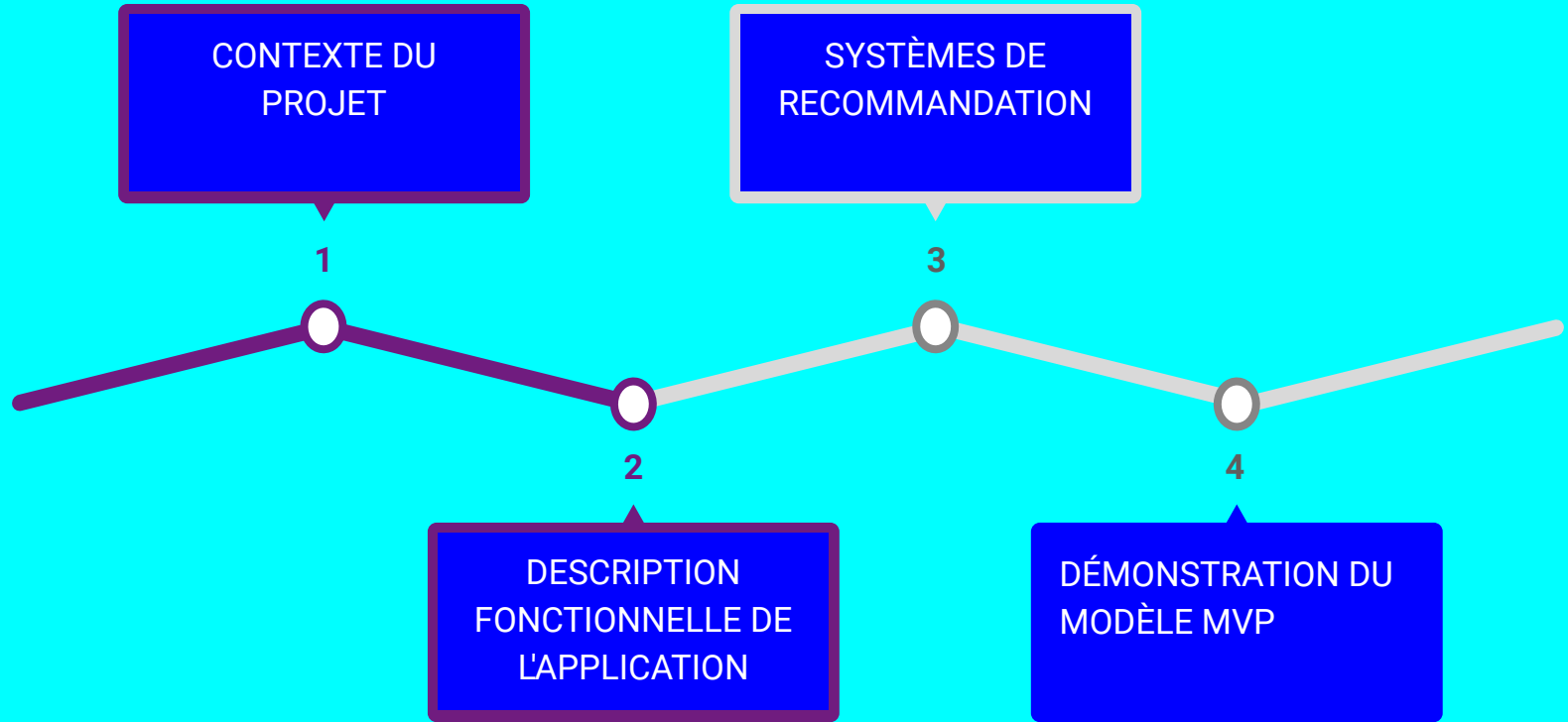


Application mobile de recommandation de contenu



SOMMAIRE



1. CONTEXTE DU PROJET



La société **My Content** souhaite encourager et promouvoir la lecture même chez les plus jeunes en concevant une application mobile. Un premier **MVP (Minimum Viable Product)** a été développé pour la recommandation d'articles et de livres pour des utilisateurs existants et futurs, et ce avec une architecture serverless d' **Azure Function**.

