Group 2

03/05/2024

Movie Recommendation System

FINAL PROJECT - DATA BUSINESS ANALYSIS

LECTURER

Phd. Tuan Minh Pham

PRESENTED BY

Group 2

Team
MEMBERS

Nguyen Minh Tu

ID: 11216040

Le Hoang Anh Duc

ID: 11219268

Vu Trong Manh

ID: 11213752

Tran Hai Nam

ID: 11219279

Nguyen Ha Phuong

ID: 11219284

Outline

1 Introduction

2 Literature Review

3 Methodology

4 Conclusion

Introduction

Problem Definition	Motivation	Problem Setting	Pipeline

a. problem definition

- **A recommender system is a system intended to suggest relevant items to the users.**

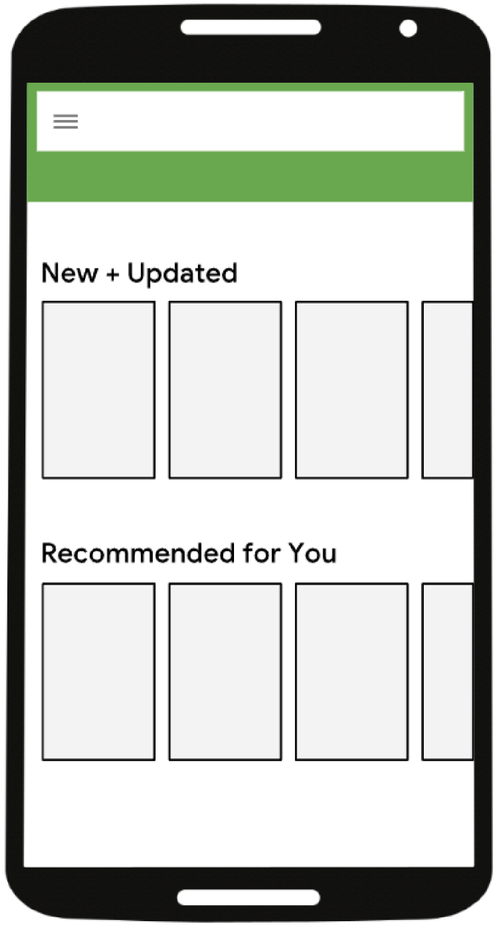
- **Essential of Recommnedation System in real world:**

  - Users get recommendations for items they might be interested in, which saves them time and effort.

  - Service providers gain more profit from users who are more loyal because they are getting recommendations that they like

b.

motivation



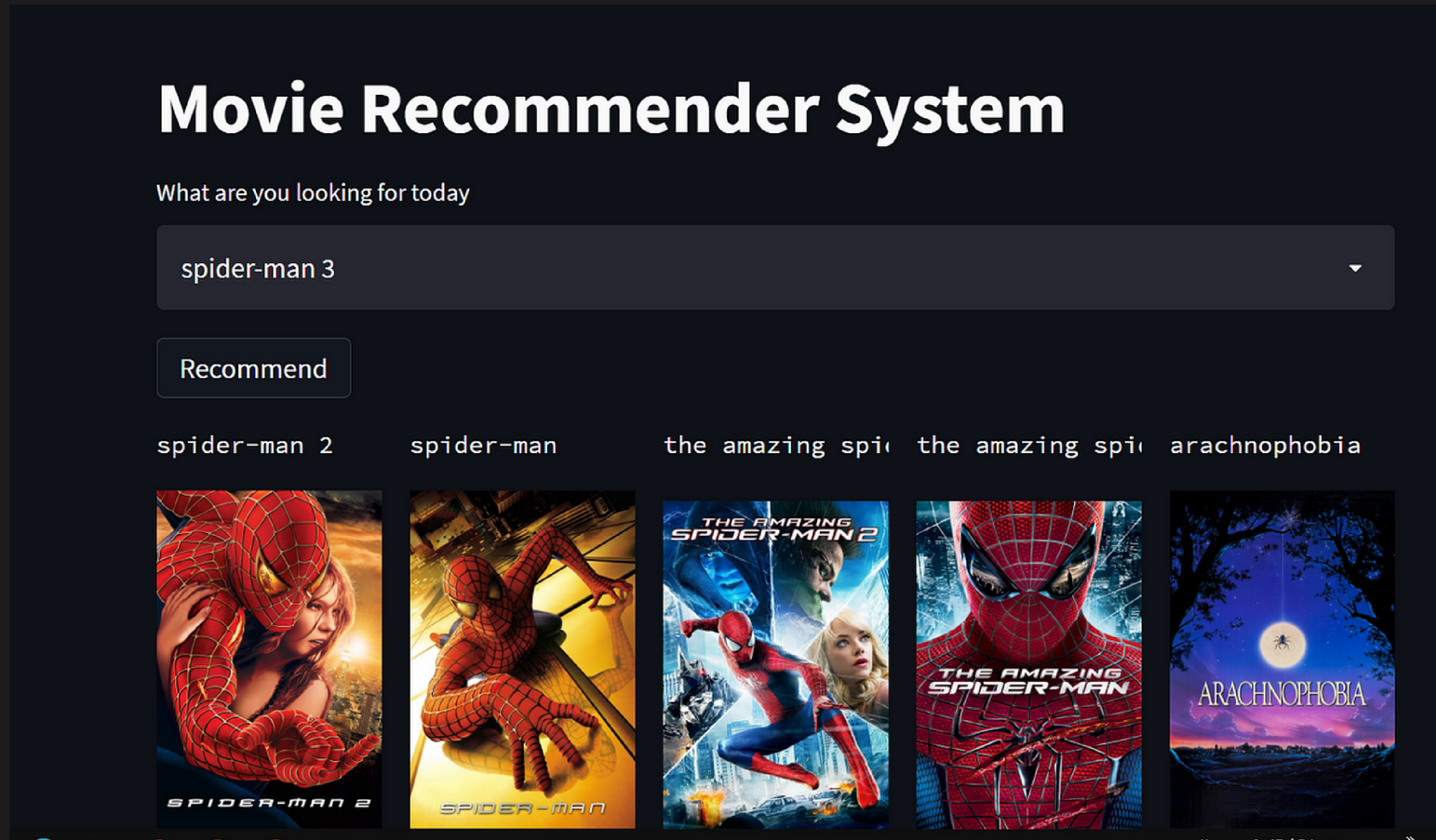
According to McKinsey, recommendations play a crucial role in:

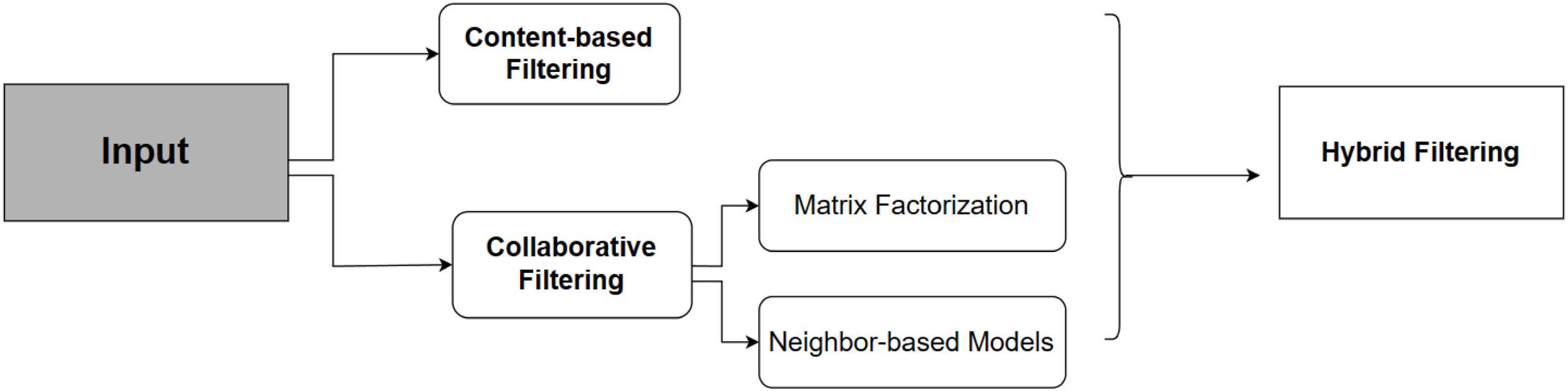
- 40% of app installs on Google Play
- 60% of watch time on Youtube
- 35% of purchases on Amazon
- 75% of movies watched on Netflix

c. problem setting

- Dataset: The Movie Dataset
  26 million ratings (1-5) from 270,000 users for all 45,000 movies

- Goal:
  The primary objective of this project is to provide users with personalized movie recommendations based on their preferences, using a combination of different filtering techniques.





d. project pipeline

Movie Recommendation System

Literature Review

Lops et al. (2011) provide a comprehensive overview of content-based recommender systems, emphasizing their ability to analyze item attributes and user preferences to generate personalized recommendations. These systems rely on techniques such as **keywords extraction, similarity computation,** and recommendation generation to match items with user preferences.

According to **Y. Koren(2009)**, author introduced matrix factorization as a powerful technique for collaborative filtering in recommender systems.

- The user-item interaction matrix is decomposed into lower-dimensional matrices, capturing latent factors underlying user preferences and item characteristics.
- **Singular Value Decomposition (SVD) is introduced as a matrix factorization method used in collaborative filtering.**

Literature Review

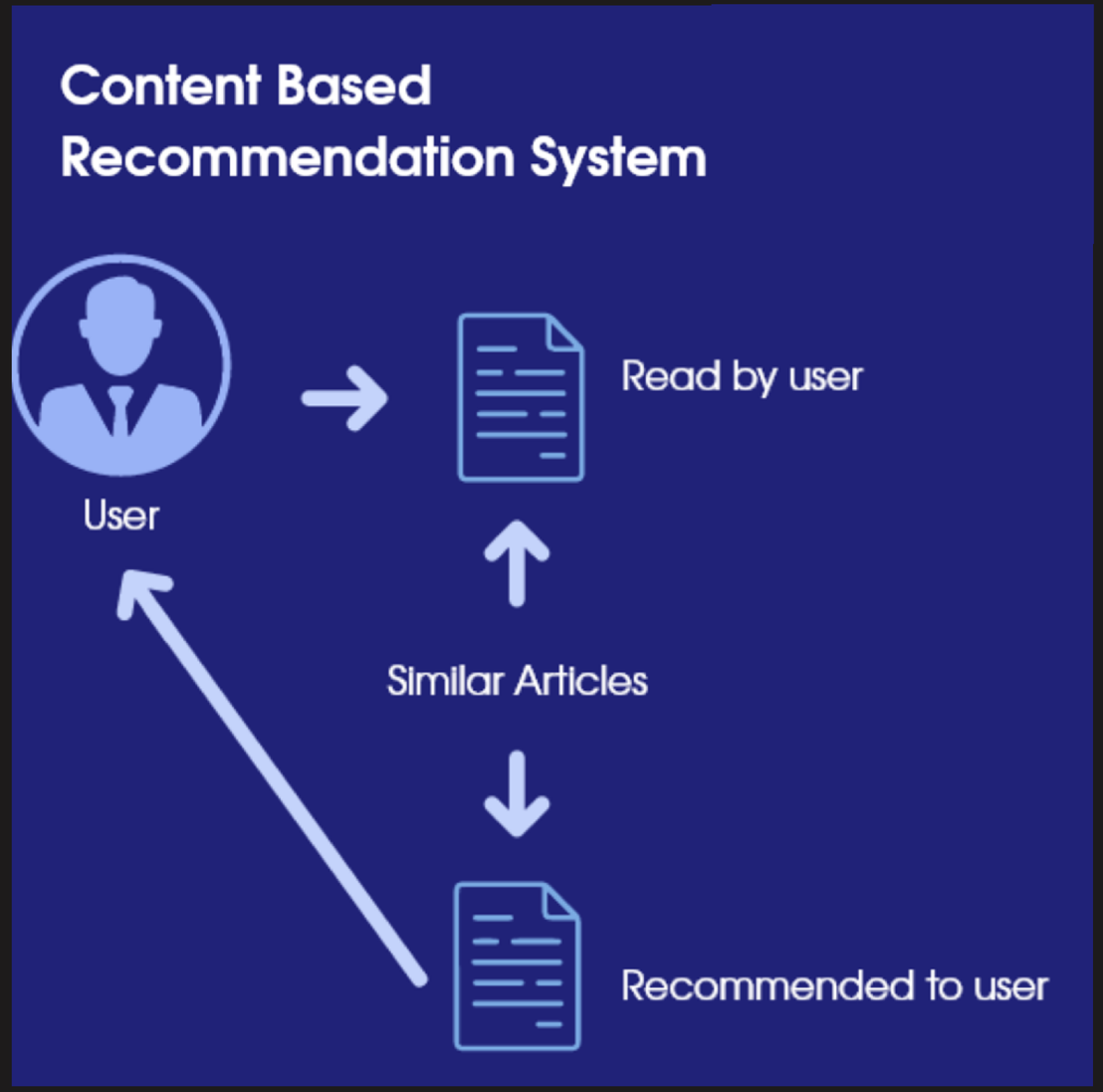
- **The main objective of this research:**

