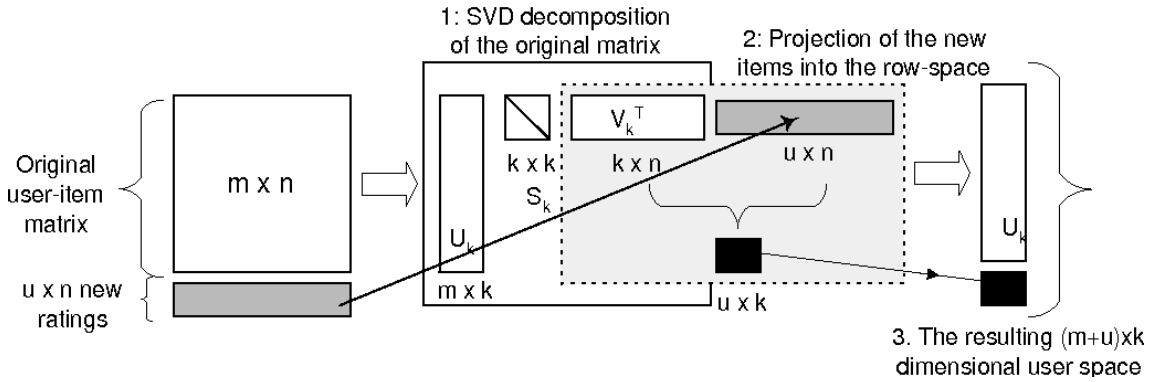


FIGURE 2.6 – Décomposition en valeur singulière

[28]

### c. Approche basée sur le filtrage hybride

Pour générer un système de recommandation hybride, nous pouvons utiliser plusieurs techniques. Et dans notre approche, nous avons choisi la fusion de la recommandation basée sur le contenu et la recommandation de filtrage collaboratif pour construire un système hybride.

Les figures suivantes expriment l'architecture du modèle hybride selon ses technique :

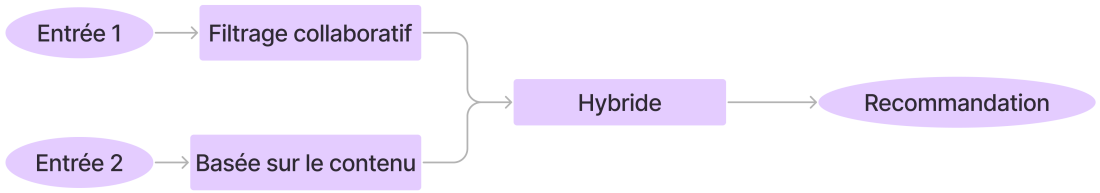


FIGURE 2.7 – Schema global du modèle Hybride (Mixte et Pondéré)

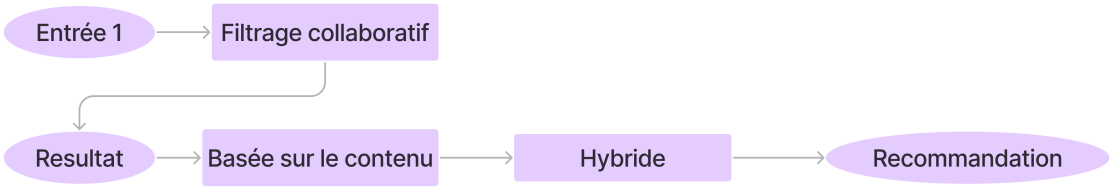


FIGURE 2.8 – Schema global du modèle Hybride (entrelacement)

Dans notre cas, nous sommes intéréssées par l'Hybride Pondéré "Weighted", l'Hybride Mixte "Mixed" et Hybride par Entrelacement "Feature Augmentation".
Le tableau suivant capture la definition de ces techniques :

| Technique | Definition |
|---|---|
| Hybride Pondéré | Les résultats de différentes techniques de recommandation (dans notre cas, le filtrage collaboratif (FC) et le filtrage basé sur le contenu (BC)) sont combinés en attribuant des poids spécifiques à chacun. |
| Hybride Mixte | Les résultats de plusieurs systèmes de recommandation sont combinés simultanément en un seul résultat final. |
| Hybride par Entrelacement | Les sorties d'un modèle du (FC) sont utilisées comme caractéristiques supplémentaires pour (CB). |

TABLE 2.3 – Les techniques de la recommandation Hybride

Dans notre approche, nous avons utilisé les trois techniques précédentes. Et au cours de ces trois techniques nous avons utilisé le résultat du filtrage collaboratif et le résultat basé sur le contenu.

**c.1 Technique 1 (Pondéré) :** La première technique consiste à affecter des poids aux résultats de FC et BC.
Par exemple : 0.4 FC + 0.6 BC = Recommandation Hybride.

**c.2 Technique 2 (mixte) :** Dans cette technique, nous avons combiné le résultat de FC et BC.
Par exemple : 2 FC + 3 BC = 5 recommandations Hybrides.

• L'idée est de prendre les avantages de plusieurs systèmes de recommandation pour fournir une meilleure recommandation globale.

**c.3 Technique 3 (par Entrelacement) :** Au cours de cette technique nous avons utilisé le résultat du FC comme entrée dans BC.
Par exemple : 5 FC –> entrée dans BC –> 10 programmes de la recommandations Hybride (5 FC + 5 (FC $\times$ BC)).
Entrée (FC) : Utilisateur 2.
Sortie (FC) : 5 programmes recommandees selon les profiles similaire à l'utilisateur 2.
Entrée (BC) : 5 programmes (résultat FC).
Sortie (BC) : les 5 Programmes les plus similaires aux 5 programmes passées en paramètre.
Sortie (FH) : 10 programmes 5 FC + 5 (FC $\times$ BC)

Ensuite, nous passons à la troisieme phase, qui est l'intégration de notre modèle,

### 2.2.3 Phase 3 : Intégration et Déploiement

L'intégration des recommandations basées sur Word2Vec dans une application suit un processus conceptuel simple.

Tout d'abord, les titres des programmes sont prétraités pour en extraire les caractéristiques pertinentes, telles que la suppression de la ponctuation, la mise en minuscules et la lemmatisation. Ensuite, un modèle Word2Vec pré-entraîné est chargé. Ce modèle associe chaque mot à un vecteur dans un espace vectoriel de haute dimension, où des mots similaires sont placés près les uns des autres en fonction de leurs relations sémantiques. Lorsqu'un utilisateur soumet le titre d'un programme, le modèle Word2Vec est utilisé pour trouver d'autres programmes présentant des similarités sémantiques avec celui-ci. Cette similarité est évaluée en mesurant la proximité dans l'espace vectoriel entre les vecteurs correspondant aux titres des programmes. Les programmes les plus similaires sont alors recommandés à l'utilisateur, lui fournissant ainsi une sélection personnalisée en fonction de ses préférences et des relations sémantiques entre les titres des programmes.

En résumé, l'intégration de Word2Vec dans le processus de recommandation permet de générer des recommandations pertinentes et personnalisées en exploitant les similitudes sémantiques entre les titres des programmes.

## 2.3 Conclusion

Dans ce chapitre nous avons presenté les architectures de chaque système et la conception de notre modèle afin de réaliser notre système. Dans le chapitre suivant, nous présentons la conception de notre plateforme ainsi que la réalisation de la partie "en direct".

# Chapitre 3

# Conception de la Plateforme

## 3.1 Introduction

La conception d'une plateforme de streaming est une tâche complexe qui nécessite une compréhension approfondie des besoins fonctionnels des utilisateurs, des principaux éléments de fonctionnalité et des différentes interactions système. Ce chapitre se concentre sur la conception d'une telle plateforme, en mettant en évidence les besoins fonctionnels essentiels, les principales fonctionnalités à implémenter et les diagrammes nécessaires pour modéliser et visualiser l'architecture système.

## 3.2 Spécification des besoins du système

Dans cette étape nous allons identifier les fonctionnalités de notre système, en déterminant les acteurs et les besoins fonctionnels du site.

### 3.2.1 Identification des acteurs

Dans notre propre conception, nous avons deux acteurs qui puisssent interagir avec le système :

— **Administrateurs :** c'est l'acteur qui utilise le système et qui peut accéder à toutes les fonctionnalités du site ainsi que la gestions de tous les programmes.

— **Spectateurs / Utilisateurs :** C'est l'acteur qui utilise le système et qui peut interagir seulement.

Les critères fonctionnels ci-dessous sont élaborés à la suite de nos études :

### 3.2.2 Les besoins fonctionnels

Afin de répondre aux besoins des utilisateurs et de créer un système concret qui répond à leurs attentes, nous avons réalisé diverses études et entretiens qui nous aideront à mieux comprendre les caractéristiques de notre système.

Le rôle de chaque acteur est décrit dans le tableau suivant :

| Acteur | Rôle |
|---|---|
| Spectateurs | - Visionner les programmes.<br>- Accéder aux programmes.<br>- Créer des profils.<br>- Modifier ses profils.<br>- Accéder aux pages en direct.<br>- Avoir des recommandations personnalisées. |
| Administrateurs | - Gérer les comptes.<br>- Gérer les programmes.<br>- Gérer les sous-programmes.<br>- Gérer les acteurs.<br>- Gérer les profils.<br>- Gérer les pages en direct. |

TABLE 3.1 – Les rôles des acteurs

## 3.3 Conception de la partie "En direct"

Dans un contexte où la consommation de médias évolue vers le streaming en direct, certaines chaînes de télévision en Algérie, telles qu'Ennahar TV et Echorouk TV, ont pris l'initiative d'embrasser le monde numérique en offrant à leur public la possibilité de regarder leurs programmes en direct sur leurs sites web. Cette démarche innovante vise à attirer davantage de spectateurs et à les fidéliser, tout en répondant aux besoins changeants d'une audience de plus en plus connectée. L'EPTV a accumulé du retard dans la mise en place du direct en raison d'un développement insuffisant, avant que le projet ne soit finalement annulé pour des raisons de confidentialité. En intégrant la transmission en direct sur son site officiel, l'EPTV pourrait renforcer son lien avec son public et s'adapter aux tendances médiatiques contemporaines, ce qui lui permettrait de maintenir sa pertinence et d'étendre son influence dans le paysage médiatique algérien.

La réalisation de notre rubrique "Streaming" a nécessité beaucoup de recherches et de déplacements, notamment au siège d'El Watania. Ces efforts ont été essentiels pour remédier à l'une des problématiques et offrir aux spectateurs de l'EPTV cette possibilité.

### 3.3.1 Solutions Proposées

Après plusieurs recherches, nous avons identifié trois méthodes qui sont les suivantes :

**Méthode 01 :** prendre le flux de base de l'entreprise à partir du nodal qui est une cellule responsable de la transmission des programmes dans les 9 chaînes puis le stocker à l'aide d'une plateforme de streaming.

**Inconvénients :**

-Le flux du nodal est confidentiel.

-Stockage non disponible au niveau de l'entreprise.

**Méthode 02 :** Une fois le direct lancé sur les réseaux sociaux de l'entreprise, nous récupérons la clé du flux et la diffusons sur notre plateforme.

**Inconvénients :**

-Le direct n'est pas continu car les plateformes telles que Facebook bloquent parfois le direct en raison de droits d'auteur que l'entreprise ne possède pas encore, même si c'est leur propre production.