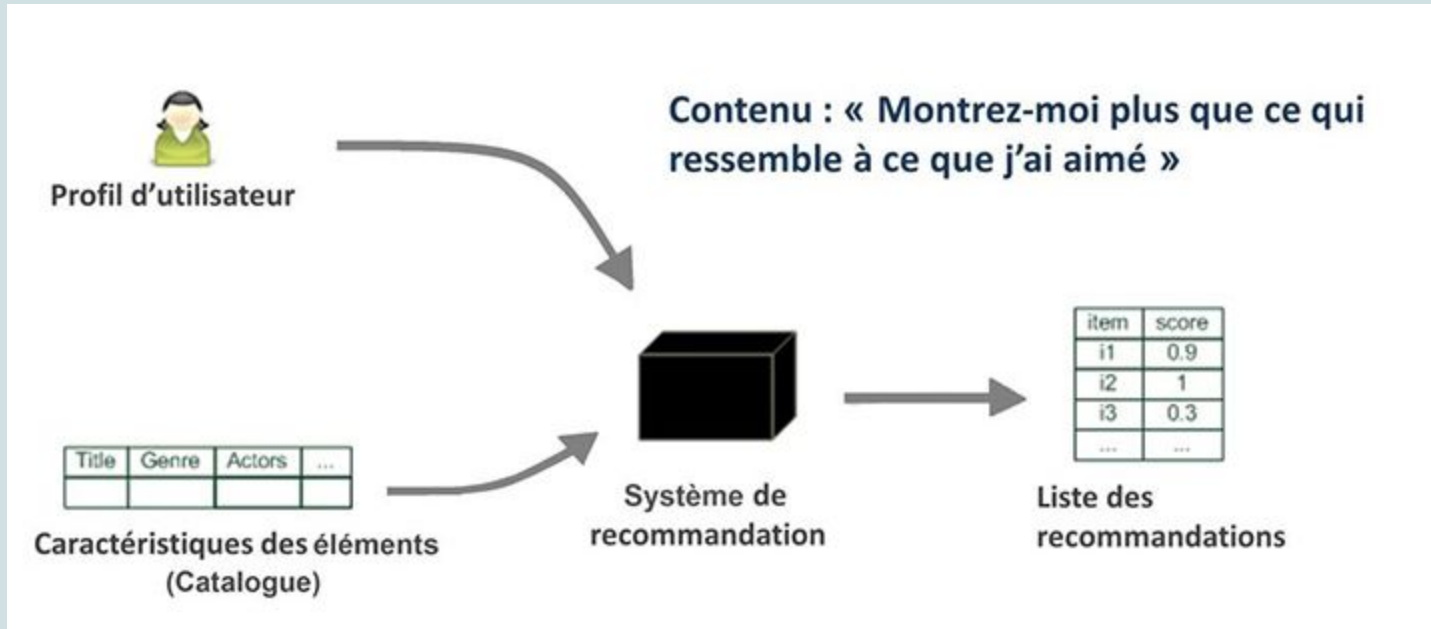
2. DESCRIPTION FONCTIONNELLE DE L'APPLICATION

On part d'un jeu de données composé d'un fichier du log des clicks utilisateur sur les articles du site **Globo.com** ainsi que le fichier des articles contenant des informations sur la catégorie, l'id de l'article, le nombre de mots décrivant l'article



