

# Système de recommandation d'anime

2022-2023

Laurent HE

# Sommaire

## 1. Introduction

- a. Définitions
- b. Contexte

## 2. Dataset

## 3. Visualisation

## 4. Solutions

- a. Content-based filtering
- b. Collaborative filtering

## 5. Conclusion

# 1. Introduction

- But/ problématique du projet ?
- Construire un système de recommandation pour la proposition d'animes.

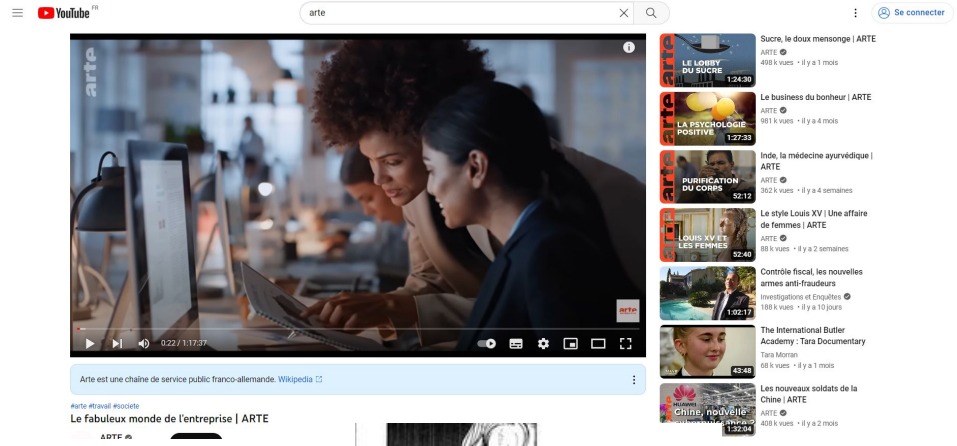
# 1. Introduction

- système de recommandation ?

amazon.com

NETFLIX

- anime ? manga ?



## 2. Dataset



### Anime.csv

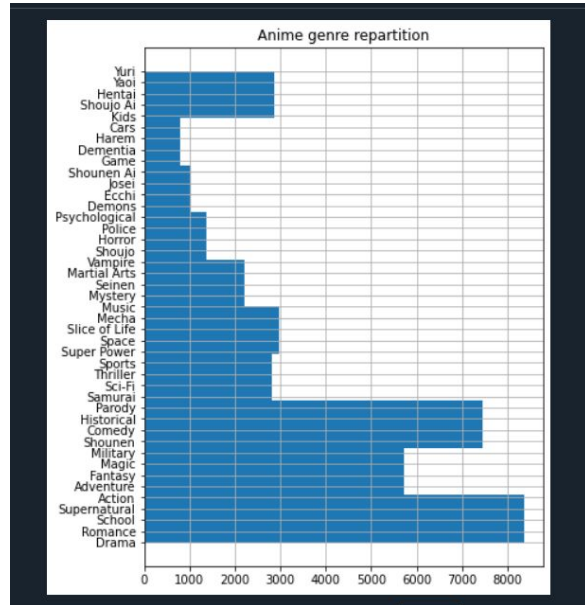
- anime\_id - myanimelist.net's unique id identifying an anime.
- name - full name of anime.
- genre - comma separated list of genres for this anime.
- type - movie, TV, OVA, etc.
- episodes - how many episodes in this show. (1 if movie).
- rating - average rating out of 10 for this anime.
- members - number of community members that are in this anime's "group".

### Rating.csv

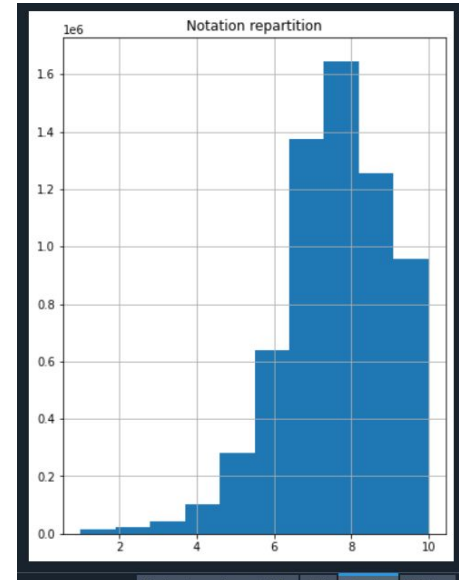
- user\_id - non identifiable randomly generated user id.
- anime\_id - the anime that this user has rated.
- rating - rating out of 10 this user has assigned (-1 if the user watched it but didn't assign a rating).