The overarching objective of this paper is to **develop a comprehensive movie recommendation system** that **integrates content-based filtering with collaborative filtering.**

- A content-based recommender will be built to provide baseline recommendations.
- Then, collaborative filtering techniques, such as KNN and SVD, will be incorporated to enhance accuracy and relevance further.

Methodology

- Content-based Filtering
- Collaborative Filtering
- Hybrid Filtering



Methodology

Content-based

Content-Based Filtering recommends movies by analyzing the content and features of each movie. This method takes into account user preferences and suggests movies with similar characteristics to those the user has enjoyed in the past.

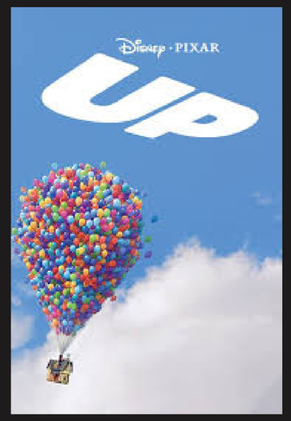
Content-based Filtering

$$W = \frac{R \cdot v + C \cdot m}{v + m}$$

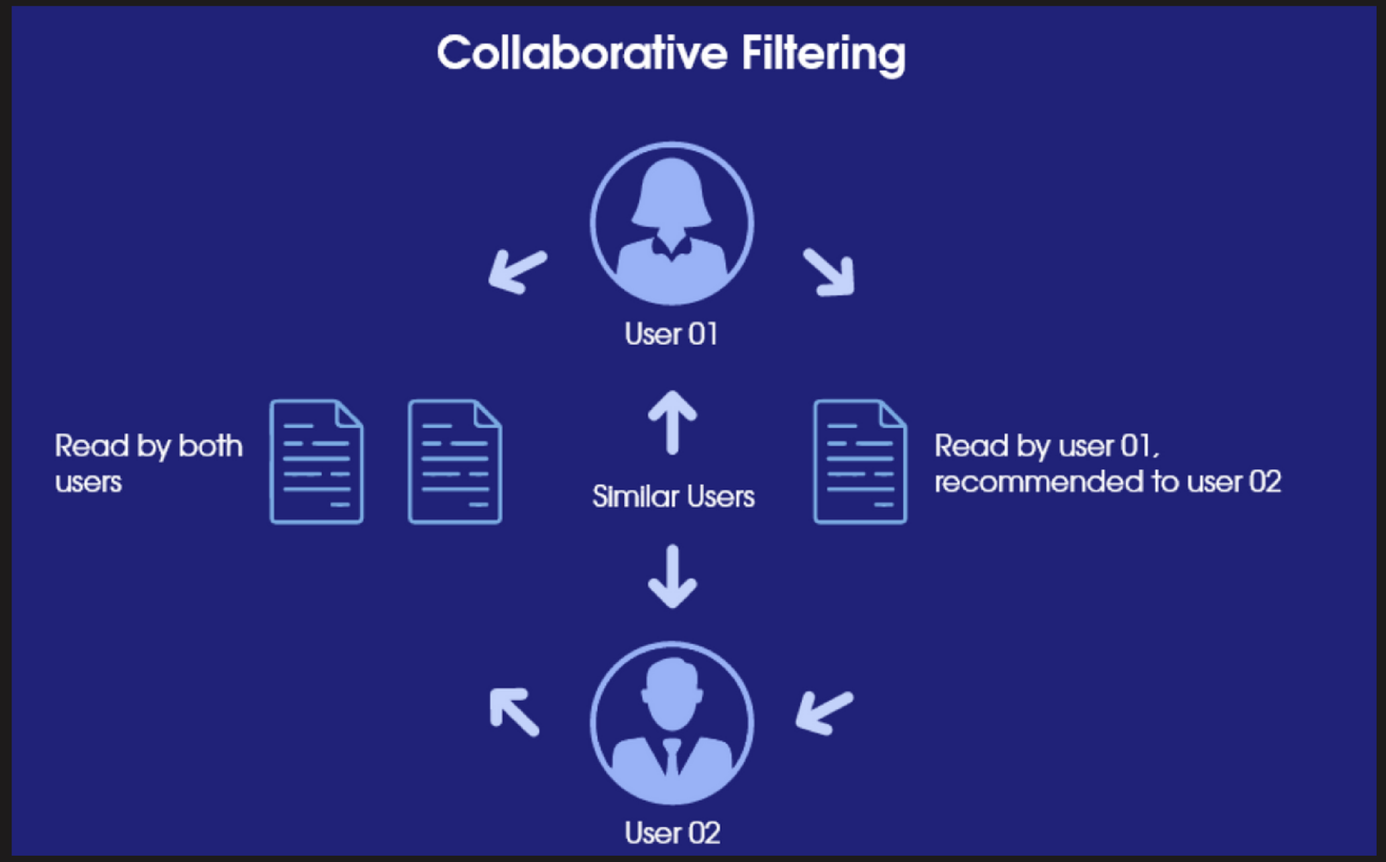
where:

- W = weighted rating
- R = average for the movie as a number from 1 to 10 (mean) = (Rating)
- v = number of votes for the movie = (votes)
- m = minimum votes required to be listed in the Top 250 (currently 25,000)
- C = the mean vote across the whole report (currently 7.0)

Content-based Filtering



original_title	score	similarity	final_score
Toy Story	0.502729	1.000000	0.850819
Toy Story 2	0.460652	0.533157	0.511405
Toy Story 3	0.489255	0.275823	0.339853
Toy Story of Terror!	0.422935	0.299218	0.336333
Small Fry	0.375331	0.271239	0.302467
Hawaiian Vacation	0.389081	0.264853	0.302121
WALL·E	0.508330	0.199791	0.292353
Finding Nemo	0.495991	0.197794	0.287253
Up	0.510838	0.173065	0.274397
A Bug's Life	0.411551	0.204652	0.266722

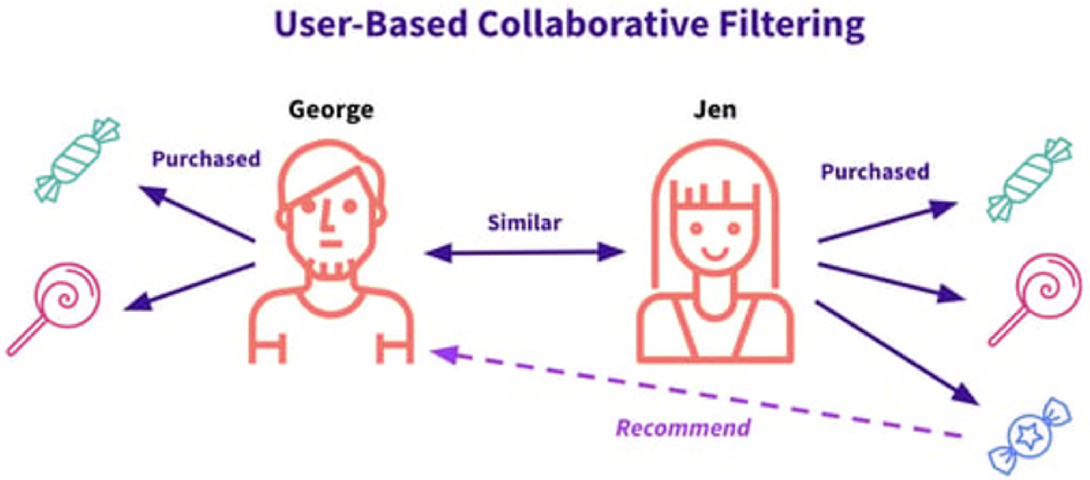


Methodology

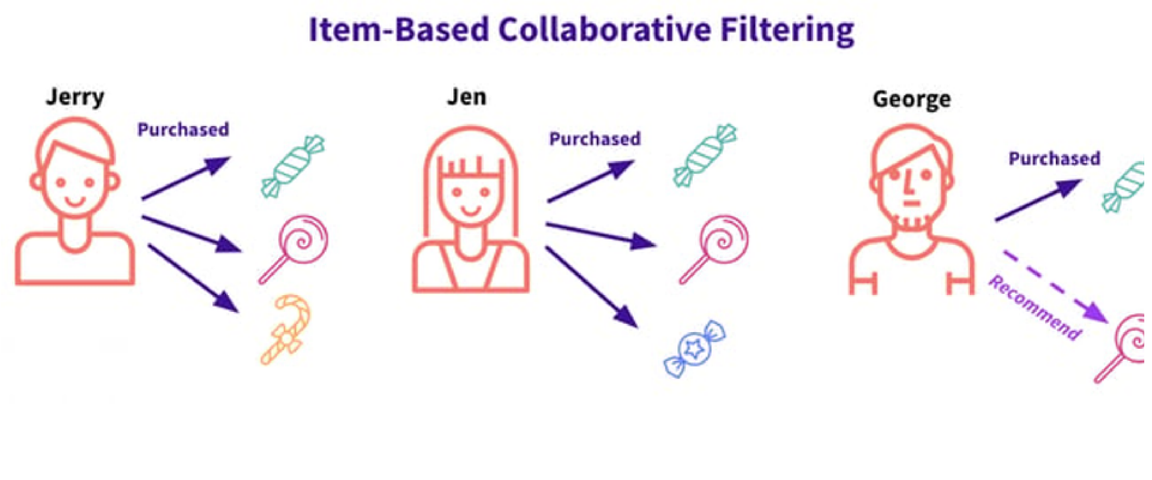
Collaborative

Collaborative Filtering is a technique used by recommendation systems to make predictions or recommendations about which movies a user might like, based on the preferences or behavior of similar users.

User-based vs Item-based CF



a. User-based CF



b. Item-based CF

Matrix Factorization

An approach transforms both movie and user to the same latent factor space.