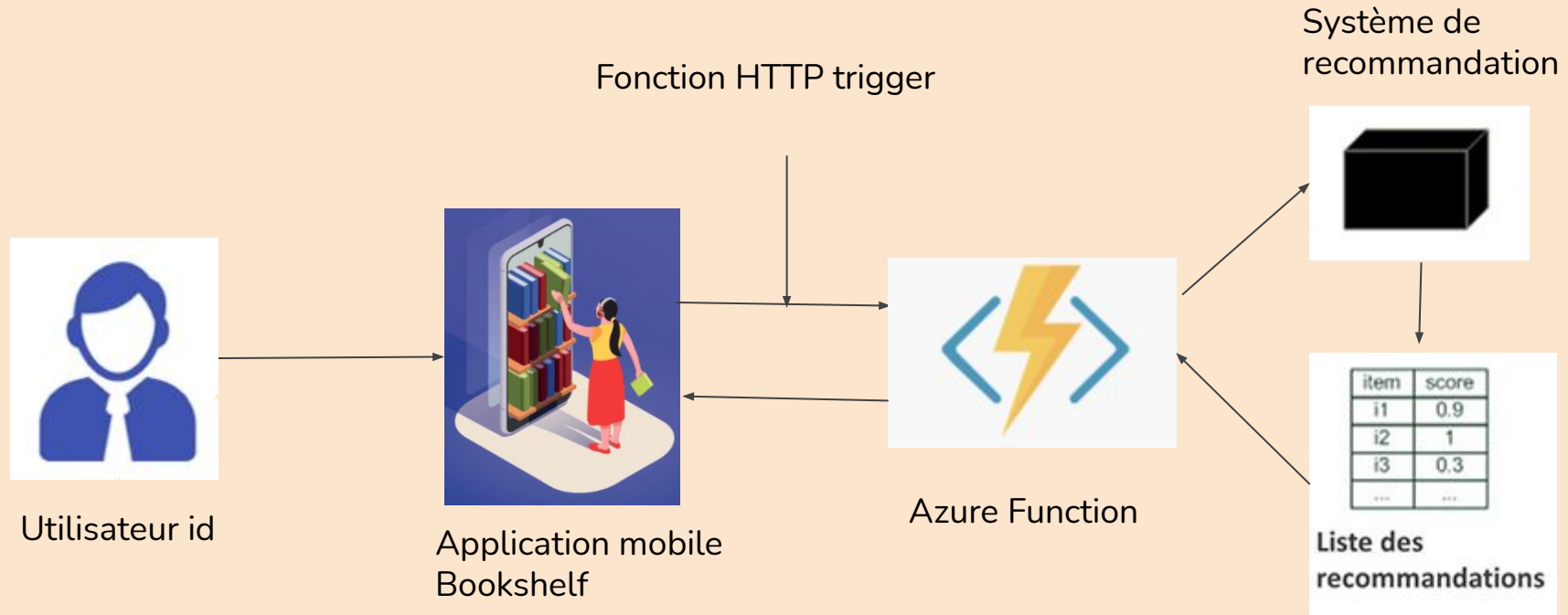
Afin d'alimenter notre modèle de recommandation, on peut définir un scoring des articles par catégorie