-La latence est très importante, donc passer par cet intermédiaire n'est pas judicieux.

-On peut avoir qu'un seul directe à la fois et l'eptv dispose de 9 chaînes.

**Méthode 03 :** Nous récupérons le signal depuis un démodulateur satellite, utilisons deux plate formes intermédiaires pour une bonne configuration et un hébergement, puis lançons la diffusion.

**Inconvénients** :

-Nécessite une carte d'acquisition / satellite et un stockage à l'étranger qui est payant.

Et donc nous choisissons la 3ème méthode qui est la plus appropriée car la deuxième méthode est plus exposée au coupures et la première est réalisable que depuis l'entreprise.

#### Architecture de la partie en direct

Notre système de diffusion en direct repose sur une architecture efficace conçue pour offrir des diffusions en direct de haute qualité à notre public.

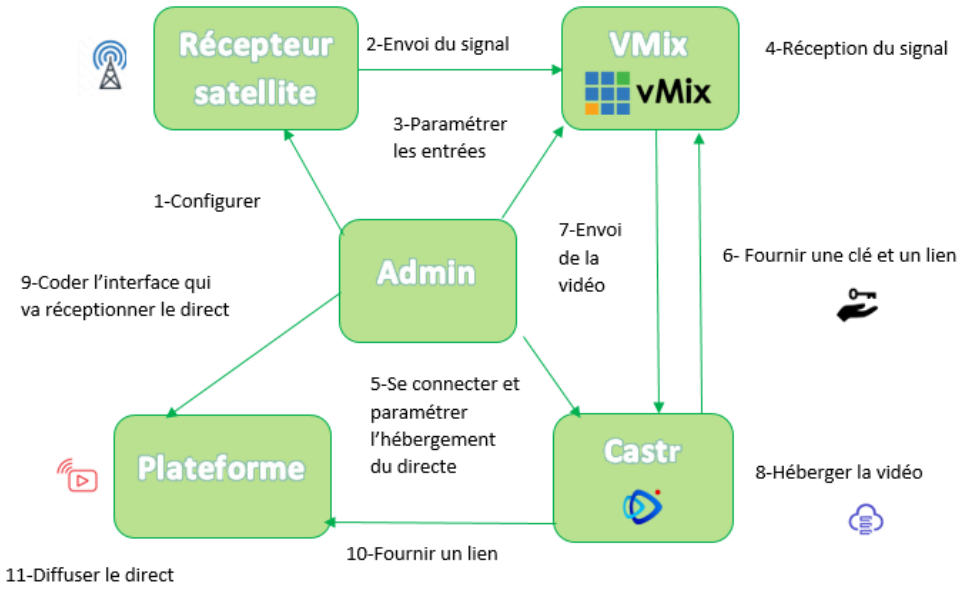


FIGURE 3.1 – Architecture de la partie en direct

Pour commencer, il est essentiel de capter le signal via un démodulateur satellite. Ensuite, VMix est utilisé pour lancer la diffusion en direct, tandis que Castr assure l'hébergement du flux en direct.

## 3.4 Conception de la plate-forme

Dans cette partie, nous présentons la conception de notre logiciel. Nous décrivons comment le système fonctionne en utilisant un langage de modélisation. Nous avons opté pour l'utilisation de la méthode UML (Unified Modeling Language), un langage de modélisation visuelle. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels aussi bien que leur comportement. [29].
Et dans notre projet nous avons utilisé :

1. Les diagrammes de modélisation statique :

— Diagramme de cas d'utilisation.

— Diagramme de classe.

2. Les diagrammes de modélisation dynamique :

— Diagramme de séquence.

### 3.4.1 Diagramme de cas d'utilisation

La première étape de l'analyse UML consiste à réaliser le diagramme des cas d'utilisation en représentant les besoins des utilisateurs. Repérant les fonctionnalités principales et les limites du système.
Les tableaux ci-dessous présentent des données essentielles concernant le cas d'utilisation.

| Terme | Description |
|---|---|
| Programme | - Série.<br>- Film.<br>- Emission. |
| Sous-Programme | - Saison.<br>- Episode.<br>- Numéro d'emission. |
| Gérer | - Ajouter.<br>- Modifier.<br>- Consulter.<br>- Supprimer. |

TABLE 3.2 – Description des termes du diagramme de cas d'utilisation

Plusieurs cas d'utilisation ont été identifiés sur notre plate-forme de streaming, et nous les avons regroupés dans le diagramme de cas d'utilisation global suivant :

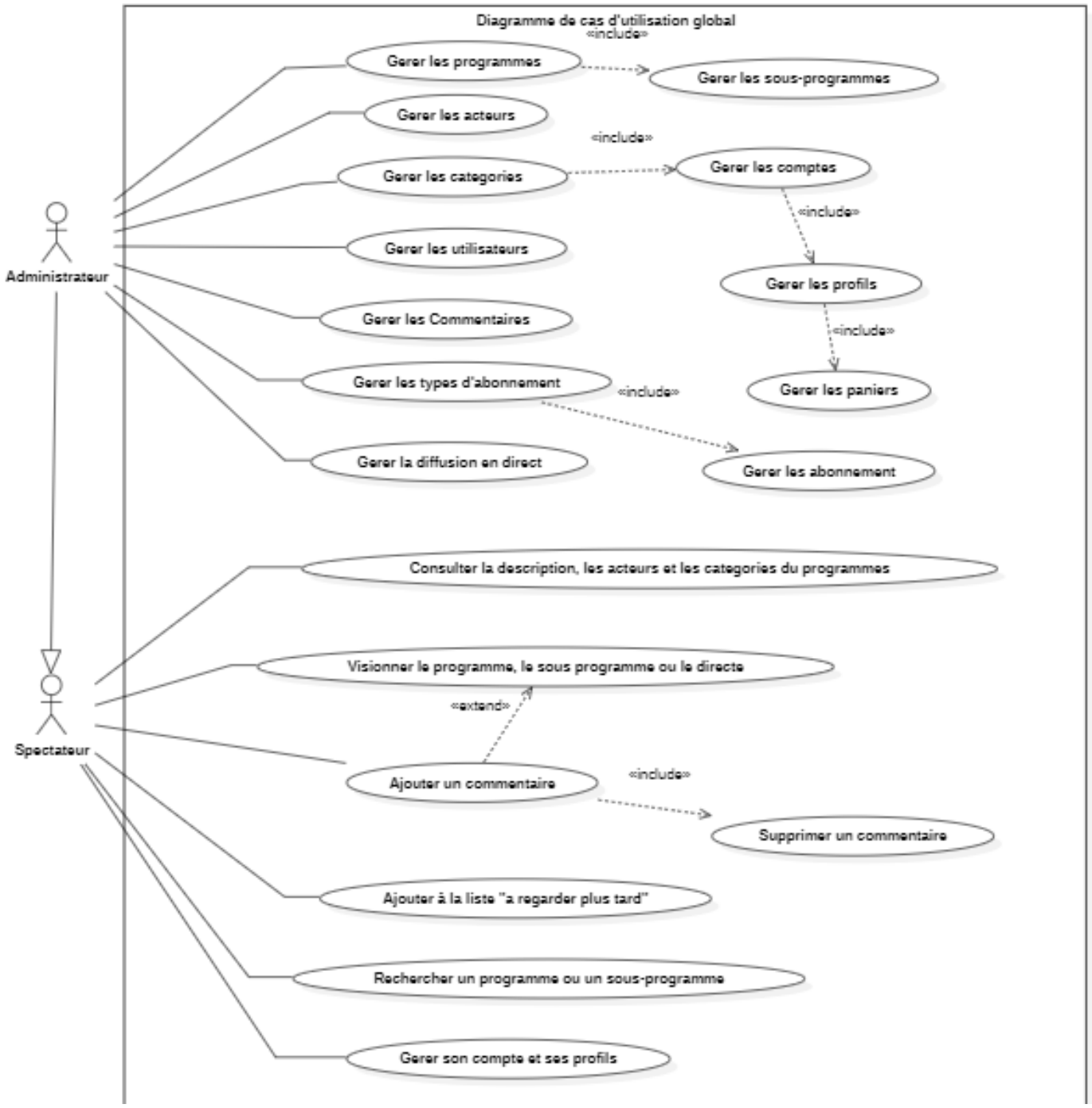


FIGURE 3.2 – Diagramme de cas d'utilisation globale

### 3.4.2 Diagramme de classe

Un diagramme de classe détaille la structure d'un système particulier en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets.
Notre diagramme de cas d'utilisation global a été realisé afin de concevoir notre diagramme de classe, ce dernier est exposé dans la figure suivante :

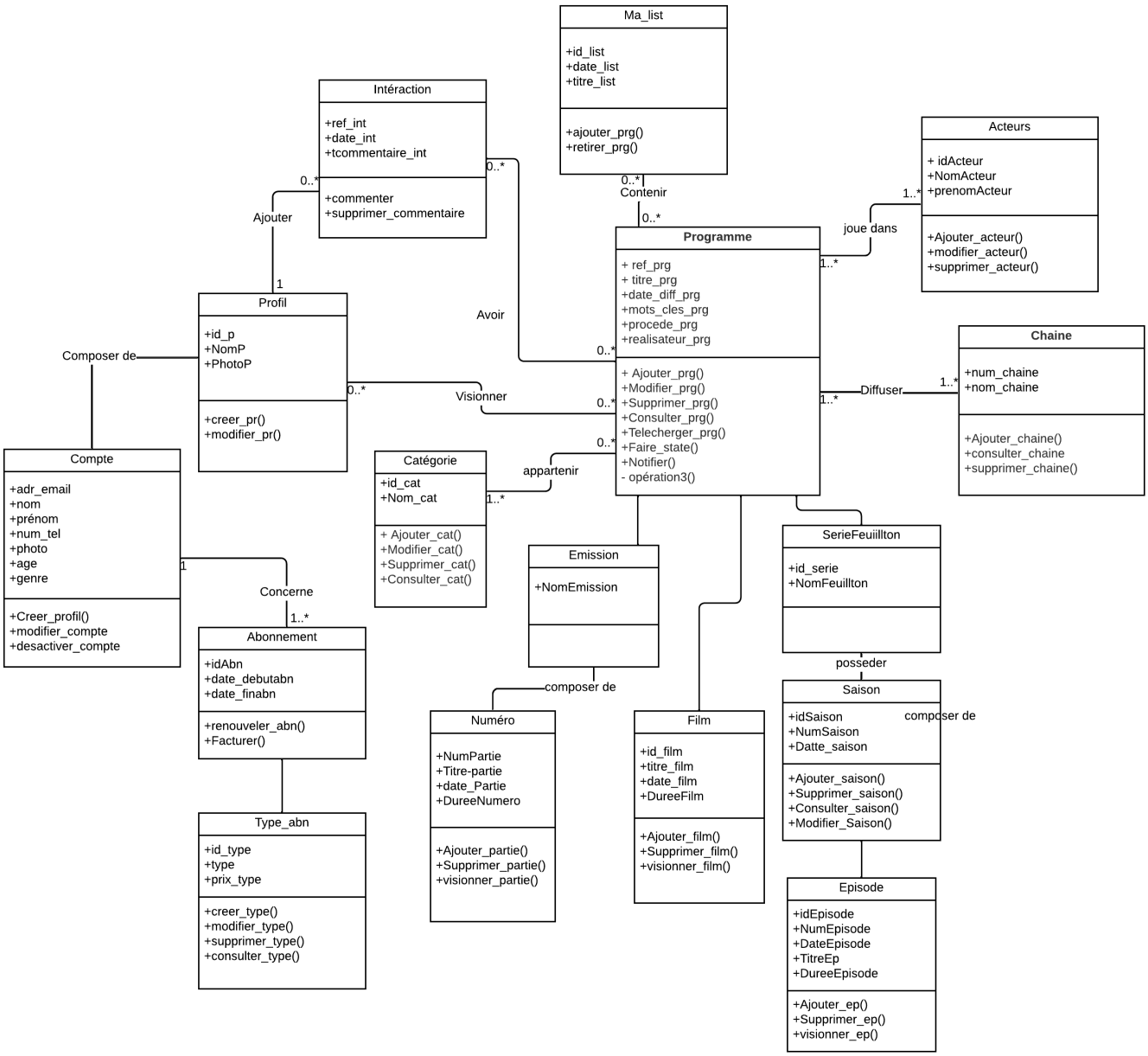


FIGURE 3.3 – Diagramme de classe

### 3.4.3 Diagramme de séquence

Les diagrammes de séquence montrent les interactions entre les objets du système. En se basant sur le diagramme de classe effectué, voici nos diagrammes de séquence :

— **La relation entre les spectateurs et les programmes :** les spectateurs ont le droit de visionner et interagir seulement.