	 Harry Potter	 The Triplets of Belleville	 Shrek	 The Dark Knight Rises	 Memento
	✔		✔	✔	
		✔			✔
	✔	✔	✔		
				✔	✔

≈

1	0,1
-1	0
0,2	-1
0,1	1

0,9	-1	1	1	-0,9
-0,2	-0,8	-1	0,9	1

0,88	-1,08	0,9	1,09	-0,8
-0,9	1	-1	-1	0,9
0,38	0,6	1,2	-0,7	-1,18
-0,11	-0,9	-0,9	1	0,91

SVD and SVD++

	Score	***Loss function***
SVD	$\hat{r}_{ui} = q_i^T \times p_u$	$J = \sum (r_{ui} - \hat{r}_{ui})$
SVD++	$\hat{r}_{ui} = q_i^T \times (p_u + \frac{1}{\sqrt{N(u)}} \sum_{j \in N(u)} y_j) + \mu + b_i + b_u$	$J = \sum (r_{ui} - \hat{r}_{ui}) + \lambda \left(\left\|\|q_i\right\|\right\|^2 + \left\|\|p_u\right\|\right\|^2 + b_u^2 + b_i^2 + \sum_{j \in N(u)} \left\|\|y_j\right\|\right\|^2 \right)$

- μ: global average rating
- b_u: user bias
- b_i: item bias
- q_i: latent factors of item i
- p_i: latent factor of user u
- $N(u)$: number of seen movies by user u
- y_j: item implicit feedback expressed by j

SVD and SVD++

Base line estimate for Titanic rating by Joe

- Average rating of all movie: 3.5
- Titanic is better than average 0.5
- Joe tend to rate lower than average 0.3

=> Rating: 3.5 + 0.5 -0.3 = 3.7

	Score	***Loss function***
SVD	$\hat{r}_{ui} = q_i^T \times p_u$	$J = \sum (r_{ui} - \hat{r}_{ui})$
SVD++	$\hat{r}_{ui} = q_i^T \times (p_u + \frac{1}{\sqrt{N(u)}} \sum_{j \in N(u)} y_j) + \mu + b_i + b_u$	$J = \sum (r_{ui} - \hat{r}_{ui}) + \lambda \left(\|\|q_i\|\|^2 + \|\|p_u\|\|^2 + b_u^2 + b_i^2 + \sum_{j \in N(u)} \|\|y_j\|\|^2 \right)$

Model can infer user preference from their past behaviour (implicit feedback) like browsing history, purchase history or watched movie.

Regularization to prevent overfiting

Neighborhood based (KNN)

$$\hat{r}_{ui} = \frac{\sum_{v \in N_i^k(u)} sim(u,v) * r_{vi}}{\sum_{v \in N_i^k(u)} sim(u,v)}$$

- r_{ui}: This is the predicted rating of user u for item i.
- $N_i^k(u)$: This is the set of k items rated by user u that are most similar to item i. In other words, these are the k nearest neighbors of item i based on the ratings by user u.
- $sim(u,v)$: This is the similarity between user u and user v.

Mean Squared Differences

$$MSD(u,v) = \frac{1}{|I_{uv}|} * \sum_{i \in I_{uv}} (r_{ui} - r_{vi})^2 \qquad sim(u,v) = \frac{1}{MSD(u,v) + 1}$$

- I_{uv}: This is the number of items rated by both user u and user v.
- r_{ui}: This is the rating of user u for item i.
- r_{vi}: This is the rating of user v for item i.

KNN with mean

$$\hat{r}_{ui} = \mu_u + \frac{\sum_{v \in N_i^k(u)} sim(u,v) * (r_{vi} - u_v)}{\sum_{v \in N_i^k(u)} sim(u,v)}$$

- μ_u: is the mean rating of user u.
- $N_i^k(u)$: is the set of k items most similar to item i that were rated by user u.
- $sim(u,v)$:is the similarity between users u and v.
- r_{vi}: is the rating of item i by user v.
- μ_v: is the mean rating of user v.

