

Projet IA 316



IMPLÉMENTATION D'UN SYSTÈME DE RECOMMANDATION

Mariem Kachouri Dinara Veshchezerova



Sommaire

Introduction

Problématique 3ème Environnement

Méthodologie

Résultats

Conclusion et perspectives

INTRODUCTION

PROJET IA 316

Système de recommandation avec
feedback explicite

Système de recommandation avec
feedback Implicite

Environnement 1

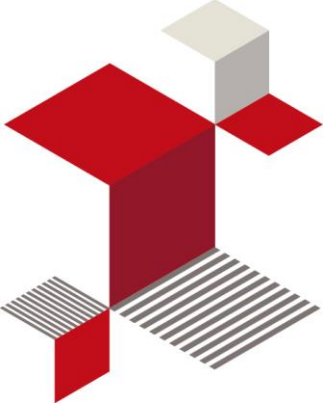
- Modèle Matrix Factorization
- Modèle deep learning avec produit scalaire sans tenir compte des métadonnées.

Environnement 2

- Modèle deep learning avec introduction des covariates
- Création d'une API

Environnement 3

- Modèle Siamois avec Triplet Loss (avec et sans covariates)
- Gestion des nouveaux utilisateurs
- Introduction d'un nouveau modèle
- Modèle Hybride combinant deux approches différentes.



Sommaire

Introduction

Problématique 3ème Environnement

Méthodologie

Résultats

Conclusion et Perspectives

PROBLÉMATIQUE

FOCUS 3^{ÈME} ENVIRONNEMENT

Contexte

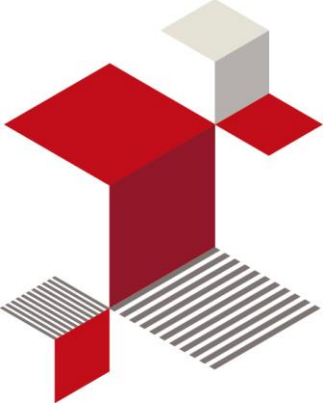
- Pas d'historique d'achats pour un user donné (état initial)
- A chaque appel de la requête 'predict' , on accède à un nouvel état avec la liste d'items disponibles et un seul user id.
- Gestion des nouveaux utilisateurs (pas vus lors de l'apprentissage) : *Cold Start Issue*

But

- Maximiser la récompense moyenne sur 1000 prédictions.
- Maximiser le taux de conversion (taux de prédictions réussies aboutissant à l'achat de l'item recommandé) sur 1000 prédictions.

Approche 1

Approche 2



Sommaire

Introduction

Problématique 3ème Environnement

Méthodologie

Résultats

Conclusion et Perspectives

MÉTHODOLOGIE

PRÉSENTATION DES APPROCHES ADOPTÉES



1^{ère} approche

- **RN Siamois avec Triplet Loss** avec / sans métadonnées (user et items)
- Utilisation de couches fully connected pour calculer les similarités positive et négative en input de la triplet loss.
- Se baser sur l'état initial pour identifier des **triplets { user, item +, item - }** et construction des données +.