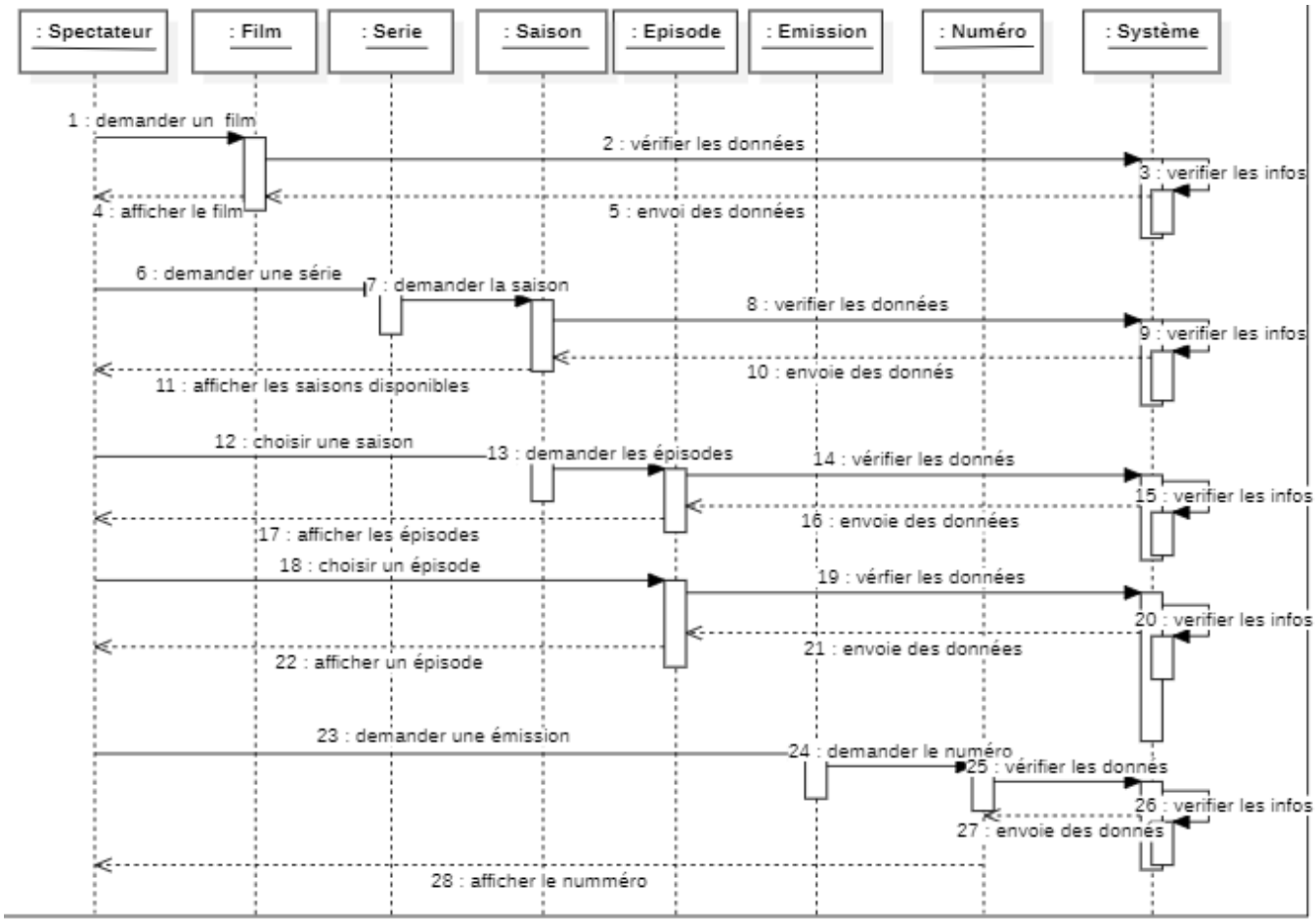


FIGURE 3.4 – Diagramme de séquence "Spectateur-Programme"

— **La relation entre les administrateurs et les programmes :** un administrateur après l'authentification, il peut gérer toutes les fonctionnalités du système.

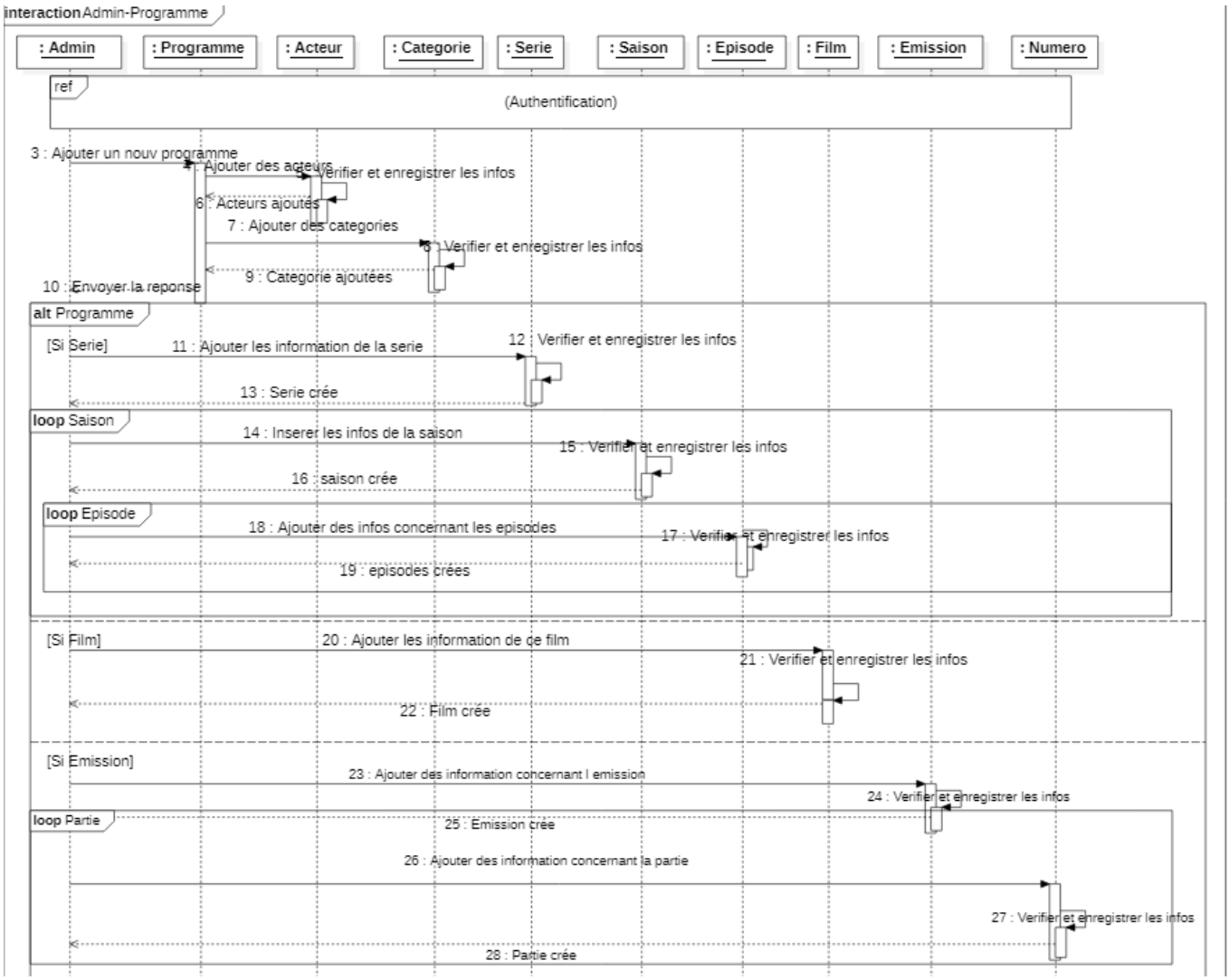


FIGURE 3.5 – Diagramme de séquence "Admin-Programme"

## 3.5 Conclusion

Nous avons exposé les divers concepts utilisés pour concevoir notre système. Nous avons donc entamé en définissant les caractéristiques de notre système. Par la suite, nous les avons structurés à l'aide de diagrammes statiques et dynamiques. Une étape d'implémentation de cette conception consistera à mettre en place notre système. Cette section a été expliquée en détail dans le prochain chapitre. En outre, nous évaluons notre système dans le quatrième chapitre.

# Chapitre 4

# Implémentation et tests

## 4.1 Introduction

Ce chapitre met en lumière les étapes clés pour le développement de notre application, depuis le choix du matériel et des logiciels jusqu'à la réalisation du système fini. Les résultats sont présentés sous forme d'interfaces graphiques. Tout d'abord, nous présenterons l'environnement et les outils de travail utilisés. Ensuite, nous présenterons la métrique d'évaluation, ainsi que les résultats et les tests. Nous décrirons également en détail les étapes d'adaptation affectées à la plateforme de partage de vidéos, afin d'expliquer l'intégration du modèle à cette plateforme. Enfin, nous clôturerons avec la présentation des résultats sous forme d'interfaces graphiques.

## 4.2 Environnement et outils de travail

Sur le plan technique, nous catégorisons les besoins en deux types : les besoins logiciels et les besoins matériels et logiciels.

### 4.2.1 Partie matérielle

Le matériel utilisé consiste en 2 ordinateurs personnels.

• **Le premier poste de travail** dispose d'un système d'exploitation Windows 11 Professionnel 64 bits. Il est équipé d'un processeur Intel(R) Core(TM) i5-8265U CPU @ 1.60 GHz / 3.90 GHz, et d'une mémoire RAM de 12 Go.