Schéma fonctionnel du système de recommandation



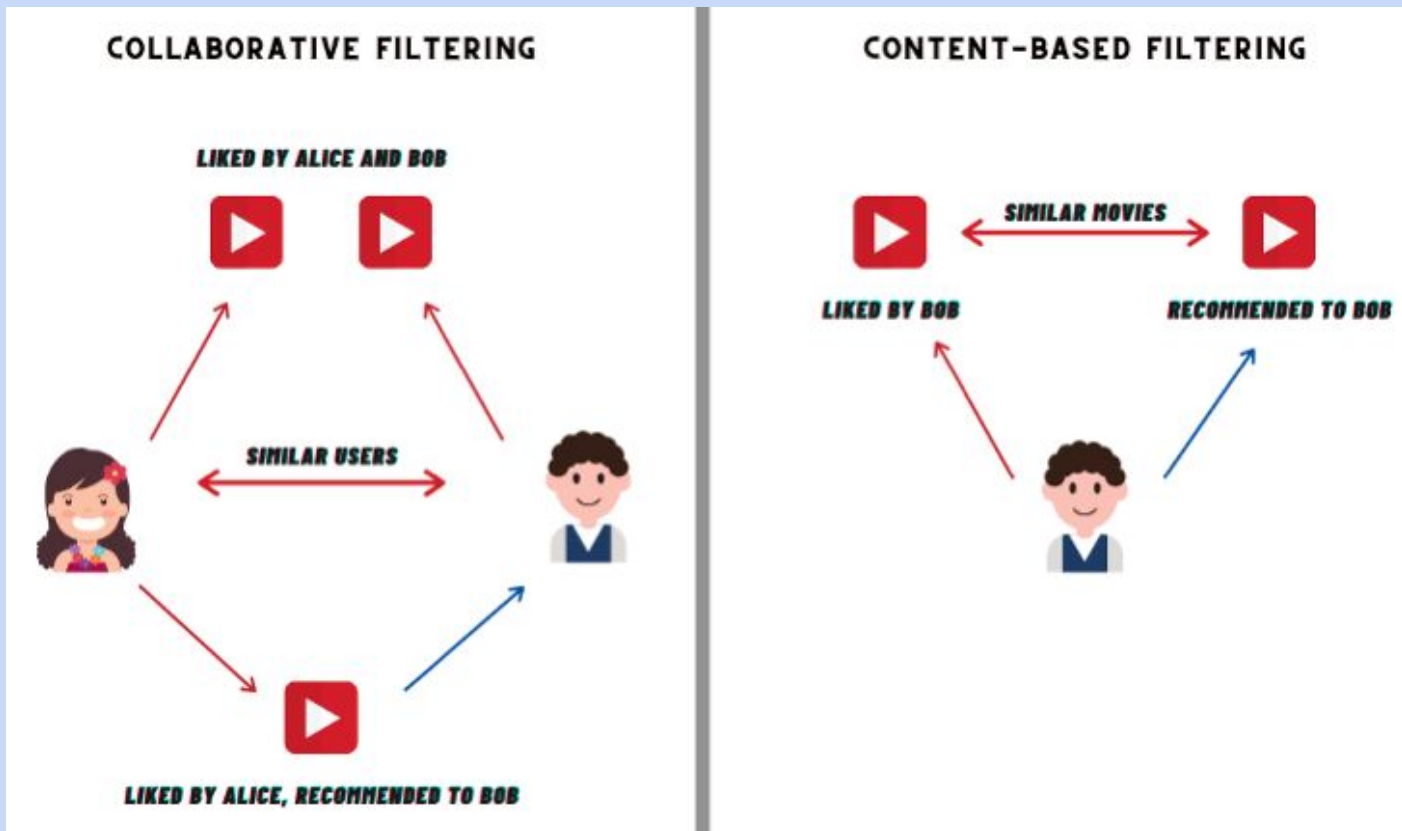
Un système de recommandation peut être construit sur des modèles populaires tels que la recommandation basée sur le contenu des articles ou celle basée sur les préférences ou informations de goût des utilisateurs