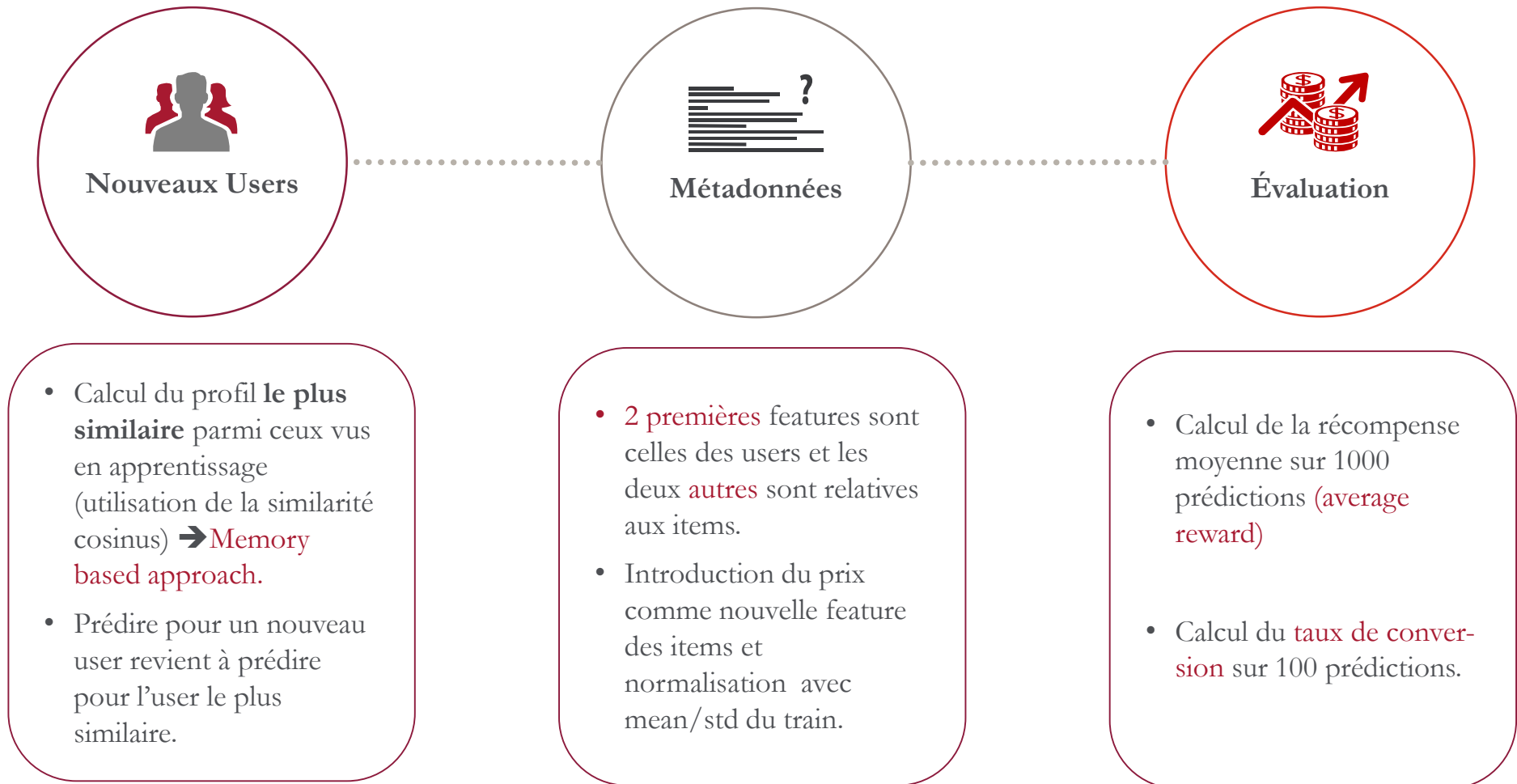


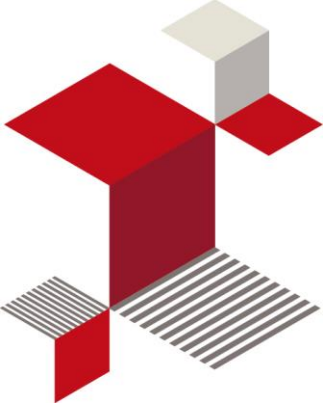
2^{ème} approche

- Introduction d'un nouveau modèle **LightFM avec Warp Loss**.
- Combinaison des résultats de deux modèles différents : LightFM avec Warp Loss et RN Siamois avec Triplet.
- Recommander **l'item le plus cher ou moins cher** en comparant les deux prédictions.

MÉTHODOLOGIE

POINTS EN COMMUN DES DEUX APPROCHES





Sommaire

Introduction

Problématique 3ème Environnement

Méthodologie

Résultats

Conclusion et Perspectives

RÉSULTATS

RÉSULTATS EN FONCTION DES VALEURS DE LA MARGE

1st model

margin	avg reward	% of postitive
0.5	132.82349842332832	33.4
1	115.27638840276973	32.7
1.5	154.3926711921762	30.9
2	119.7954770912971	27.9
2.5	123.47391156653173	23.1
3	137.85805247163256	25.8
3.5	129.54683952594434	23.3