• **Le deuxième poste de travail** dispose d'un système d'exploitation Windows 10 Professionnel 64 bits, équipé d'un processeur Intel(R) Core(TM) i5-5200U CPU @ 2.20 GHz / 2.70 GHz, et d'une mémoire RAM de 8 Go.
• **Un démodulateur** satellite pour notre cas celui de géant référence GN-RS8 MiniHD Plus
• **Carte d'acquisition satellite** ou bien Carte de Capture Vidéo VHS Box VHS VCR TV to Digital Converter

### 4.2.2 Partie langages de programmation et logiciels

Elle est composée d'un ensemble d'outils logiciels et divers langages de programmation qui permettent la création et la mise en œuvre de notre système. Notre système a été développé en utilisant les outils suivants :

**Langages de programmation et Frameworks**

• **Python** est un langage de programmation largement utilisé dans les applications Web, le développement de logiciels, la science des données et le machine learning (ML). [30]

• **Django** est un framework sur le langage de programmation Python. [31]

• **SQL** est l'un des plus anciens langages de programmation informatiques pour bases de données relationnelles. Il s'agit aussi du plus populaire. Des données peuvent être ajoutées ou supprimées. En outre, SQL permet de créer ou de modifier la structure d'un système de base de données, de l'optimiser et d'en contrôler l'accès. [32]

**Outils et bibliothèques**

• **Numpy** est en fait l'abréviation de « Numerical Python ».NumPy est une bibliothèque Python pour le calcul scientifique, offrant des tableaux multidimensionnels et des fonctions mathématiques performantes pour des opérations rapides et efficaces sur ces tableaux. [33]

• **Gensim** est une bibliothèque Python pour le NLP, spécialisée dans la modélisation de sujets, le calcul de similitudes et la transformation de grands corpus en vecteurs. Elle utilise des algorithmes comme Word2Vec, Doc2Vec et LDA pour traiter efficacement de grands volumes de texte. [34]

• **Sckit-learn** est une bibliothèque open-source pour l'apprentissage automatique, offrant des outils pour la classification, la régression, le regroupement, et la réduction de la dimensionnalité, ainsi que pour le prétraitement des données. [35]

• **Matpolib** est une bibliothèque Python pour créer des visualisations graphiques variées, comme des courbes, histogrammes, nuages de points et barres. Elle est utilisée pour explorer et comprendre les données grâce à des représentations visuelles intuitives et personnalisables. [36]

• **SciPy** est une bibliothèque open-source de Python pour le calcul scientifique et technique. Basée sur NumPy, elle propose des algorithmes avancés pour l'optimisation, l'intégration, l'interpolation, l'algèbre linéaire et les statistiques, utilisée largement en sciences, ingénierie et mathématiques. [37]

• **Psycopg2** est une bibliothèque open-source pour Python permettant de se connecter et d'interagir avec des bases de données PostgreSQL. Elle facilite l'exécution de requêtes SQL, la gestion des transactions et l'accès aux fonctionnalités avancées de PostgreSQL, reconnue pour sa robustesse et ses performances. [38]

• **NLTK** est une bibliothèque Python open-source pour le traitement automatique du langage naturel (NLP). Elle offre des outils pour manipuler et analyser des textes, incluant le tokenization, le stemming, le tagging, l'analyse syntaxique et la classification de textes. [39]

**Formats de données**

• **Json** fait référence à un format de données et à un format de fichier. JSON est principalement utilisé pour échanger les données d'une application web entre un navigateur et un serveur. Quand un internaute remplit un formulaire en ligne, par exemple, les données renseignées peuvent être stockées sur un serveur au format JSON. C'est un format relativement lisible, qui représente les données structurées sous forme de paires clé/valeur. [40]

**Logiciels et éditeurs de textes**

• **Kaggle Notebooks** est une plateforme web qui accueille la plus grande communauté de Data Science au monde, et qui fournit des outils et des ressources puissants pour nous aider à atteindre tous les progrès de science des données. [41]

• **PostgreSQL** est un SGBDR open-source, réputé pour sa robustesse et sa conformité aux normes SQL. Il offre une grande extensibilité, prenant en charge des fonctionnalités avancées comme les transactions imbriquées, les index multicritères et le stockage de données en format JSON. C'est une solution fiable pour une gamme d'applications, des petites aux systèmes d'entreprise complexes. [38]

• **vMix** est une solution complète de production vidéo en direct et de diffusion en direct. Créez, mixez, basculez, enregistrez et diffusez en direct des productions professionnelles sur un PC ou un ordinateur portable Windows. [42]

• **Castr** est une plateforme de streaming en direct qui permet aux utilisateurs de diffuser du contenu audio et vidéo en temps réel sur Internet. [43]

• **Visual Studio Code** est un éditeur de code développé par Microsoft en 2015. Il est l'un de ces premiers produits open source et gratuit, et surtout disponible sur les systèmes d'exploitation Windows, Linux et Mac. Vs code est développé avec le framework Electron. [44]

• **Git** est un projet open-source qui a été lancé en 2005. Il s'agit d'un système de contrôle de version distribué. Cela signifie que tout développeur de l'équipe ayant un accès autorisé peut gérer le code source et l'historique des modifications à l'aide des outils de ligne de commande Git. [45]

• **Github** est une plate-forme de gestion et d'organisation de projets basée sur le cloud qui intègre les fonctions de contrôle de version de Git. En d'autres termes, tous les utilisateurs de GitHub peuvent suivre et gérer les modifications apportées au code source en temps réel tout en ayant accès à toutes les autres fonctions de Git disponibles au même endroit. [45]

La figure 4.1 montre l'architecture globale de notre système :

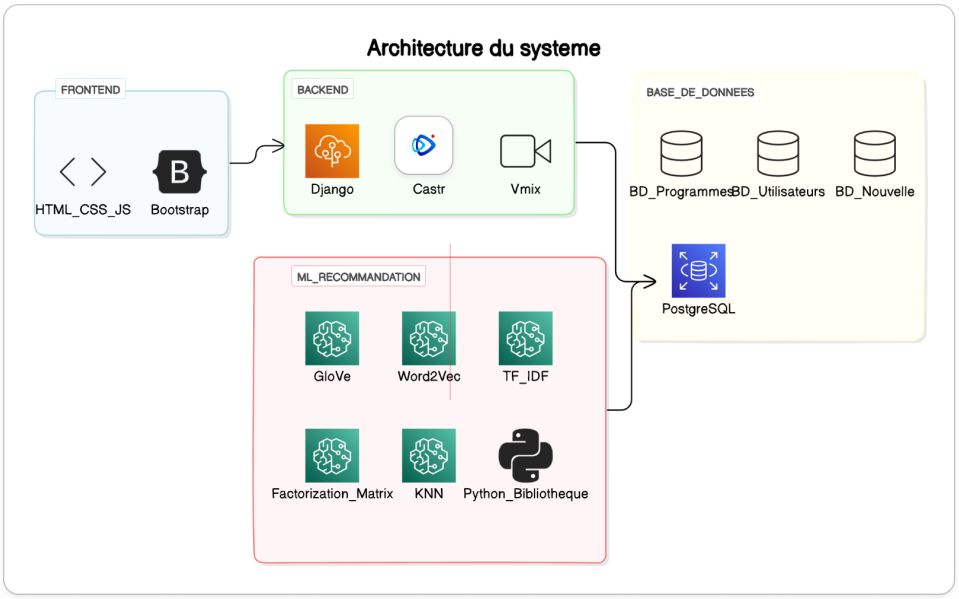


FIGURE 4.1 – Architecture globale du système

Nous avons opté pour HTML, CSS et JavaScript pour le frontend, et pour le backend un framework Python qui est Django, il s'est avéré très efficace pour intégrer les systèmes de recommandation grâce à ses bibliothèques spécialisées. Pour la fonctionnalité de la diffusion en direct, nous avons utilisé Castr et VMix, tandis que PostgreSQL a été sélectionné pour la gestion de la base de données.
La figure 4.2 montre le fonctionnement global de notre système :

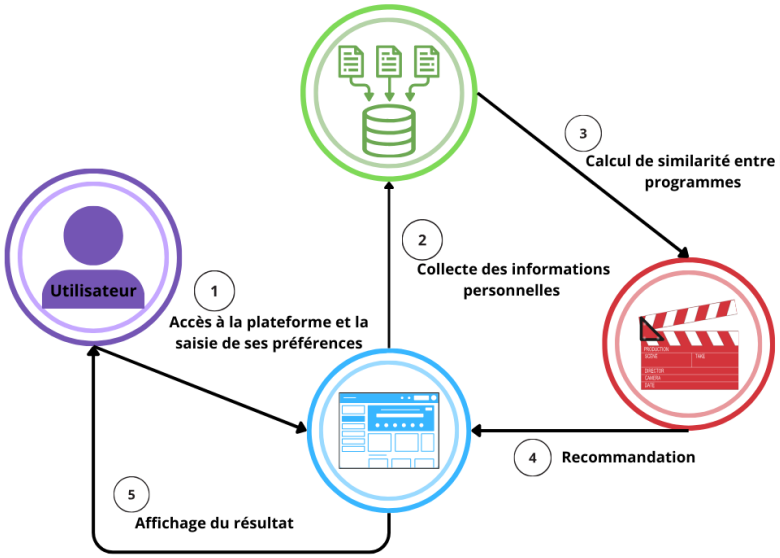


FIGURE 4.2 – Fonctionnement global du système

En ce qui concerne la recommandation, elle a été intégrée de deux manières :

1. Dans le scénario 1, pour un nouvel utilisateur, il doit choisir trois programmes afin que les recommandations de ces trois programmes soient affichées sur l'accueil.

2. Dans le scénario 2, lorsque l'utilisateur sélectionne un programme, le système calcule la similarité entre ce programme choisi et les autres, afin de lui recommander d'autres programmes pertinents.

## 4.3 Résultats et Tests

Dans le but de juger notre travail, nous devons évaluer les systèmes que nous avons conçus. Comme mentionné précédemment, chaque système a son propre traitement et sa propre évaluation. Nous commençons donc par l'évaluation du système basé sur le contenu (BC) :

### 4.3.1 Évaluation des systèmes basé sur le contenu

Dans cette section nous allons expliquer le processus d'évaluation des trois algorithmes proposés dans l'approche BC. Nous débutons par le TF-IDF :

**TF-IDF :** Le test de cet algorithme a été effectué après avoir divisé notre dataset en 70 % pour l'entraînement et 30 % pour le test. Notre cible était les classes des programmes ainsi que leurs thèmes. les titres suivants présentent le résultat de la recommandation en utilisant TF-IDF pour le programme "le liban aujourdhui, liban beyrouth guerre civile documentaire beyrouth, arabe, politique" :

— ramadhan dans le monde islamique, le liban.

— waqafat le liban guerre du liban.

— qissat ardh révolution libanaise.

— liban agression contre le sud de liban.

— les grands dossiers, ayam laha tarikh, le liban guerreliban.

**Word2Vec (W2V) :** Nous avons testé ce modèle en utilisant des clusters pour regrouper les programmes similaires dans le même cluster. Cela a été exécuté après la division du dataset en ensembles de test et d'entraînement. Ce modèle a été entraîné en utilisant les données de l'ENTV, qui contenaient plus de 80 000 programmes avant le nettoyage et plus de 10 000 programmes après le nettoyage. les titres suivants présentent le résultat de la recommandation en utilisant W2V pour le programme "souq el moughafalin" :

— hal wa ahwal.