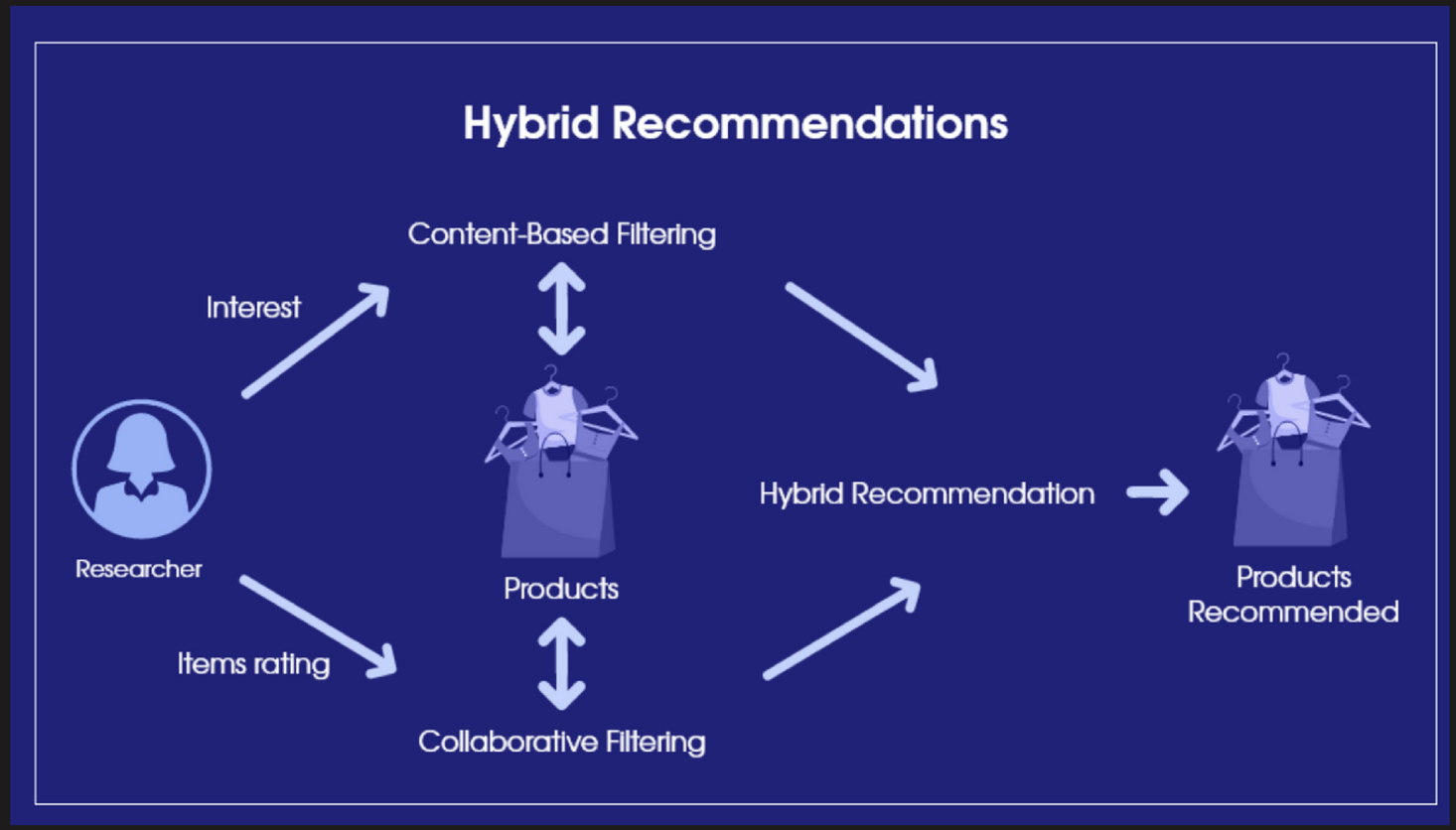
Model Performance

STT	Algorithm	RMSE	MAE
1	Item-based CF	2.5511	2.2585
2	User-based CF	1.7188	1.3484
3	Baseline SVD	0.9041	0.6974
4	**SVD++**	**0.8986**	**0.6924**
5	KNN Basic (user-based)	0.943	0.73
6	KNN Mean (user-based)	0.918	0.705

Methodology

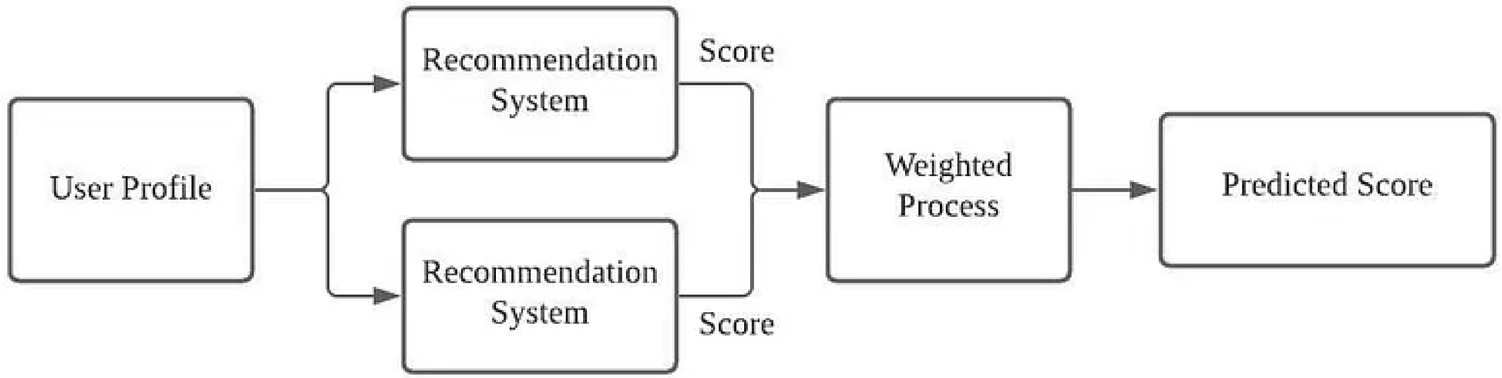
Hybrid Method

- Combination of Collaborative Filtering and Content-Based Filtering
- Model-based Hybrid Methods



Combination of Collaborative Filtering and Content-Based Filtering

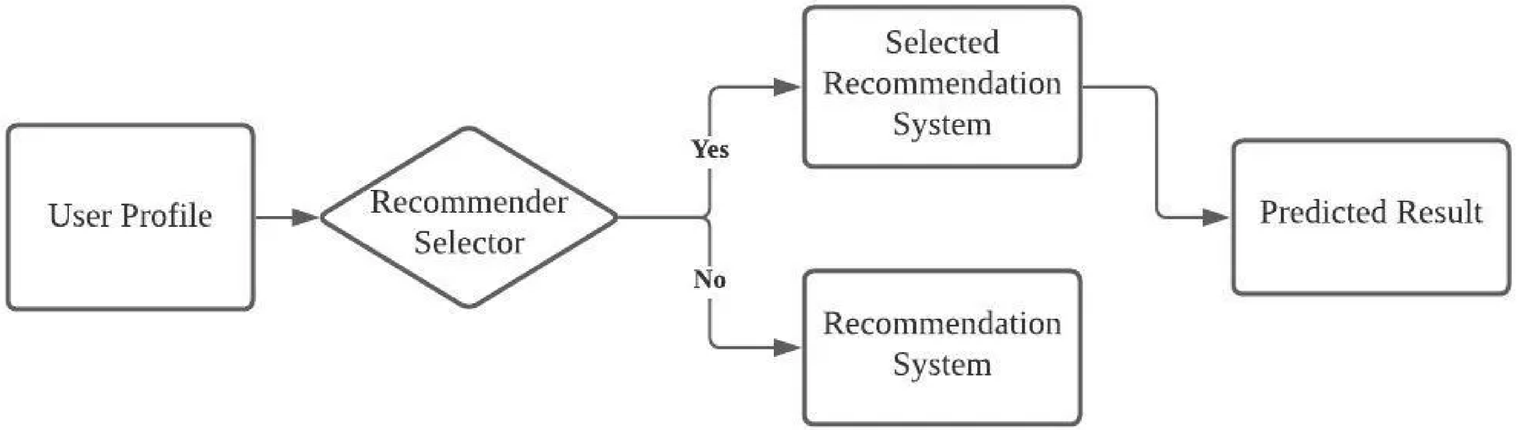
- **Weighted recommendation system**



- The weighted recommendation system will take the outputs from each of the models and combine the result in static weightings.
- For example, we can combine a content-based model and a item-item collaborative filtering model, and each takes a weight of 50% toward the final prediction.

