
SYMA: RECSYS CHALLENGE 2022

Moustapha Diop
moustapha.diop@epita.fr

Pierre-Louis Landouzi
pierre-louis.landouzi@epita.fr

Mathieu Rivier
mathieu.rivier@epita.fr

Théo Perinet
theo.perinet@epita.fr

Marc Monteil
marc.monteil@epita.fr

July 2022



Abstract

Le but du challenge Dressipi est de recommander au mieux des objets à des clients sur un site.

Pour ce faire, un jeu de données contenant 1 000 000 sessions client nous est fourni. Chaque session contient un ensemble d'objets vu par le client ainsi qu'un objet acheté par le client à la fin de cette session.

De plus, l'ensemble des attributs de chaque objet nous est fourni sous la forme d'un couple (Id de l'attribut, valeur numérique de l'attribut).

Durant tout le déroulé du projet, la métrique que nous avons tenté d'améliorer est `score`. En effet, c'est cette métrique qui est utilisée pour créer le Leaderbord du site. Pour tester et améliorer rapidement nos algorithmes, nous avons fait un système de validation offline.

De plus, afin de voir les forces et faiblesses de nos implémentations nous avons fait différentes visualisations visibles dans les notebooks.

1 Collaborative Filtering - Page Rank

Nous avons commencé par recommander les objets les plus achetés. Cependant, le problème était que nous ne prenions pas en compte les objets vus par le client durant la session. Ainsi, pour résoudre ce problème, nous avons fait du "Collaborative Filtering" en implémentant l'algorithme de Page Rank avec personnalisation.

En effet, le but du "Collaborative Filtering" est de recommander les objets qui ont plu à des personnes similaire au client actuel. Page Rank est un ancien algorithme utilisé par les moteurs de recherche pour classer/recommander les sites, nous avons décidé de l'implémenter pour recommander des objets.

Après de très nombreux tests, nous avons trouvé qu'il était nécessaire d'exécuter l'algorithme de Page Rank avec certains attributs particuliers. Nous allons en parler par la suite.

Pour la matrice d'adjacence utilisée par l'algorithme, il est nécessaire qu'elle représente un ensemble de liens où chaque objet vu, a un lien avec l'objet acheté en fin de session. Ainsi, elle permet de savoir, en partant d'un objet vu, quels objets ont été achetés par la suite sous forme de probabilité.

Concernant le nombre d'itérations effectuées par l'algorithme, nous avons remarqué qu'il ne fallait en faire qu'une seule pour éviter de recommander des objets trop éloignés de ceux vus durant la session et également pour avoir un résultat "rapidement".

De plus, nous avons observé qu'il fallait prendre d'avantage en compte les derniers objets vus dans une session que les autres pour avoir un meilleur score.

Avec le jeu de données fourni, nous avons réussi à avoir un score offline de 0.178 (en moyenne, la recommandation optimale est en 6ème position) ! En exécutant l'algorithme sur le jeu de données utilisé pour le leaderboard, nous avons eu un score de 0.175, ce qui nous a permis d'arriver à la 46ème place le 17/05/2022 !

2 Content-Based Filtering - Cosine Similarity

Page Rank donne de très bons résultats mais nous avons un problème, Page Rank ne prend pas en compte les caractéristiques/attributs des objets. Ainsi, pour résoudre ce problème, nous avons décidé de faire du "Content-Based Filtering" en implémentant la formule du "Cosine Similarity".

En effet, le but du "Content-Based Filtering" est maintenant de recommander les objets les plus similaires à ceux vus durant la session. La "Cosine Similarity" est une formule permettant de calculer un taux de ressemblance entre deux objets suivant leurs caractéristiques/attributs :

$$\text{similarity}(A, B) = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

source: towards data science article

Dans notre implémentation, nous additionnons la similarité de chaque objet avec chaque objet vu afin de savoir quels objets sont les plus similaires avec ceux vus.