# 3. Visualisation des données

- 12 294 animes
- $(7\,813\,737 - 1\,476\,496) = 6\,337\,241$  notes



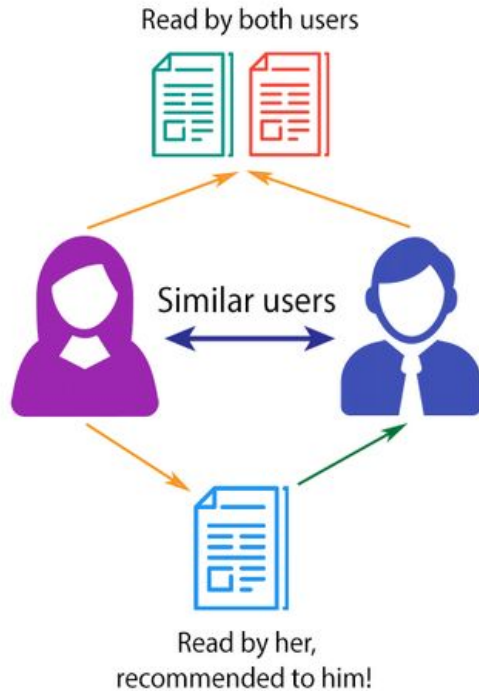
```
[ 'Drama', 'Romance', 'School', 'Supernatural' ]  
[ 'Action', 'Adventure', 'Drama', 'Fantasy', 'Magic', 'Military', 'Sho ...  
[ 'Action', 'Comedy', 'Historical', 'Parody', 'Samurai', 'Sci-Fi', 'Sho ...  
[ 'Sci-Fi', 'Thriller' ]  
[ 'Action', 'Comedy', 'Historical', 'Parody', 'Samurai', 'Sci-Fi', 'Sho ...  
[ 'Comedy', 'Drama', 'School', 'Shounen', 'Sports' ]
```



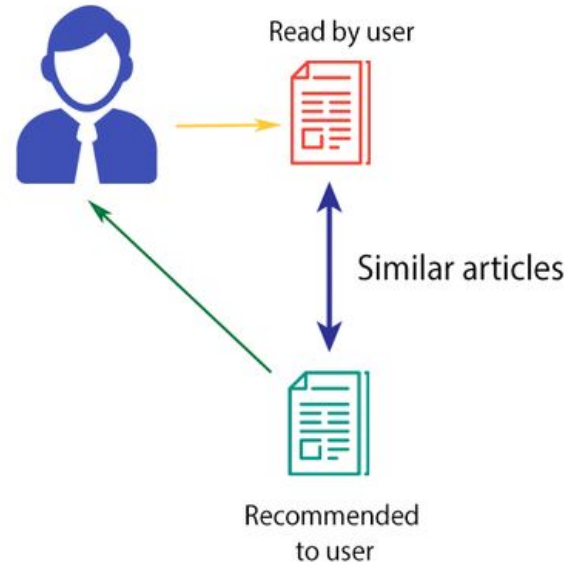
8	1646019
-1	1476496
7	1375287
9	1254096
10	955715
6	637775
5	282806
4	104291
3	41453
2	23150
1	16649