— mechtaq tah fi hrira.

— zidouhoum fi echarte.

— ahlam wa awham.

— ila wa aila.

**GloVe / Word2Vec 2 :** L'évaluation de ces deux modèles a été effectuée de la même manière que pour le modèle précédent. Cependant, ces deux algorithmes ont été entraînés avec les données de Twitter et de Google respectivement. Le tableau suivant présente le résultat de la recommandation en utilisant GloVe/ W2V 2 pour le titre "bonjour dalgerie" :

| Modèle | Les programmes recommandés |
|---|---|
| W2V 2 | arts et traditions.<br>discours du president houari boumediene a saida.<br>min waqiina.<br>journal televise.<br>hiwar maa el moudjtamaa. |
| GloVe | aid télévision 28 octobre<br>lumiere culturelle<br>direct du stade 5 juillet<br>entretien avec salah boucha<br>ambassadeur et conseiller au mae<br>tissage traditionnel |

TABLE 4.1 – Le résultat de la recommandation en utilisant GloVe et W2V 2

Le tableau suivant présente l'évaluation de Word2Vec, TF-IDF, GloVe et Word2Vec 2 :

| Modèle | W2V | TF-IDF | W2V 2(Pré-entrainée) | GloVe (Pré-entrainée) |
|---|---|---|---|---|
| Accuracy | 0.97 | 0.88 | 0.91 | 0.91 |
| Précision | 0.97 | 0.72 | 0.95 | 0.93 |
| Rappel | 0.96 | 0.69 | 0.90 | 0.89 |
| F1-Score | 0.97 | 0.69 | 0.92 | 0.91 |

TABLE 4.2 – Tableau de comparaison des évaluations des modèles de BC

Comme le tableau 4.8 indique, le modèle de la première technique est plus pertinent que les autres modèles, et cela peut être justifié par :

- Word2Vec capture bien les relations contextuelles et sémantiques à partir des données d'entraînement.
- Word2Vec génère des vecteurs denses et continus qui peuvent mieux représenter les nuances des relations entre les mots. Contrairement à TF-IDF, qui génère des vecteurs sparses.
- Word2Vec, lorsqu'il est entraîné sur des données réelles, il est capable de fournir des recommandations plus pertinentes en capturant le contexte de ces données. GloVe et W2V 2, bien que performants,

peuvent être limités par la qualité et la spécificité des données d'entraînement.

### 4.3.2 Évaluation des systèmes basé sur le filtrage collaboratif

En ce qui concerne l'évaluation du filtrage collaboratif, les résultats suivants sont pour tous les utilisateurs :

| | Précision | Rappel | F1-Score |
|---|---|---|---|
| MF | 0.84 | 0.83 | 0.83 |
| KNN | 0.81 | 0.75 | 0.77 |

TABLE 4.3 – Tableau de comparaison des évaluations des modèles de FC

En premier lieu, les données de la matrice de factorisation ont été divisées comme suit : 70 % pour l'entraînement et 30 % pour le test. Après plusieurs itérations, il s'est avéré que 10 était le meilleur paramètre pour le composant latent. Pour le calcul de la précision, du rappel et du score F1, nous nous sommes basés sur ce qui suit :

- **L'intersection** entre les 8 programmes recommandés et ceux qui ont été regardés par l'utilisateur.
- **Les vrais positifs (TP)** sont les éléments recommandés qui sont effectivement pertinents (par exemple, les programmes recommandés que l'utilisateur a regardé).
- **Les faux positifs (FP)** sont les éléments recommandés qui ne sont pas pertinents (par exemple, les programmes recommandés que l'utilisateur n'a pas regardé).
- **Les faux négatifs (FN)** sont les éléments pertinents qui n'ont pas été recommandés (par exemple, les programmes que l'utilisateur a regardé mais qui n'ont pas été recommandés).

Dans notre cas :

- la Précision se concentre sur la qualité des recommandations. Elle répond à la question "Parmi les éléments recommandés, quelle proportion est correcte ?"
- Le Rappel se concentre sur la couverture des recommandations. Il répond à la question "Quelle proportion des éléments pertinents a été recommandée ?"

Passons à la recommandation des programmes TV pour chaque utilisateur en utilisant un algorithme des k(5) plus proches voisins (KNN) avec une distance de cosinus, et calculons sa **précision**, prenons un exemple avec l'algorithme du KNN :

- Utilisateur A :
  - A regardé : Film1, Film2, Film3, Film4
  - Recommandé : Film1, Film3, Film5

Ses plus proches voisins :

- Utilisateur B : Film1, Film3, Film5
- Utilisateur C : Film1, Film2
- Utilisateur D : Film3, Film4, Film5

— $Film1 = \frac{2}{3}$, $Film2 = \frac{1}{3}$, $Film3 = \frac{2}{3}$, $Film4 = \frac{1}{3}$, $Film5 = \frac{2}{3}$.

Donc les films suggérés seront Film1, Film3, Film5.

Son évaluation :

- TP (True Positives) : 2 (Film1, Film3)
- FP (False Positives) : 1 (Film5)
- FN (False Negatives) : 2 (Film2, Film4)
- Précision : $\frac{2}{2+1} \approx 0.67$
- Rappel : $\frac{2}{2+2} = 0.5$
- F1-score : $2 \times \frac{0.67 \times 0.5}{0.67+0.5} \approx 0.57$

En conclusion, nous remarquons que les résultats de la matrice de factorisation présentés ci-dessus sont meilleurs, bien que la différence par rapport à ceux du KNN ne soit pas énorme.

**Comparaison de la recommandation entre le KNN et la matrice de factorisation**

On choisit un utilisateur au hasard parmi ceux qui ont rempli le formulaire et on compare les programmes recommandés.

Pour l'utilisateur **31** : il a regaré les programmes suivants : achour 10, dama, MUSTAPHA BEN BOULAID, FOOT-BALL ALG A/ NIGERIA

**Knn :** les voisins de l'utilisateur **31** sont [22, 2, 75, 0, 53]

**Recommendation :** bila hodoud, djemai family, foot ball mca - asm oran, a maison hante, la palestine vaincra

**MF : Recommandations** pour l'utilisateur **3**1 : djemai family, timoucha, nass mlah city, bila hodoud, el bedhra,

### 4.3.3 Evaluation des systèmes basé sur le filtrage hybride

Nous finissons par l'évaluation du système basé sur l'approche hybride (FH) :

L'évaluation de ce modèle a été réalisée en utilisant plusieurs méthodes :

— **Méthode 01 :** le calcul de la moyenne des résultats d'évaluation des méthodes de filtrage collaboratif (FC) et basé sur le contenu (BC), car le (FH) combine et fusionne ces deux approches.

— **Méthode 02 :** La création d'un petit jeu de test pour évaluer le résultat de chaque recommandation, sachant que cette méthode n'est pas optimale, vu que notre dataset contient plus de 10 000 programmes aprés le nettoyage.

— **Méthode 03 :** L'utilisation d'une autre technique de vectorisation pour calculer la similarité entre le programme en entrée et les programmes en sortie.

Dans notre cas, nous avons utilisé Word2Vec et la factorisation matricielle (MF), chacun de ces algorithmes offrant les meilleurs résultats dans son approche respective.

Nous avons réalisé un modèle de recommandation hybride en utilisant trois techniques : mixte, pondérée et par entrelacement. La technique mixte et la technique pondérée prennent en entrée un programme et un utilisateur, tandis que celle par entrelacement ne prend que l'utilisateur en paramètre.

La technique mixte calcule la similarité du programme en utilisant Word2Vec et, en parallèle, la similarité des utilisateurs avec U1 en utilisant la factorisation de matrice (MF). Ensuite, ce modèle recommande les cinq programmes les plus similaires par rapport au programme fourni en paramètre, ainsi que les cinq programmes les plus similaires par rapport à U1.

La technique pondérée est similaire à la technique mixte, sauf qu'elle attribue des poids aux recommandations de Word2Vec et de MF, et recommande les programmes les plus proches en fonction de ces poids.

Enfin, la technique par entrelacement prend en entrée uniquement l'utilisateur U1, prédit les utilisateurs similaires à U1 et recommande les cinq programmes que cet utilisateur pourrait aimer en utilisant la MF. Ensuite, le modèle calcule les programmes les plus similaires selon Word2Vec pour chacun de ces cinq programmes, et en sortie, nous obtenons les dix programmes recommandés selon la méthode hybride.

Le tableau suivant présente les résultats de ces trois techniques pour l'utilisateur U1.

| Technique | Programme d'entrée | Les programmes recommandés |
|---|---|---|
| Mixte | thifawat n laid | - le mouton de laid<br>- zaza et le mouton de laid<br>- priere de laid<br>- table ronde sur laid<br>- nass mlah city<br>- djemai family<br>- timoucha<br>- el bedhra<br>- fatma nsoumer |
| Pondération | thifawat n laid | - zaza et le mouton de laid<br>- thifawat n laid<br>- nass mlah city<br>- le mouton de laid sketch<br>- djemai family |
| Par entrelacement | / | - nass mlah city<br>- djemai family<br>- timoucha<br>- el bedhra<br>- fatma nsoumer<br>- nass mlah city3<br>- nass mlah city2<br>- mandjam el ouenza<br>- el awda yousfi tawfik |

TABLE 4.4 – le résultat de la recommandation Hybride

Les tableaux suivants présentent l'évaluation de l'approche hybride :

| Modèle | Précision | Rappel | F1-Score |
|---|---|---|---|
| FH | 0.906 | 0.895 | 0.900 |

TABLE 4.5 – Évaluations de (FH) en utilisant la moyenne

| Modèle | Précision | Rappel | F1-Score |
|---|---|---|---|
| FH | 0.87 | 0.87 | 0.88 |

TABLE 4.6 – Évaluations de (FH) en utilisant le dataset de test

| Modèle | Similarité entre les programmes en entré et les programmes en sortie |
|---|---|
| FH | 0.96 |

TABLE 4.7 – Évaluations de (FH) en utilisant une autre technique de vectorisation

### 4.3.4 Evaluation subjective des systèmes

Dans cette section, nous avons sollicité les employés de l'ENTV pour évaluer les recommandations basées sur la similarité des programmes et des utilisateurs. En tant que connaisseurs des programmes existants, ils sont les mieux placés pour juger de la pertinence des résultats.