En utilisant cette formule, nous avons réussi à avoir un score offline de 0.054 (en moyenne, la recommandation optimale est en 19ème position) ! Ce qui est un bon résultat au vu du fait que nous nous basons uniquement sur les attributs des objets.

3 Hybride Filtering - Page Rank x Cosine Similarity

Le problème actuel est que nous faisons soit du "Collaborative Filtering" grâce à Page Rank, soit du "Content-Based Filtering" grâce au "Cosine Similarity". Ainsi, nous allons faire du "Hybride Filtering". Il permet de tirer avantage des deux techniques en recommandant des objets suivant ce que les personnes similaires au client actuel ont aimé ainsi que des objets similaires à ceux vus durant la session.

Pour cela, nous multiplions le résultat des deux techniques, afin qu'à partir du résultat de Page Rank, nous puissions faire remonter les objets similaires grâce au "Cosine Similarity".

Grâce à cela, nous obtenons un score offline de 0.18 (en moyenne la recommandation optimale est en 6ème position) ! En exécutant l'algorithme sur le jeu de données utilisé pour le leaderboard, nous avons eu un score de 0.18, ce qui nous a permis d'arriver à la 36ème place le 19/05/2022 ! À la date du 06/07/2022, avec le même score, nous sommes 78ème du classement.

4 Collaborative/Content-Based Filtering - Bayes

Le problème actuel est que, pour le "Content-Based Filtering", nous ne savons pas ce que représentent les attributs des objets. En effet, les attributs sont représentés par deux nombres sans signification. Cela pose problème car certains attributs devraient être plus pris en compte que d'autres lors de la recommandation (ex: type d'objet vs couleur).

Ainsi, nous avons voulu utiliser un prédicteur de Bayes afin qu'il puisse comprendre par lui-même les attributs qui caractérisent au mieux chaque objet. En effet, les prédicteurs de Bayes sont connus par leur capacité à deviner le titre d'un livre en lui passant en entrée un extrait du livre, après entraînement sur plusieurs livres dont celui cherché. Nous avons voulu faire la même chose en considérant les attributs des objets comme les mots de l'extrait et l'objet acheté comme le titre du livre.

Pour cela, nous avons entraîné un modèle `MultinomialNB` de sklearn en lui donnant en entrée la somme de tous les attributs de tous les objets vus dans chaque session, accompagnée de l'id de l'objet acheté. En faisant cela, nous partons du principe que les attributs des objets vus représentent en général l'objet acheté.

Après entraînement et test, nous obtenons un score offline de 0.02 (en moyenne, la recommandation optimale est en 50ème position), ce qui est peu. Cela étant sûrement dû au fait que les objets présents dans la session n'ont pas forcément de rapport avec l'objet acheté.

5 Optimisations

Tout l'enjeu de ce challenge était en réalité de réussir à avoir un bon score dans un temps acceptable tout en étant limité par notre machine. Le PC utilisé à 16 Go de ram avec un processeur Intel core i7 9ème génération. Durant l'exécution des algorithmes, la ram et le processeur sont souvent utilisés à 100 % !

Pour cela, nous avons utilisé massivement les librairies pandas et numpy afin que les opérations soit les plus optimisées possible, rapides et ne prenant pas trop de place en ram.

Afin de travailler rapidement et surtout pour ne pas avoir de problème de mémoire pleine, nous avons:

- Utilisé `partial_fit()` pour entraîner le modèle `MultinomialNB` afin que l'ensemble des données rentre dans la ram. Malheureusement, cela diminue la performance du modèle.
- Fais un système de batch pour l'algorithme de Page Rank pour ne pas avoir de problème de mémoire pleine.
- Nous avons également apporté un système de sauvegarde des tableaux utilisés régulièrement afin de gagner du temps en période de test/debug.

6 Améliorations

Voici quelque piste d'amélioration de notre rendu:

- Faire des meilleures visualisations pour mieux voir les forces et faiblesses de nos algorithmes.
- Utiliser Spark afin de ne pas avoir à faire des optimisations manuellement pour que les données rentrent en mémoire. Cella permettrait également de répartir la charge sur plusieurs ordinateurs.
- Tester l'utilisation d'un modèle d'IA pour améliorer les résultats.