Architecture du modèle MVP

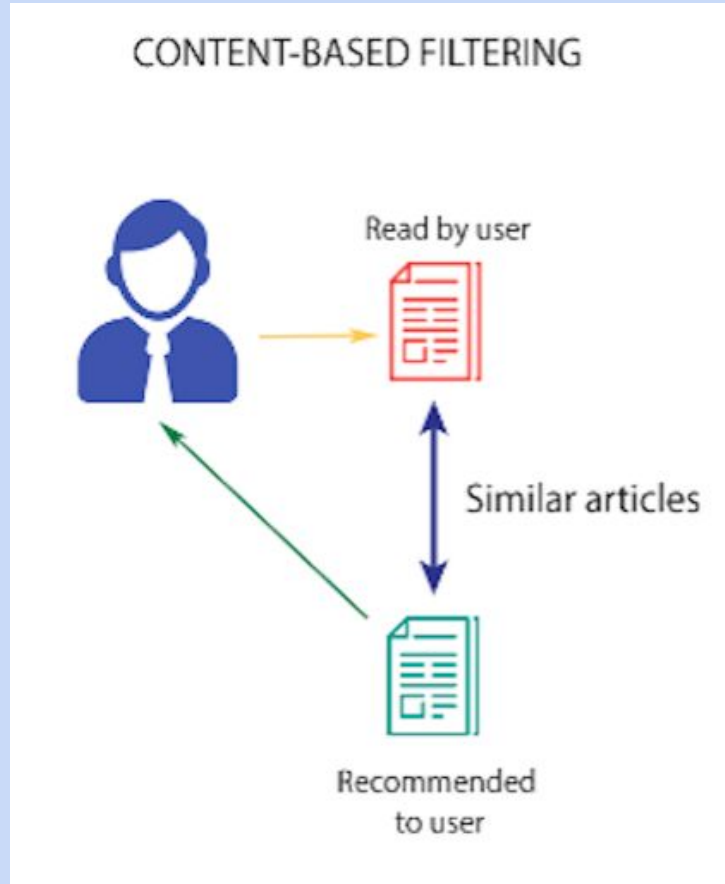


L'identification d'un utilisateur défini par son **id** déclenche un évènement **Http trigger** qui va appeler la fonction **Azure** par l'intermédiaire de l'uri (<https://recommandation.azurewebsites.net/api/httptrigger1>). Dans celle-ci, le système de recommandation de contenu renvoie une liste de 5 articles adaptée à l'utilisateur identifié.

3. SYSTÈMES DE RECOMMANDATION



Représentation du modèle Content-Based Filtering



La recommandation basée sur le contenu exploite la description des articles avec lesquels l'utilisateur a interagi afin de recommander des articles similaires. Cela ne dépend que des choix précédents de l'utilisateur, ce qui rend cette méthode robuste pour éviter le problème de démarrage à froid.

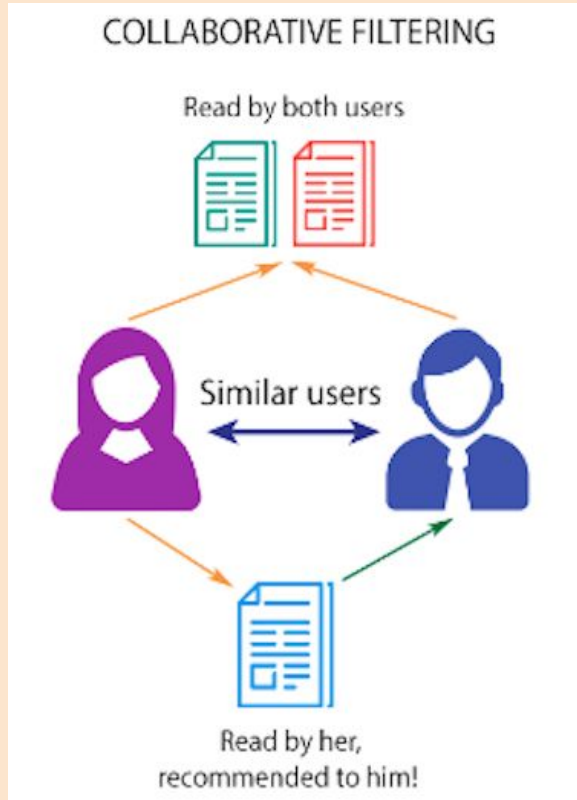
Résultat du modèle en exploitant la matrice embedding des articles

```
[25, 462, 423, 1473, 1419, 588, 473, 958, 576, 693]
```

Voici la liste des 10 articles similaires lus et vus par l'utilisateur 0.

La méthode **TF-IDF** est la plus couramment utilisée pour convertir l'article en une structure vectorielle, où chaque mot est représenté par une position dans le vecteur, et la valeur mesure la pertinence d'un mot donné dans cet article

Représentation du modèle Collaborative Filtering



La recommandation basée sur la **collaboration** fait des prédictions automatiques (filtrage) sur les intérêts d'un utilisateur en collectant les préférences ou les informations de goût de nombreux utilisateurs. L'hypothèse sous-jacente de l'approche de filtrage collaboratif est que si une personne A a la même opinion qu'une personne B sur un ensemble d'éléments, A est plus susceptible d'avoir l'opinion de B pour un élément donné que celle d'une personne choisie au hasard