## 4.a. Solutions

### COLLABORATIVE FILTERING



### CONTENT-BASED FILTERING



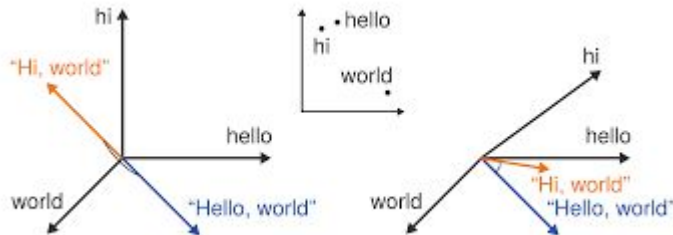
## 4.a. Solutions - content-based filtering

$$w_{x,y} = \text{tf}_{x,y} \times \log\left(\frac{N}{\text{df}_x}\right)$$

**TF-IDF**

Term  $x$  within document  $y$

$\text{tf}_{x,y}$  = frequency of  $x$  in  $y$   
 $\text{df}_x$  = number of documents containing  $x$   
 $N$  = total number of documents



- `from sklearn.feature_extraction.text`  
`import TfidfVectorizer` : importer la bibliothèque
- `tfidf_matrix =`  
`tfv.fit_transform(genres_str)` :  
appliquer l'algorithme sur l'ensemble des  
animes. On obtient alors une matrice  
avec comme taille : "nombre  
d'anime"x"nombre de genre".

- `from sklearn.metrics.pairwise` `import`  
`cosine_similarity` : importer la bibliothèque
- `cosine_sim` `=`  
`cosine_similarity(tfidf_matrix,tfidf_matrix)` :  
appliquer la similarité cosinus sur l'ensemble des  
genres des animes et les comparer un à un. On  
obtient alors un tableau 2D avec comme taille :  
"nombre d'anime"x"nombre d'anime"



## 4.a. Solutions - content-based filtering

```
anime_id      21
name          One Piece
genre         Action, Adventure, Comedy, Drama, Fantasy, Sho...
type          TV
episodes      Unknown
rating        8.58
members       504862
Name: 74, dtype: object

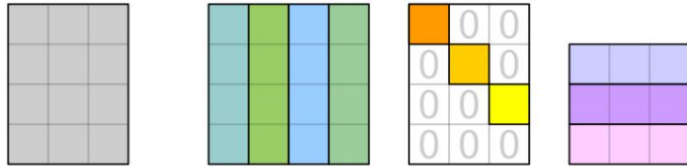
              Nom                                Genre
0  One Piece: Episode of Nami - Koukaishi no Nami... Action, Adventure, Comedy, Drama, Fantasy, Sho...
1  One Piece: Episode of Sabo - 3 Kyoudai no Kizu... Action, Adventure, Comedy, Drama, Fantasy, Sho...
2      One Piece Film: Strong World Episode 0      Action, Adventure, Comedy, Fantasy, Shounen, S...
3  One Piece: Episode of Luffy - Hand Island no B... Action, Adventure, Comedy, Fantasy, Shounen, S...
4      One Piece Movie 4: Dead End no Bouken      Action, Adventure, Comedy, Fantasy, Shounen, S...
anime_id      2167
```

```
anime_id      2167
name          Clannad
genre         Comedy, Drama, Romance, School, Slice of Life,...
type          TV
episodes      23
rating        8.3
members       566690
Name: 223, dtype: object

              Nom                                Genre
0      Kokoro Connect: Michi Random      Comedy, Drama, Romance, School, Slice of Life,...
1              Kokoro Connect      Comedy, Drama, Romance, School, Slice of Life,...
2      Little Busters!: EX      Comedy, Drama, Romance, School, Slice of Life,...
3  Inou-Battle wa Nichijou-kei no Naka de      Comedy, Romance, School, Slice of Life, Supern...
4      Little Busters!      Comedy, Drama, School, Slice of Life, Supernat...
```



## 4.b. Solutions - collaborative filtering - SVD



The diagram shows the SVD decomposition of a matrix  $M$  into three matrices:  $U$ ,  $\Sigma$ , and  $V^*$ .  $M$  is a 4x4 gray grid.  $U$  is a 4x4 grid with columns of different colors (teal, green, blue, green).  $\Sigma$  is a 4x4 grid with a diagonal of colored squares (orange, yellow, yellow, yellow) and zeros elsewhere.  $V^*$  is a 4x4 grid with rows of different colors (light blue, purple, purple, pink).

$$\begin{matrix} \mathbf{M} \\ m \times n \end{matrix} = \begin{matrix} \mathbf{U} \\ m \times m \end{matrix} \begin{matrix} \mathbf{\Sigma} \\ m \times n \end{matrix} \begin{matrix} \mathbf{V}^* \\ n \times n \end{matrix}$$