2nd model

margin	avg reward	% of postitive
0.5	91.20541694875492	20.8
1	169.34868136789063	31.2
1.5	86.15808849619859	30.09
2	59.38050090655821	22.6
2.5	93.63423404862137	23.40
3	99.31253386815727	24.9
3.5	129.38822175934718	23.0

Hybrid model

margin	avg reward	% of postitive
0.5	212.89257648123777	32.5
1	195.70383298090158	27.0
1.5	168.80267799513533	26.20
2	129.62339979739562	23.1
2.5	108.08911179481099	22.90
3	110.22737259594653	23.5
3.5	91.52149443484106	20.59

3rd model

margin	avg reward	% of postitive
0.5	117.42578061727096	19.7
1	121.62543696438043	24.9
1.5	169.73302930663246	28.1
2	113.18605313848596	24.0
2.5	111.52637722525041	17.8
3	87.66752154725002	18.8
3.5	107.42923167915677	19.2

4th model

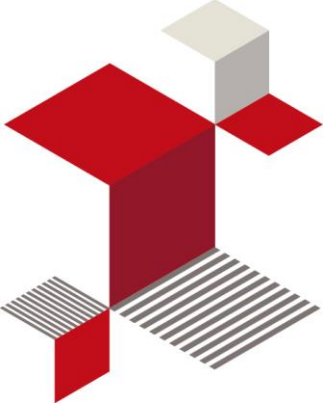
margin	avg reward	% of postitive
0.5	157.17093943610718	34.30
1	150.76404292177008	28.19
1.5	121.38626228014208	22.6
2	98.01896432391887	19.2
2.5	80.01904241963929	16.8
3	78.6337802661393	18.5
3.5	95.7018348709514	22.90

RÉSULTATS

FOCUS MODÈLE HYBRIDE

```

Test for margin: 0.5      |=====>
    Average reward:  212.89257648123777
    Percentage of positive rewards:  32.5 %
    Nb times we recommended based on LightFM results:  273
    Nb times the switch in predictions was a success:  31.135531135531135
Test for margin: 1        |=====>
    Average reward:  195.70383298090158
    Percentage of positive rewards:  27.0 %
    Nb times we recommended based on LightFM results:  230
    Nb times the switch in predictions was a success:  22.608695652173914
Test for margin: 1.5      |=====>
    Average reward:  168.80267799513533
    Percentage of positive rewards:  26.200000000000003 %
    Nb times we recommended based on LightFM results:  554
    Nb times the switch in predictions was a success:  27.79783393501805
Test for margin: 2        |=====>
    Average reward:  129.62339979739562
    Percentage of positive rewards:  23.1 %
    Nb times we recommended based on LightFM results:  772
    Nb times the switch in predictions was a success:  22.797927461139896
Test for margin: 2.5      |=====>
    Average reward:  108.08911179481099
    Percentage of positive rewards:  22.900000000000002 %
    Nb times we recommended based on LightFM results:  846
    Nb times the switch in predictions was a success:  24.349881796690305
  
```



Sommaire

Introduction

Problématique 3ème Environnement

Méthodologie

Résultats

Conclusion et Perspectives

CONCLUSION & PERSPECTIVES

Conclusion

- Modèle Hybrid = Content Based Approach (**RN Siamois**) + Collaborative filtering (**Light FM**) + Memory Based Approach (**Nouveaux users**)
- Recommander l'item le plus cher avec **marge = 0.5** :
 - ➔ Meilleur récompense en moyenne et meilleur taux de conversion

Perspectives

- Avoir **plus de données**
 - ➔ Construction d'un historique d'achat pour chaque user (*filtrage collaboratif basé sur le comportement*)
- Tester des modèles basés sur le **Reinforcement Learning/Online Learning**
- Vote à la majorité sur les profils de users les plus similaires (*cold start issue*)