Performance du modèle kNNWithMeans

Les modèles sont développés à l'aide de différents algorithmes d'apprentissage automatique pour recommander des éléments aux utilisateurs. Il existe de nombreux algorithmes CF basés sur des modèles, tels que les réseaux de neurones, les réseaux bayésiens, les modèles de clustering et les modèles de facteurs latents tels que la décomposition de valeur singulière (SVD) et l'analyse sémantique latente probabiliste

```
9.04694921100913
```

```
{'sim_options': {'name': 'cosine', 'min_support': 5, 'user_based': False}, 'k': 10, 'min_k': 3}
```

```
Evaluating RMSE, MAE of algorithm KNNWithMeans on 3 split(s).
```

	Fold 1	Fold 2	Fold 3	Mean	Std
RMSE (testset)	10.3729	8.5198	8.1432	9.0120	0.9745
MAE (testset)	4.7079	4.6061	4.5105	4.6082	0.0806
Fit time	23.80	23.40	22.17	23.12	0.69
Test time	11.75	12.05	12.14	11.98	0.16

```
{'fit_time': (23.796727895736694, 23.39501667022705, 22.17151403427124),  
 'test_mae': array([4.7078935 , 4.60612373, 4.5104795 ]),  
 'test_rmse': array([10.37288533,  8.5198167 ,  8.14321407]),  
 'test_time': (11.754831314086914, 12.050522089004517, 12.14085054397583)}
```

Performance du modèle SVD

Il s'agit d'un algorithme de factorisation de matrice les plus utilisés.
L'algorithme **SVD** est efficace car il permet de trouver des variables latentes qui lient les utilisateurs et les articles

```
9.02624979339465  
{'n_epochs': 5, 'lr_all': 0.002, 'reg_all': 0.4}
```

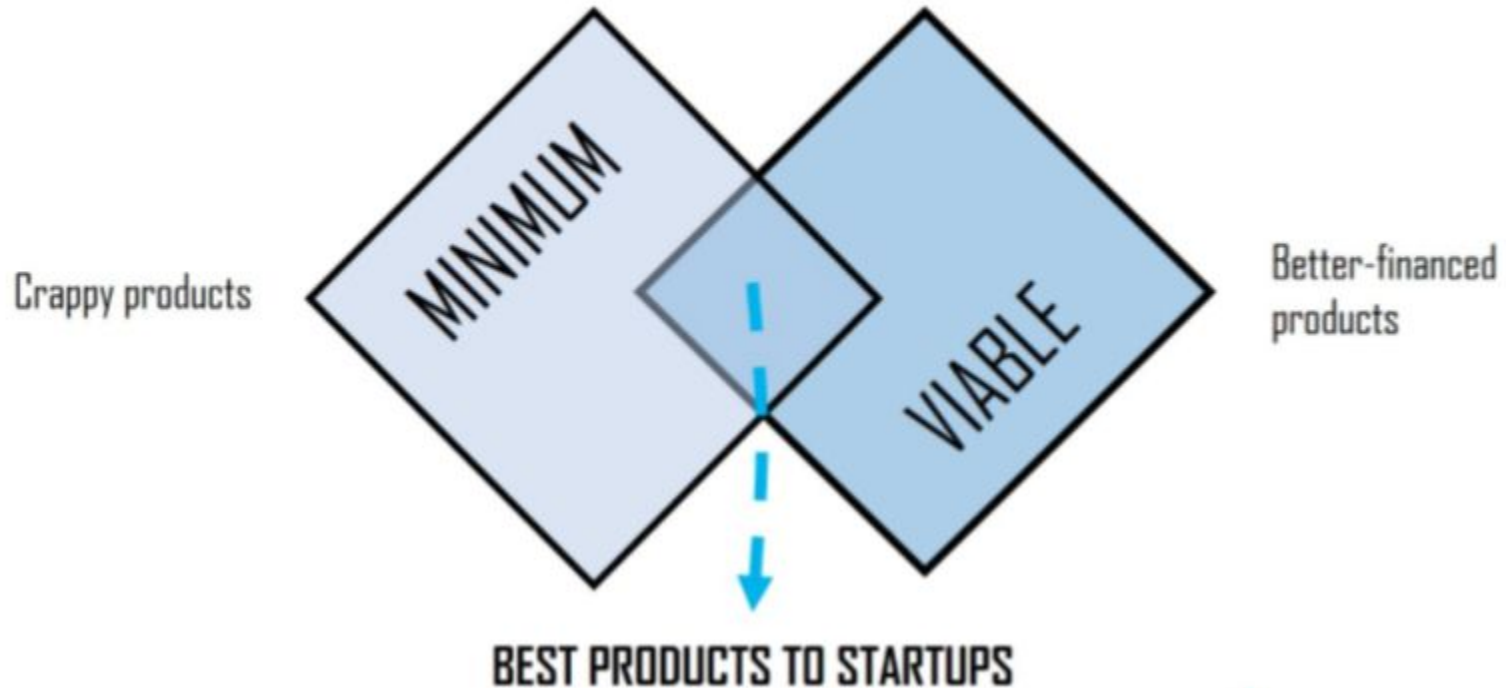
Evaluating RMSE, MAE of algorithm SVD on 3 split(s).