Nous avons combiné les deux types de questionnaires afin d'évaluer la l'expérience utilisateur.

| Questions | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | score moyen | Score calculé |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A-t-il été simple pour vous d'utiliser la navigation de l'application EPTV et d'autres fonctionnalités ? | 4 | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4.4 | 3.4 |
| Je pense que j'aurai besoin de l'aide d'un technicien pour être capable d'utiliser cette application | 1 | 1 | 1 | 2 | 1 | 1 | 3 | 4 | 4 | 4 | 2.2 | 2.8 |
| Je trouve que la recommandation BC1 est correcte | 4 | 5 | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4.4 | 3.4 |
| Je trouve que la recommandation BC2 n'est pas correcte | 2 | 1 | 1 | 2 | 2 | 2 | 2 | 1 | 2 | 1 | 1.6 | 3.4 |
| Je trouve que la recommandation BC3 est correcte | 4 | 5 | 5 | 4 | 4 | 5 | 5 | 4 | 4 | 5 | 4.5 | 3.5 |
| Je trouve que la recommandation BC4 n'est pas correcte | 2 | 1 | 1 | 1 | 2 | 2 | 1 | 2 | 2 | 1 | 1.5 | 3.5 |
| Je trouve que la recommandation FC1 est correcte | 5 | 5 | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 4.4 | 3.4 |
| Je trouve que la recommandation FC2 n'est pas correcte | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 1 | 1.4 | 3.6 |
| Je trouve que la recommandation FH1 (Technique 1) est correcte | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 4 | 4 | 5 | 4.7 | 3.7 |
| Je trouve que la recommandation FH2 (Technique 2) n'est pas correcte | 2 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 1 | 1.6 | 3.4 |
| Je trouve que la recommandation FH3 (Technique 3) est correcte | 3 | 5 | 5 | 5 | 4 | 5 | 5 | 5 | 4 | 4 | 4.5 | 3.5 |
| score SUS mulltiplie par 2.5 | | | | | | | | | | | | 94 |

TABLE 4.8 – Évaluations par l'humain

Le Score-SUS (System Usability Scale) constitue un outil essentiel pour évaluer la performance des systèmes. Il se présente sous la forme d'un questionnaire composé de 10 questions, chacune offrant cinq réponses graduées de "Pas du tout d'accord" à "Tout à fait d'accord".

Pour les questions impaires (1, 3, 5, 7, 9), il est convenu de soustraire 1 de la note attribuée. De même, pour les questions paires (2, 4, 6, 8, 10), il faut également retrancher la note de 5.

Ensuite, les scores ajustés sont additionnés et multipliés par 2.5 pour obtenir le score SUS final. Ce score varie de 0 à 100. Un score de 70 ou plus indique que les utilisateurs apprécient l'application et la recommanderaient à leurs amis.

Dans notre cas, nous avons obtenu une valeur SUS de 94% (>70 %), nous pouvons conclure que les résultats de l'évaluation subjective indiquent que notre application est plus facile à utiliser ainsi que son resultat de recommmandation est pertinent.

## 4.4 Intégration et le déploiement

L'intégration et le déploiement de recommandations de programmes basées sur Word2Vec dans une application Django peuvent être réalisés efficacement en suivant quelques étapes clés. Cette approche permet aux utilisateurs de recevoir des recommandations de programmes similaires en fonction du titre d'un programme sélectionné.

Dans ce qui suit, nous souhaitons présenter l'enchaînement des étapes suivies pour intégrer un des modèles réalisés dans la plateforme :

**1. Prétraiter le programme sélectionné par l'utilisateur :**

• Implémenter des fonctions pour supprimer la ponctuation, mettre en minuscules, tokeniser, supprimer les stopwords et lemmatiser le titre du programme.

**2. Charger le modèle Word2Vec :**

• Charger le modèle Word2Vec pré-entraîné à partir d'un chemin spécifié.

**3. Générer les recommandations :**

• Recevoir les requêtes POST avec le titre du programme, le prétraiter et utiliser Word2Vec pour trouver les titres les plus similaires.

• Retourner les recommandations sous forme de JSON.

**4. Créer les vues Django :**

•Définir la fonction `recommend_programs` pour gérer les requêtes POST et retourner des recommandations.

• Créer `recommendation_form` pour afficher un formulaire de soumission de texte.

**5. Développer les templates :**

• Incorporer les titres recommandés dans la section recommandation de la plateforme développée préalablement.

**6. Gérer les erreurs :**

• Implémenter des gestionnaires pour les erreurs courantes telles que les erreurs de JSON et les fichiers non trouvés.

**7. Journaliser les activités :**

• Ajouter des logs pour suivre le chargement du modèle et les erreurs.

En plus de cela, Django, en tant que framework web Python, offre une intégration aisée des modèles d'apprentissage automatique sans nécessiter la mise en place d'une API distincte. Sa structure basée sur le modèle MVC (Modèle-Vue-Contrôleur) permet une séparation claire des préoccupations, simplifiant ainsi l'ajout et la gestion des composants d'apprentissage automatique. Par ailleurs, Python étant le langage principal de Django, il offre une cohérence linguistique, permettant aux développeurs d'utiliser les mêmes compétences et outils pour le développement web et l'intégration d'algorithmes d'apprentissage automatique.

## 4.5 Réalisation de la partie "en direct"

-Brancher le démodulateur satellite et le connecter à la carte de capture vidéo.

-Configurer les entrées du logiciel vMix pour recevoir le signal vidéo.

-Se connecter à Castr et choisir un emplacement pour héberger la vidéo.

-Copier la clé et le lien fournis par Castr dans l'interface de diffusion en direct de vMix.

-Lancer la diffusion en direct.

-Copier la balise HTML fournie par Castr dans la plateforme où l'information doit être partagée, permettant ainsi au public de suivre la diffusion.

## 4.6 Présentation de l'application

NAVision est une plateforme de streaming innovante conçue pour offrir une expérience de visionnage diversifiée et enrichissante. Voici un aperçu des sections clés de notre plateforme :

### Sections de la Plateforme

#### Authentification

— **Fonctionnalité** : Permet aux utilisateurs de créer un compte et de gérer des profils associés.

— **Objectif** : Assurer une expérience personnalisée et sécurisée pour chaque utilisateur.

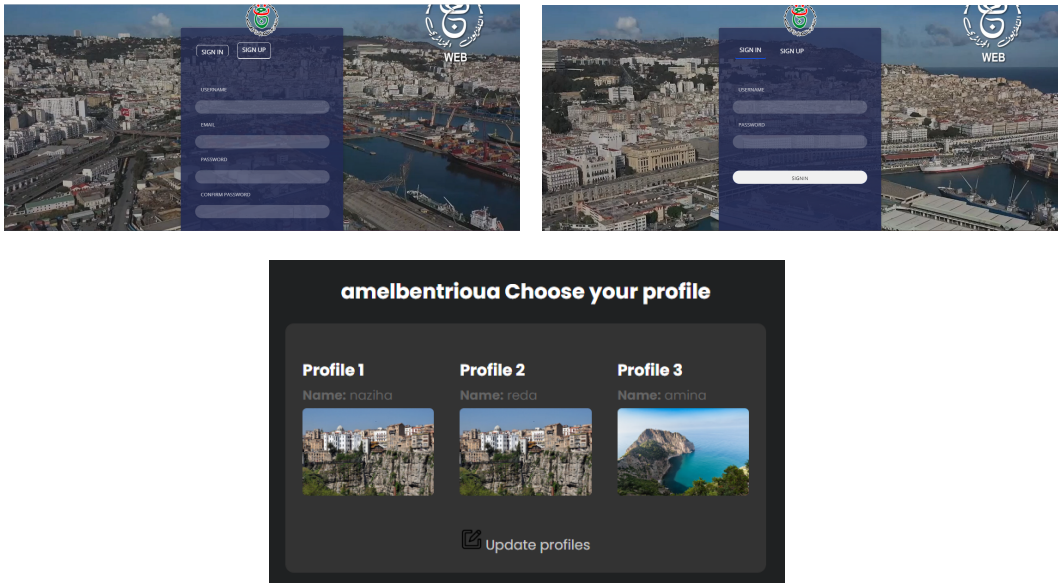


FIGURE 4.3 – Authentification

### Home

— **Contenu** : Affiche les derniers ajouts ainsi que les programmes les plus regardés.

— **Objectif** : Offrir une vue d'ensemble des contenus tendance et des nouveautés.

— **Fonctionnalité Supplémentaire** : Permet aux utilisateurs d'ajouter des programmes à leur liste de "Regarder plus tard".

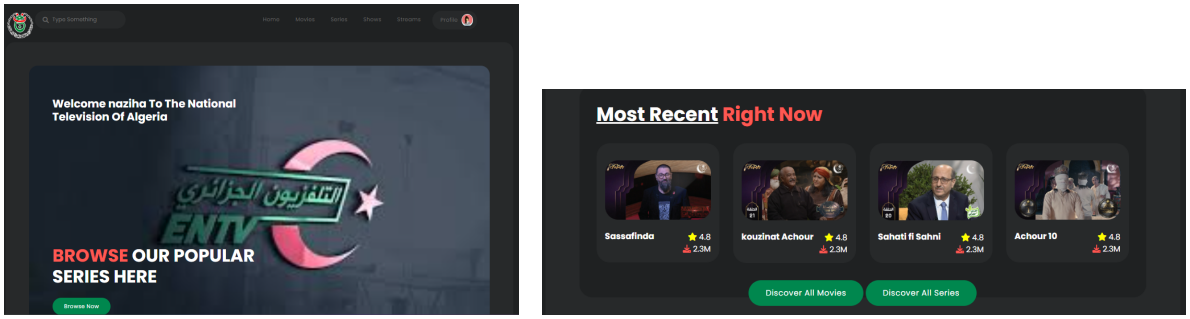


FIGURE 4.4 – Home

### Movies

— **Contenu** : Une rubrique dédiée aux films de l'EPTV (Établissement Public de Télévision).

— **Objectif** : Proposer une vaste sélection de films pour tous les goûts.

### Series

— **Contenu** : Une rubrique dédiée aux séries de l'EPTV (Établissement Public de Télévision ), couvrant tous les genres, de la comédie au drame.

— **Objectif** : Offrir un large éventail de séries, avec plusieurs saisons et épisodes par série.

### Shows

— **Contenu** : Rubrique dédiée aux émissions diverses telles que les émissions de cuisine, politiques, et même les émissions sportives.

— **Objectif** : Fournir un choix varié d'émissions pour satisfaire tous les intérêts.

### Stream

— **Contenu** : Permet de regarder les 9 chaînes de la TV traditionnelle directement sur la plateforme.

— **Objectif** : Offrir une expérience de visionnage en temps réel pour les chaînes de télévision traditionnelles.


FIGURE 4.5 – Stream

### Profile

— **Fonctionnalité** : Espace dédié à la gestion du profil utilisateur.

— **Objectif** : Permettre aux utilisateurs de personnaliser leur expérience de visionnage.

— **Fonctionnalité Supplémentaire** : Les utilisateurs peuvent consulter et gérer leur liste de "Regarder plus tard".