Exemple :

$$\begin{matrix} (2,5,4) \\ (3,3,0) \end{matrix} = \begin{bmatrix} -0.52 & 0.85 \\ -0.85 & -0.52 \end{bmatrix} * \begin{bmatrix} 7.46 & 0.00 \\ 0.00 & 2.23 \end{bmatrix} * \begin{bmatrix} -0.63 & -0.77 \\ -0.77 & 0.63 \end{bmatrix}$$

On cherche la note de l'utilisateur 2 pour l'anime 3 :

$$M'[2,3] = U[2,:] * S[:,2] * V^T[:,3] = [-0.85, -0.52] * [0, 2.23] * [-0.77] = 1.21$$



- **RMSE : 1.10**
- **MAE : 0.82**

## 4.b. Solutions - collaborative filtering - SVD

```
Recommendations pour l'utilisateur 10350:  
['Kara no Kyoukai 5: Mujun Rasen']  
['Usagi Drop']  
['Natsume Yuujinchou Shi']  
['Suzumiya Haruhi no Shoushitsu']  
['Kara no Kyoukai 2: Satsujin Kousatsu (Zen)']
```

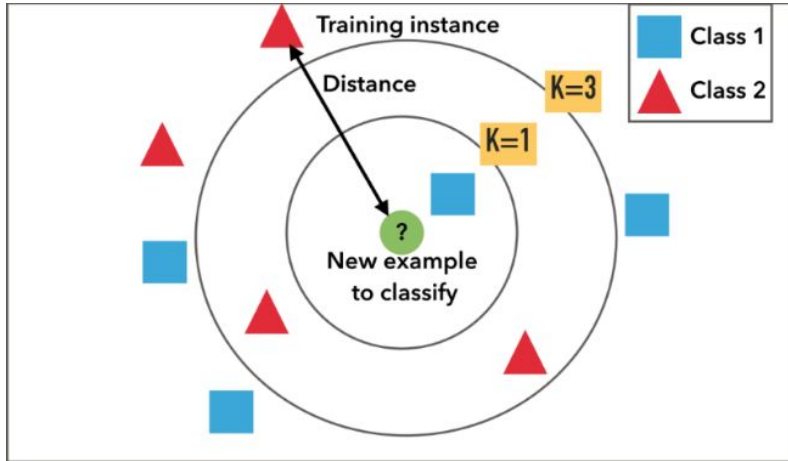
```
Recommendations pour l'utilisateur 54612:  
['Clannad: After Story']  
['Suzumiya Haruhi no Shoushitsu']  
['Tengen Toppa Gurren Lagann']  
['Fate/Zero']  
['Kara no Kyoukai 7: Satsujin Kousatsu (Kou)']
```

```
Recommendations pour l'utilisateur 10421:  
['Fullmetal Alchemist: Brotherhood']  
['Dragon Ball']  
['Tonari no Totoro']  
['Bleach']  
['Kaze Tachinu']
```



## 4.b. Solutions - collaborative filtering - KNN

- $nrows = 1\,000\,000$  !!



- **RMSE : 1.4270**
- **MAE : 1.0975**



## 4.b. Solutions - collaborative filtering - KNN

Recommendations pour l'utilisateur 7235:

- ['Hunter x Hunter']
- ['One Piece Film: Strong World']
- ['Neon Genesis Evangelion']
- ['Death Parade']
- ['Black Lagoon: The Second Barrage']

Recommendations pour l'utilisateur 7915:

- ['Fullmetal Alchemist: Brotherhood']
- ['Code Geass: Hangyaku no Lelouch R2']
- ['Kami nomi zo Shiru Sekai: Megami-hen']
- ['Clannad']
- ['Stranger: Mukou Hadan']

Recommendations pour l'utilisateur 6079:

- ['Ookami Kodomo no Ame to Yuki']
- ['Akira']
- ['Samurai Champloo']
- ['Mushishi Special: Hihamukage']
- ['Ghost in the Shell: Stand Alone Complex 2nd GIG']



## 5. Conclusion