Combination of Collaborative Filtering and Content-Based Filtering

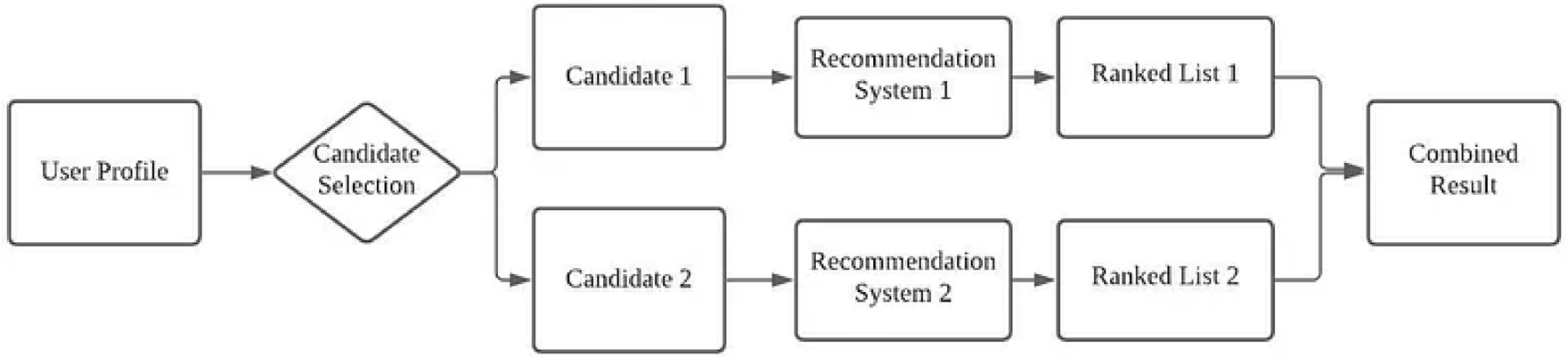
- **Switching Recommendation System**



  - The switching hybrid selects a single recommendation system based on the situation.
  - The switching hybrid approach introduces an additional layer upon the recommendation model, which select the appropriate model to use. The recommender system is sensitive to the strengths and weakness of the constituent recommendation model.

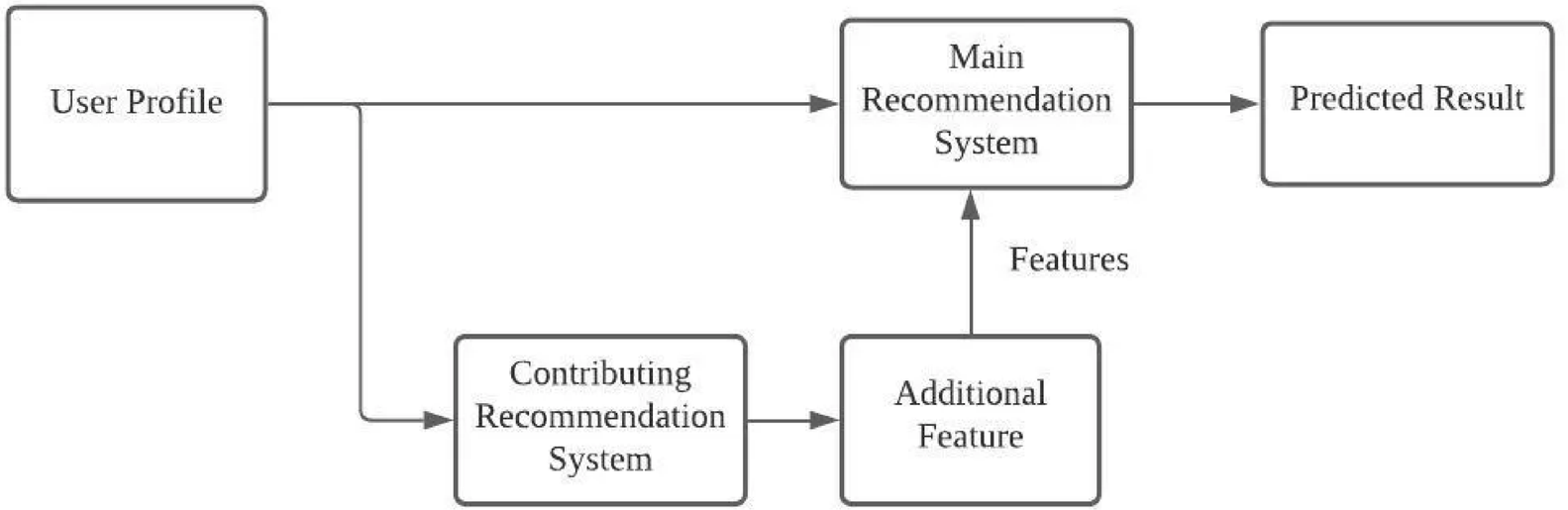
Combination of Collaborative Filtering and Content-Based Filtering

- **Mixed recommendation system**

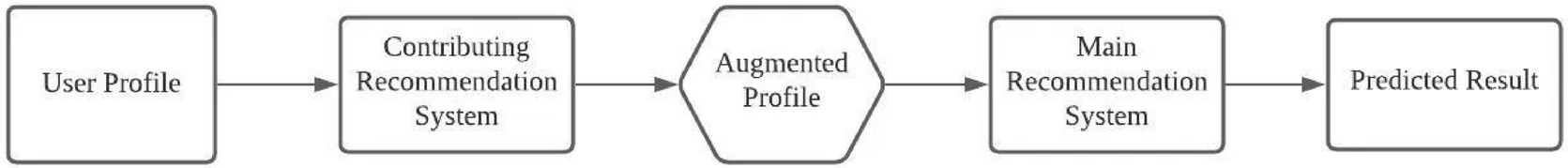


Mixed hybrid approach first takes the user profile and features to generate different set of candidate datasets. The recommendation system inputs different set of candidate to the recommendation model accordingly, and combine the prediction to produce the result recommendation.