La section 'Série' comporte un large choix de séries, chacune possédant plusieurs saisons et chaque saison ayant plusieurs épisodes." Pour plus de clarté les figures si dessous le démontre

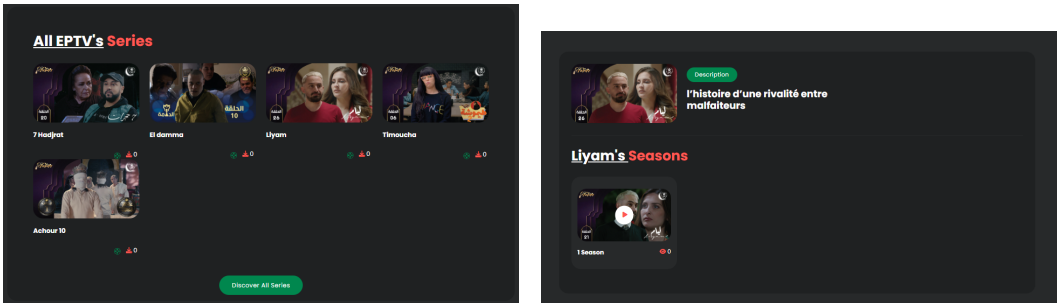


FIGURE 4.6 – Exemple séries et saison Liyam

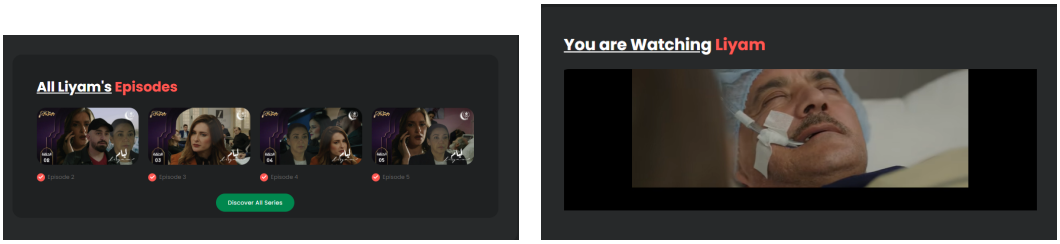


FIGURE 4.7 – Exemple épisodes Liyam

Cette logique est aussi appliquée dans les émissions qui ont des numéros.

**Recommandations**

— **Solution au démarrage à froid**
Lorsque vous démarrez NAVision pour la première fois sur un nouveau profil, la plateforme doit être capable de suggérer du contenu initial en se basant sur le catalogue de choix. Cette sélection n'apparaîtra qu'une seule fois lors de la première connexion, et ses résultats apparaîtront dans la page d'accueil ("home").

— **Recommandation basée sur le contenu**
Chaque programme sur la plateforme dispose d'une icône qui génère des recommandations spécifiques à ce programme. Par exemple, pour "Souq El Moughafalin", l'icône associée propose des suggestions en lien direct avec ce contenu.

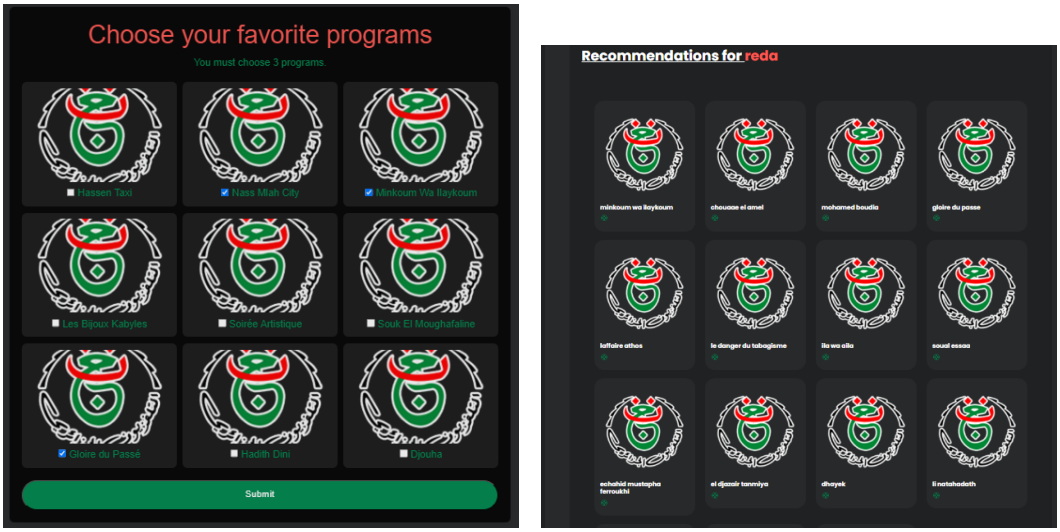


FIGURE 4.8 – solution au démarrage à froid

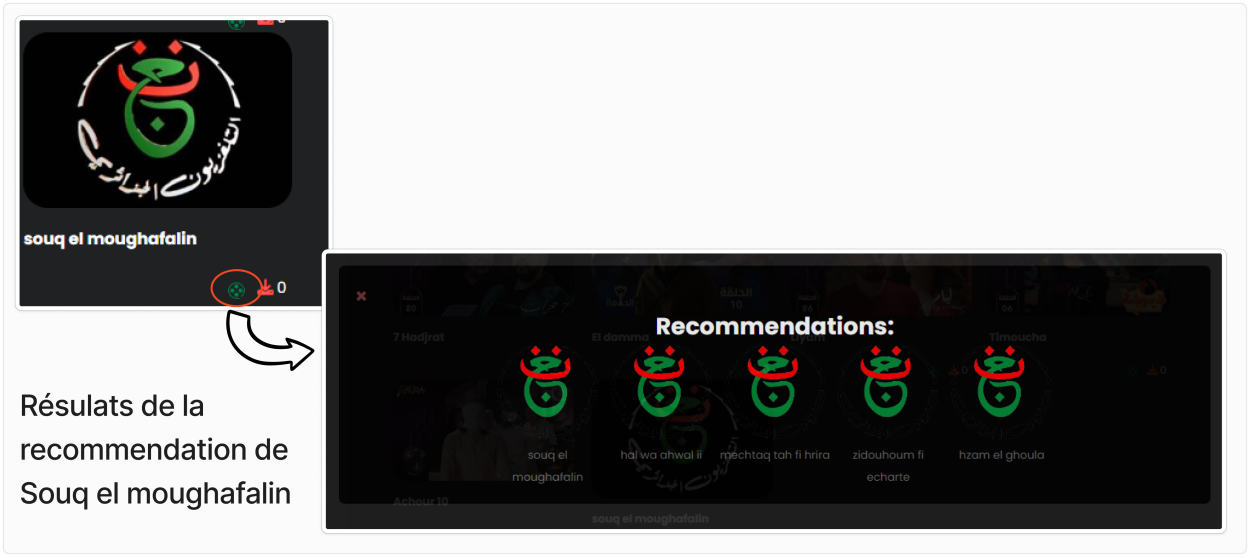


FIGURE 4.9 – Recommendations "Souq el moughafalin"

## 4.7 Discussion

Les résultats des recommandations basées sur le contenu montrent que celles utilisant Word2Vec sont plus efficaces que celles utilisant GloVe ou Word2Vec 2, car le modèle Word2Vec a été entraîné avec des données réelles d'ENTV, tandis que GloVe et Word2Vec 2 ont été entraînés respectivement avec des données de Twitter et de Google. Cependant, Word2Vec 2 est meilleur que GloVe en raison du volume de données plus important utilisé pour entraîner Word2Vec 2 par rapport à GloVe. En outre,

Word2Vec est considéré comme supérieur au TF-IDF pour la recommandation dans de nombreux contextes grâce à sa capacité à capturer les relations sémantiques et contextuelles entre les mots.

D'autre part, les résultats des recommandations basées sur le filtrage collaboratif indiquent que la matrice de factorisation est plus pertinente que le modèle du KNN grâce à leur capacité à capturer des relations complexes dans les données et à généraliser les préférences des utilisateurs. MF permet de capturer les relations latentes entre les utilisateurs et les items, réduisant ainsi le bruit et révélant des patterns sous-jacents dans les données, comme il est capable de modéliser des interactions complexes entre les utilisateurs et les items en apprenant des facteurs latents. Les modèles factorisés sont souvent plus scalables que KPPV. Une fois le modèle entraîné, les prédictions sont rapides car elles impliquent simplement des produits de vecteurs. Et finalement permet des recommandations plus fines par rapport au KPPV. Par contre, KNN peut avoir des performances réduites avec de grandes quantités de données en raison de la nécessité de calculer les distances pour chaque observation, ce qui peut entraîner des temps de calcul plus longs et une scalabilité limitée.

De plus, Les recommandations basées sur l'approche hybride semblent avoir le plus grand potentiel parmi toutes les techniques explorées et implémentées. Grâce à cette technique, nous avons pu fusionner et combiner les résultats des deux premières approches (BC) et (FC). Bien que le volume du dataset pour le filtrage collaboratif (FC) ne soit pas suffisamment important pour recommander de bons choix, en raison du nombre limité d'utilisateurs, nous avons intégré les résultats du (FC) comme une entrée dans le filtrage basé sur le contenu (BC) pour optimiser et améliorer les recommandations. Le dataset utilisé dans le (BC) est important en termes de volume et a été bien nettoyé, ce qui a contribué à la qualité des résultats.

En ce qui concerne les critères de comparaison étudiés dans le chapitre 01, l'objectif principal est de promouvoir notre patrimoine tout en ajoutant de nouvelles fonctionnalités. Le contenu disponible comprend une large gamme de programmes produits par l'EPTV.Quant à l'impact financier et commercial, il sera déterminé une fois que l'application sera mise en ligne.

## 4.8 Conclusion

Après avoir franchi les étapes cruciales de l'implémentation et de l'évaluation de nos modèles d'IA, ainsi que la présentation détaillée des interfaces de notre application, nous avons pleinement concrétisé notre ambition initiale. En harmonisant habilement la puissance de l'intelligence artificielle avec une conception d'interface utilisateur intuitive, nous avons créé un système robuste et intuitif qui répond aux besoins de nos utilisateurs.

# Conclusion Générale

En conclusion, nos objectifs principaux ont été atteints avec succès. Nous avons créé NAVision, une plateforme centralisée qui regroupe et diffuse les programmes de l'EPTV, offrant ainsi une expérience utilisateur simplifiée et enrichie. La diffusion en direct sur cette plateforme a également été mise en place, permettant aux spectateurs de suivre leurs contenus préférés en temps réel. De plus, nous avons implémenté un système de recommandations personnalisé, basé sur les goûts et préférences de chaque spectateur en temps réel, afin d'améliorer leur satisfaction et leur fidélité. En effet, plus les utilisateurs utilisent le système de recommandation (SR) d'une entreprise, plus sa valeur augmente, et plus sa valeur augmente, plus les utilisateurs l'utilisent.

En regardant vers l'avenir, nous avons plusieurs perspectives prometteuses. Nous prévoyons d'ajouter d'autres catégories de contenus pour diversifier notre offre et répondre aux attentes variées de notre audience. Une amélioration continue de notre base de données est essentielle pour tester de nouvelles approches. Par ailleurs, nous envisageons d'introduire des options de paiement et de téléchargement, ainsi que la restriction du partage d'écrans.

Nous souhaitons remplacer le dataset actuel par celui intégré dans la plateforme après avoir collecté de nouveaux jeux de données. Ces données doivent contenir plusieurs paramètres tels que les évaluations et le nombre de vues de chaque programme, ainsi qu'un paramètre ajusté pour faire des découvertes au hasard, différentes des goûts habituels de l'utilisateur, afin de résoudre le défi de la sérendipité.

La prochaine décennie annonce l'émergence des objets connectés grâce à l'avènement des moyens de communications sans fil et à l'amélioration des technologies embarquées. Ces nouveaux objets offrent autant d'opportunités de collecter des données pouvant servir à la génération de recommandations et rendront les communications pratiquement en temps réel.

Utiliser des techniques de traitement du langage naturel pour analyser les commentaires et les avis des utilisateurs, ce qui peut enrichir les recommandations.