Merci pour votre attention



ANNEXE 1: TYPES DE FILTRAGE COLLABORATIF

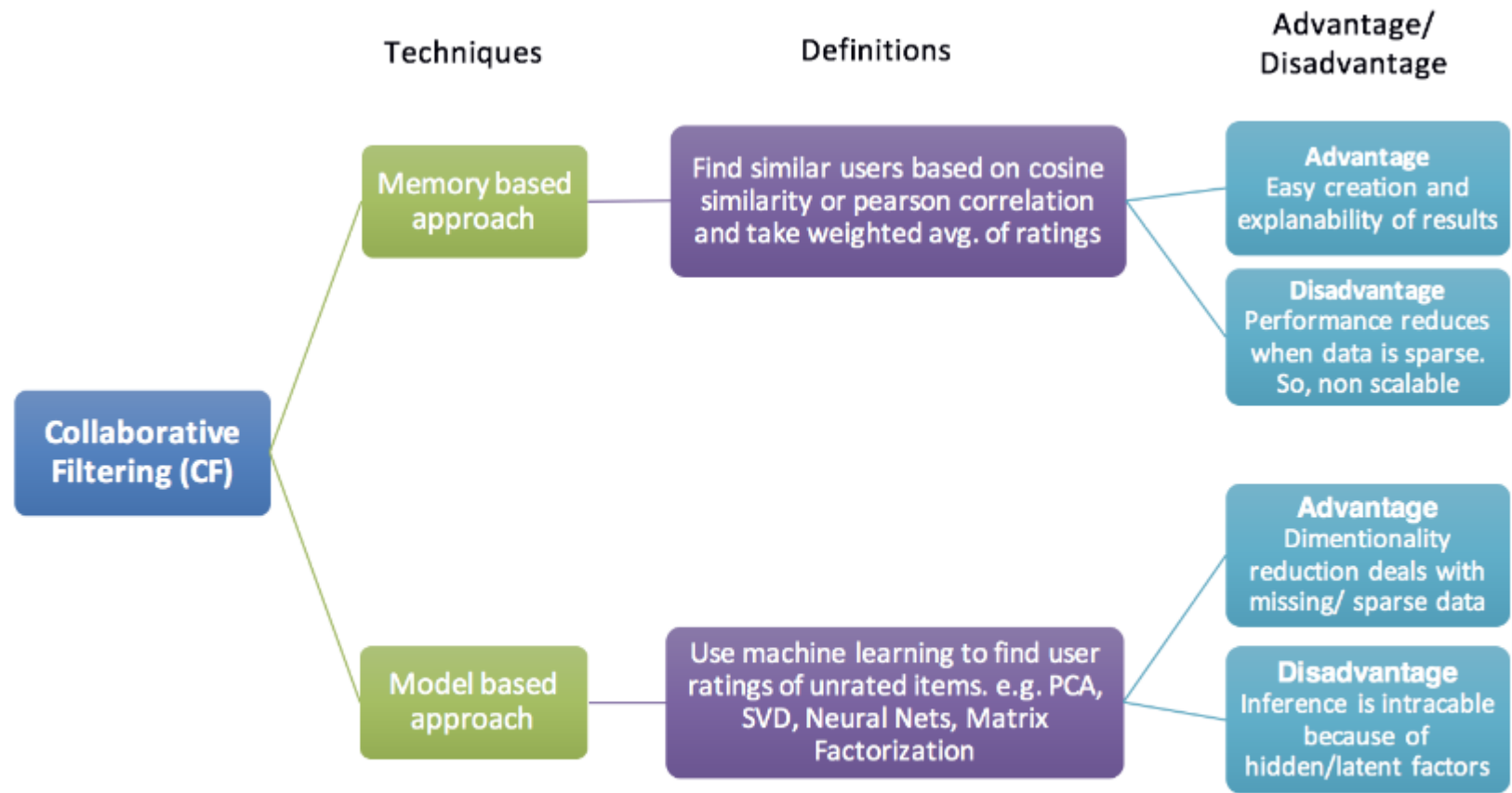


Figure 2: Types of collaborative filtering approaches. Reference: [Wikipedia](#)

ANNEXE 2: FORMULES DES FONCTIONS DE COUT

WARP LOSS – TRIPLET LOSS

$$\mathcal{L} = \max(d(a, p) - d(a, n) + \text{margin}, 0)$$

Triplet Loss Function

WARP LOSS:

For a given (user, positive item pair):

1. Sample a negative item at random from all the remaining items.
2. Compute predictions for both items

If the negative item's prediction > prediction positive item plus a margin:

Perform a gradient update to rank the item (+) higher and the item (-) lower.

If there is no rank violation:

Continue sampling items (-) until a violation is found.