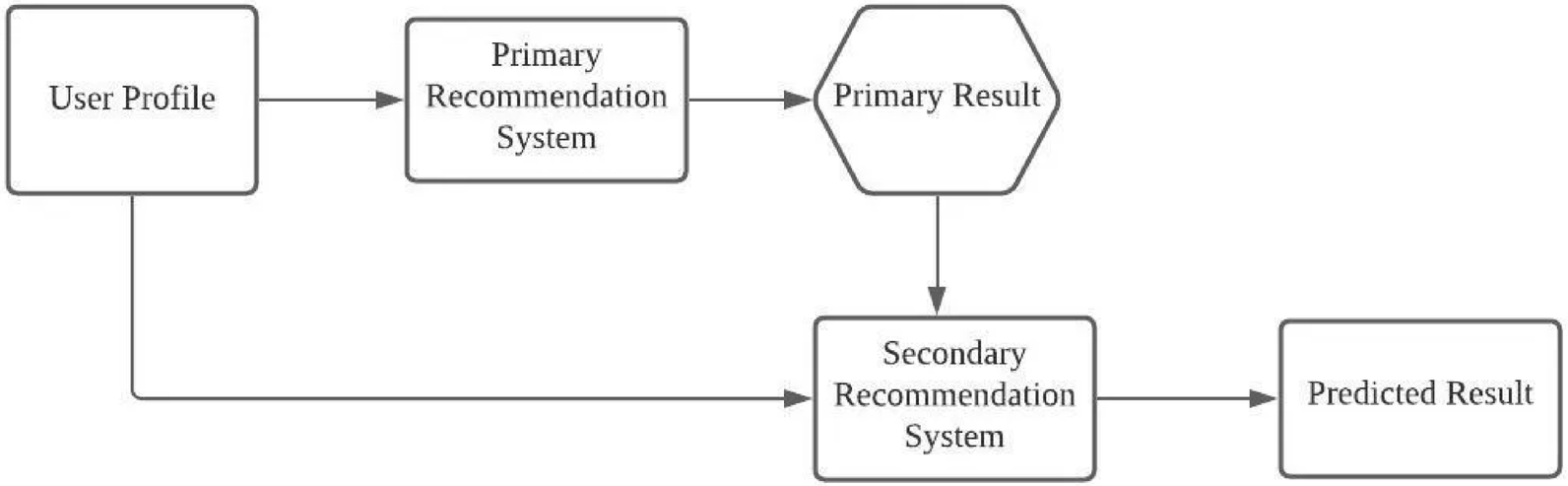
- **Feature Combination**



- **Feature Augmentation**

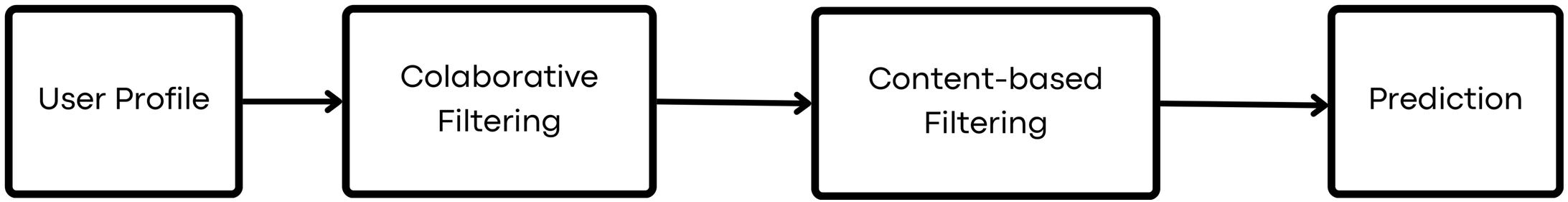


- **Cascade**



Cascade hybrid defines a strict hierarchical structure recommendation system, such that the main recommendation system produce the primary result, and we use the secondary model to resolve some minor issues of the primary result, like breaking tie in the scoring.

- **Our hybrid recommendation system**



	title	est
2514	Toy Story 2	3.224177
7589	Toy Story 3	3.203545
3817	Monsters, Inc.	3.140483
8555	The Lego Movie	3.024008
1875	A Bug's Life	3.020554
7133	A Matter of Loaf and Death	2.985126
4317	The Looney, Looney, Looney Bugs Bunny Movie	2.947804
2743	Creature Comforts	2.881752
7360	Cloudy with a Chance of Meatballs	2.879039
8479	Toy Story of Terror!	2.813607

UserId = 1, Movie = 'Toy Story'

	title	est
8555	The Lego Movie	3.529343
7589	Toy Story 3	3.455460
7360	Cloudy with a Chance of Meatballs	3.371939
3817	Monsters, Inc.	3.284562
6502	Monster House	3.244078
6464	Cars	3.244075
4317	The Looney, Looney, Looney Bugs Bunny Movie	3.198106
7218	Kung Fu Panda: Secrets of the Furious Five	3.159527
1875	A Bug's Life	3.109210
2514	Toy Story 2	3.074789

UserId = 500, Movie = 'Toy Story'

Conclusion

Future work

Utilize data source

Target # 1

Making Recommendation System contextual

Apply deep learning

Target # 2

Handle limitation of Machine Learning methods

Thanks for your listening.

FINAL PROJECT - DATA BUSINESS ANALYSIS

LECTURER

Phd. Tuan Minh Pham

PRESENTED BY

Group 2