	Fold 1	Fold 2	Fold 3	Mean	Std
RMSE (testset)	9.7908	8.5266	8.7991	9.0388	0.5433
MAE (testset)	4.6473	4.5686	4.5654	4.5938	0.0379
Fit time	1.52	1.47	1.49	1.49	0.02
Test time	0.10	0.09	0.10	0.10	0.00

```
{'fit_time': (1.5248782634735107, 1.4735164642333984, 1.486208200454712),  
 'test_mae': array([4.64730263, 4.56855263, 4.56543194]),  
 'test_rmse': array([9.79084226, 8.52660696, 8.79905581]),  
 'test_time': (0.09808540344238281, 0.09408044815063477, 0.09716606140136719)}
```

4. DÉMONSTRATION DU MODÈLE MVP

MINIMUM VIABLE PRODUCT

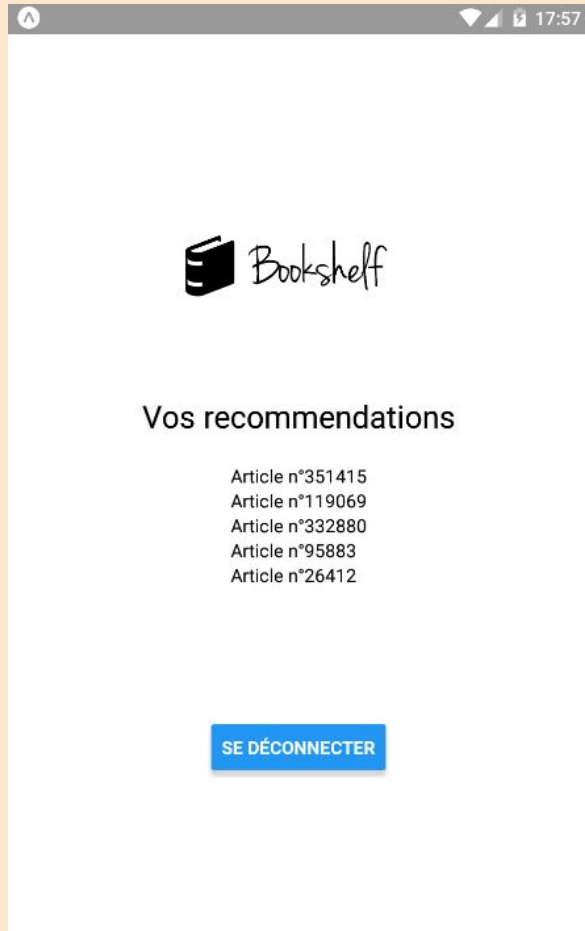


Modèle fonctionnel de recommandation d'articles et de livres



L'application **Bookshelf** permet de recommander une liste de 5 articles pour un utilisateur identifié par son **userId**.

Résultat de l'application Bookshelf de recommandation



Notre moteur de recommandation est basé sur le **Collaborative Filtering (ou CF)** qui a donné des meilleures performances. Ce modèle est aussi le plus adapté à l'ajout de nouveaux utilisateurs et d'articles.