Utiliser des techniques de recommandation d'image et des vidéos pour recommander selon le contenu des programmes.

Prendre en compte l'heure de la journée et la localisation pour proposer des recommandations plus pertinentes.

Intégrer des fonctionnalités de réseau social pour permettre aux utilisateurs de voir ce que leurs amis regardent.

Permettre aux utilisateurs de créer et de partager des listes de recommandations.

Assurer que les données des utilisateurs sont protégées.

Il est recommandé d'analyser les systèmes de recommandation, non pas comme une simple stratégie de vente, mais plutôt comme une ressource renouvelable pour améliorer constamment notre connaissance des clients et nos propres connaissances. Plusieurs sociétés possèdent déjà des milliers d'utilisateurs et de données, mais n'ont pas réussi à atteindre le succès escompté. Le manque de connaissance sur la manière de convertir leurs données utilisateur en informations exploitables, pouvant ensuite être utilisées pour améliorer leurs produits ou services, explique pourquoi leur cycle vertueux n'a pas pris l'importance attendue.

Nous vivons dans un monde de plus en plus disruptif où l'accès à la connaissance n'a jamais été aussi facile. En agissant intelligemment, nous pouvons créer le changement qui façonnera le monde de demain et assurera le succès de nos entreprises.

# Bibliographie


[1] A. INCONNU, *Interstices* **2024**.

[2] A. INCONNU, Netflix : Une success story basée sur des algorithmes de recommandation, **2024**, https://mediego.com/fr/blog/netflix-success-story-basee-sur-algorithmes-de-recommandation/.

[3] N. GARNINE, Filière : Informatique, Option : Systèmes Informatiques, Mémoire de fin d'études Master, **2020**.

[4] ARCBEES, Introduction aux systèmes de recommandation, https://medium.com/@Arcbees/introduction-aux-systÃĺmes-de-recommandation-d2f98d3e4160.

[5] DATASCIENTEST, KNN, https://datascientest.com/knn.

[6] D. BOKDE, S. GIRASE, D. MUKHOPADHYAY, *International Journal of Computer Applications* **2014**, *101*, 2348-4853.

[7] I. ESSLIMANI, Interface homme-machine [cs.HC], Université Nancy II, **2010**.

[8] C. OBEID, thèse de doct., Université de Lille, **2021**.

[9] ARCBEES, Introduction aux systèmes de recommandation, https://medium.com/@Arcbees/introduction-aux-syst%C3%A8mes-de-recommandation-d2f98d3e4160.

[10] I. BENOUARET, Thèse présentée pour l'obtention du grade de Docteur de l'UTC, Thèse de doctorat, Université de Technologie de Compiègne, Compiègne, France, **2017**.

[11] J. BASTIN, Master en sciences de gestion (Horaire décalé), HEC-Ecole de gestion de l'Université de Liège, **2021**.

[12] GEEKSFORGEEKS, Word Embeddings in NLP, https://www.geeksforgeeks.org/word-embeddings-in-nlp/.

[13] EDUSCOL, Word Embedding : Les Mots et le Machine Learning, **Accessed : 24 mars 2024**, https://eduscol.education.fr/sti/sites/eduscol.education.fr.sti/files/ressources/pedagogiques/14960/14960-word-embedding-les-mots-et-le-machine-learning-ensps.pdf.

[14] OPENCLASSROOMS, Représentez votre corpus en bag-of-words, A section from an OpenClassrooms course explaining how to represent a text corpus using the bag-of-words model, which is a common technique in text analysis and natural language processing., **Accessed : 2024**, https://openclassrooms.com/fr/courses/4470541-analysez-vos-donnees-textuelles/4855001-representez-votre-corpus-en-bag-of-words.

[15] D. O. LAMIA, Module : Recherche d'Information, Course Material, Université Saad Dahleb Blida, **2019-2020**.

[16] IONOS, Analyse TF-IDF : Définition et explications, An article explaining the concept of TF-IDF (Term Frequency-Inverse Document Frequency), its importance in text analysis, and how it is used to evaluate the importance of words in a document relative to a corpus., **Accessed : 25 mars 2024**, https://www.ionos.fr/digitalguide/web-marketing/analyse-web/analyse-tf-idf/.

[17] GURU99, Word Embedding with Word2Vec : Everything You Need to Know, An in-depth guide on Word2Vec, a technique for natural language processing that enables the representation of words as vectors in a continuous vector space., **Accessed : 2024**, https://www.guru99.com/fr/word-embedding-word2vec.html#what-is-word2vec.

[18] T. D. SCIENCE, NLP 101 : Word2Vec, Skip-Gram, and CBOW, https://towardsdatascience.com/nlp-101-word2vec-skip-gram-and-cbow-93512ee24314.

[19] IBM, Les réseaux de neurones, https://www.ibm.com/fr-fr/topics/neural-networks#:~:text=Les%20r%C3%A9seaux%20de%20neurones%2C%20%C3%A9galement%20connus%20sous%20le,au%20c%C5%93ur%20des%20algorithmes%20de%20l%27apprentissage%20en%20profondeur..

[20] ICHI.PRO, Word2Vec, GloVe, FastText et word embeddings de base étape par étape, An article explaining the basics of Word2Vec, GloVe, FastText, and word embeddings, providing a step-by-step guide on how these models work and their applications in natural language processing., **Accessed : 2024**, https://ichi.pro/fr/word2vec-glove-fasttext-et-word-embeddings-de-base-etape-par-etape-229010274898187.

[21] DATASCIENTEST, Erreur quadratique moyenne, https://datascientest.com/erreur-quadratique-moyenne.

[22] KOBIA, Classification Metrics : F1 Score Explained, https://kobia.fr/classification-metrics-f1-score/.

[23] INSIDE MACHINE LEARNING, Recall, Precision, F1 Score Explained, https://inside-machinelearning.com/recall-precision-f1-score/.

[24] KOBIA, Régression Metrics : quelle métrique choisir ?, https://kobia.fr/regression-metrics-quelle-metrique-choisir/.

[25] P. JEAN-BAPTISTE, IA & Tokenization, https://www.linkedin.com/pulse/ia-tokenization-philippe-jean-baptiste-executive-mba-mpa-msc-mavbyje/.

[26] LINKEDIN, Comprendre la littératie de l'IA générative, https://www.linkedin.com/pulse/comprendre-la-litt%C3%A9ratie-de-lia-g%C3%A9n%C3%A9rative-af6gf/.

[27] GEEKSFORGEEKS, Cosine Similarity, **Accessed : 2024**, https://www.geeksforgeeks.org/cosine-similarity/.

[28] B. SARWAR, G. KARYPIS, J. KONSTAN, J. RIEDL, *Fifth International Conference on Computer and Information Science (ICIS 2000)* **2000**.

[29] LUCIDCHART, Langage UML, https://www.lucidchart.com/pages/fr/langage-uml.

[30] AMAZON WEB SERVICES, What is Python ?, **Accessed : 2024**, https://aws.amazon.com/fr/what-is/python/.

[31] WAYTOLEARNX, C'est quoi Django ? Avantages et inconvénients, Accessed : 2024, **2020**, https://waytolearnx.com/2020/02/cest-quoi-django-avantages-et-inconvenients.html.

[32] LEBIGDATA, SQL : Tout savoir - Guide, An article providing a comprehensive guide to SQL, covering its basics, advanced topics, and practical applications., **Accessed : 15 mai 2024**, https://www.lebigdata.fr/sql-tout-savoir-guide.

[33] DATASCIENTEST, NumPy, **Accessed : 2024**, https://datascientest.com/numpy#:~:text=Le%20terme%20NumPy%20est%20en%20fait%20l%E2%80%99abr%C3%A9viation%20de,Science%2C%20pour%20l%E2%80%99ing%C3%A9nierie%2C%20les%20math%C3%A9matiques%20ou%20la%20science..

[34] DATASCIENTEST, Gensim : Tout savoir, Une bibliothèque Open Source de traitement de langage naturel (NLP) en Python dont le but est de rendre la modélisation de sujet (topic modelling) aussi facile d'accès et efficace que possible., **Accessed : 2024**, https://datascientest.com/gensim-tout-savoir.

[35] ECOAGI, What is sklearn ?, **Accessed : 2024**, https://ecoagi.ai/fr/topics/Python/what-is-sklearn.

[36] INTELLIGENCE ARTIFICIELLE SCHOOL, Matplotlib, **Accessed : 2024**, https://www.intelligence-artificielle-school.com/ecole/technologies/matplotlib/#:~:text=Matplotlib%20est%20une%20biblioth%C3%A8que%20de%20Data%20Visualization%20en,les%20signaux%20%C3%A9lectriques%20du%20cerveau%20de%20patients%20%C3%A9pileptiques..

[37] DATASCIENTEST, SciPy, An article about SciPy, a Python library used for scientific computing and technical computing. It builds on NumPy and provides a large collection of mathematical functions, optimization, interpolation, and signal processing tools, among others., **Accessed : 2024**, https://datascientest.com/scipy.

[38] STACKLIMA, Python : Premiers pas avec Psycopg2 PostgreSQL, An article providing a beginner's guide to using Psycopg2, a PostgreSQL adapter for Python, including installation instructions and basic usage examples., **Accessed : 2024**, https://stacklima.com/python-premiers-pas-avec-psycopg2-postgresql/.

[39] DATASCIENTEST, NLTK, An article about NLTK (Natural Language Toolkit), a Python library for natural language processing (NLP). NLTK provides easy-to-use interfaces to over 50 corpora and lexical resources, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning., **Accessed : 2024**, https://datascientest.com/nltk.

[40] HUBSPOT, Comment ouvrir, lire et écrire un fichier JSON en Python, An article in French explaining how to open, read, and write JSON files in Python., **Accessed : 15 mai 2024**, https://blog.hubspot.fr/website/fichier-json.

[41] DATASCIENTEST, Kaggle : Tout ce qu'il faut savoir sur cette plateforme, An article providing comprehensive information about Kaggle, a web platform that hosts data science competitions and datasets. It serves as a hub for data scientists, machine learning practitioners, and researchers to collaborate, compete, and learn, driving progress in the field of data science., **Accessed : 15 mai 2024**, https://datascientest.com/kaggle-tout-ce-quil-a-savoir-sur-cette-plateforme.

[42] VMIX, vMix, **Accessed : 2024**, https://www.vmix.com/.

[43] CASTR, Live TV Solutions, Castr provides live TV solutions that enable users to broadcast live video content to multiple platforms simultaneously. It supports various features such as streaming to multiple destinations, live video encoding, and more to ensure high-quality live broadcasts., **Accessed : 2024**, https://castr.com/solutions/live-tv/.

[44] WEBNET, Visual Studio Code : Le guide ultime pour les développeurs, An article providing an ultimate guide for developers on Visual Studio Code, covering various features, tips, and

tricks to enhance productivity and efficiency in coding., **Accessed :15 mai 2024**, https : //blog.webnet.fr/visual-studio-code/.

[45] HOSTINGER, GitHub : c'est quoi et comment l'utiliser, An introductory tutorial on GitHub, explaining what it is and how to use it., **Accessed : 15 mai 2024**, https : //www.hostinger.fr/tutoriels/github-cest-quoi-et-comment-lutiliser.