7 Leaderboard et scores

Voici une capture d'écran de l'ensemble de nos push sur le site:

Your Profile

[Download dataset](#)

By Date		By Score		
Id	Notes	Score	Uploaded at	Rank
fb19293e-5257-47c8-8d10-08cdf14e4bb7	hybride model : page rank + cosinus similarities	0.1800994917099798	18 May 22:28	2576
f993052f-137a-4aca-b698-560ca18c33d7	hybride model : page rank + cosinus similarities please not fail this time	0.1800994917099798	18 May 23:09	2577
c0b9d0e4-1383-4fe6-af70-2dcea207d5b6	better page rank	0.17709008353595299	19 May 12:26	2805
1cab1414-5ed5-4fa4-9bb4-482f115b102a	better	0.1759403824031681	19 May 12:46	2895
5cb5e3e0-9d30-4cc2-aac7-bd9678b8207a	with candidates	0.17455708029371814	17 May 15:54	2990
65ff1633-b26d-480d-bf77-901eedc3a4fb	Page rank with last seen better for predict	0.17179152005999335	16 May 11:35	3133
cc9f4b54-0e7c-4313-a963-0420c458ca3c	Page rank with last seen better for predict but with beter g	0.17175554558583575	16 May 15:16	3137
0a618343-0737-4457-b34e-cdecfb907e52	best page rank	0.16796226074695944	16 May 08:47	3468
70483c3e-Saba-4238-a02a-b1f19b9f1e78	A bad page rank (bad adjacent matrix)	0.1592577035920073	11 May 19:01	4381
38fe9dd5-4b55-47a6-89bd-7f970da649eb	Stupid page rank maybe bugged	0.15197989933354367	11 May 13:38	4641
ae8ad5f9-15ff-4a69-85f1-ba4103390c81	why failed ?	0.07950715920290068	18 May 22:52	5761
feffe11ee-a1dd-4c8f-bf71-79fb3f9889fa	Better page rank (5 iterations) maybe bugged	0.014972212125116808	11 May 14:23	6680

Figure 1: Pushes

Voici des capture d'écran de notre rang sur le leaderboard suivant la date:



45	Extendi.srl	0.1767199230773437
46	 EPITA#1 	0.17455708029371814
47	Alexdruso	0.1727663566376727

Figure 2: Leaderboard le 17/05/2022



34	太阳照常升起	0.18042769101645245
35	Rise	0.18029775938998385
36	 EPITA#1 	0.1800994917099798
37	ZZteam	0.18006271199341659
38	League of Legends	0.17937516749267357

Figure 3: Leaderboard le 19/05/2022



71	xiaokang	0.18225206547214817	June 07, 2022 01:15
72	ESTU	0.18212972811954178	June 13, 2022 12:45
73	XITU is all you need	0.18169322715829214	April 01, 2022 02:58
74	NvGTR	0.18150234759584488	April 06, 2022 15:57
75	太阳照常升起	0.18042769101645245	May 16, 2022 03:42
76	bvcvcoovsa	0.18041042417048017	June 04, 2022 13:30
77	Rise	0.18029775938998385	May 15, 2022 11:33
78	 EPITA#1 	0.1800994917099798	May 18, 2022 23:09
79	Adia	0.17994125522629792	June 14, 2022 18:10
80	sakatani	0.17965046148153965	May 30, 2022 10:16
81	dylan	0.1786895885374489	March 23, 2022 06:01
82	baoziformer	0.17855570598194975	May 11, 2022 18:53
83	canon	0.17831690775805834	April 04, 2022 16:41
84	ExtendI.srl	0.17827511072671706	May 22, 2022 16:35
85	BBL	0.1780985708847418	July 01, 2022 03:40

Figure 4: Leaderboard le 06/07/2022