

CS246 Exam 2025 — Question 5: Recommender Systems (16 points)

Context

Song	Duration	Genre	Artist	Total Plays
S1	240 sec	Rock	Band X	50,000
S2	320 sec	Rock	Band X	120,000
S3	180 sec	Jazz	Artist Y	75,000
S4	210 sec	Pop	Artist Z	95,000
S5	400 sec	Classical	Orchestra W	30,000
S6	420 sec	Classical	Orchestra W	10,000

Part 1 (3 points) — Which Recommender Could Suggest S2 to U1?

U1 listens to **long classical music** (classical, >350 sec). S2 is Rock, 320 sec — a mismatch on both features.

Answer: (a) User-user CF and (b) Item-item CF

- **(c) Content-based — No.** It builds a profile from item features. U1's profile emphasizes Classical + long duration. S2 matches neither, so it would never be recommended.
- **(a) User-user CF — Yes.** It finds users with similar *rating patterns* and recommends what they liked. It's blind to content features, so S2 can surface if similar users enjoy it.
- **(b) Item-item CF — Yes.** It finds items with similar *rating vectors*. If S2 is rated similarly to items U1 liked, it gets recommended — regardless of genre or duration.

Key insight: Content-based is constrained by features; collaborative filtering operates purely on rating patterns and can recommend across genres.

Part 2 (3 points) — Which System Best Explains Recommendations?

Answer: (c) Content-based recommender system

A content-based system can explain: *"We recommended this because you like Jazz, and this is a Jazz song with similar duration."* It points to specific features that drove the recommendation.

Collaborative filtering can only say *"users similar to you liked this"* or *"this is rated similarly to songs you enjoyed"* — offering no insight into *what* about the song makes it a good fit.

Part 3 (7 points) — Computing Predicted Ratings

Ratings matrix (? = unknown):

Song	U1	U2	U3	U4
S1	1	0	?	1
S2	1	1	0	1
S3	1	1	1	0
S4	0	1	1	1
S5	1	0	1	1

Part 3a — Item-Item Collaborative Filtering

$$r_{U3,S1} = \frac{\sum_j s_{S1,j} \cdot r_{U3,j}}{\sum_j s_{S1,j}}$$

Using all 4 songs U3 has rated as neighbors:

Song j	U3's rating	Sim to S1	s × r
S2	0	0.75	0.00
S3	1	0.60	0.60
S4	1	0.50	0.50
S5	1	0.65	0.65

$$r_{U3,S1}^{\text{item-item}} = \frac{0.00 + 0.60 + 0.50 + 0.65}{0.75 + 0.60 + 0.50 + 0.65} = \frac{1.75}{2.50} = 0.70$$

Part 3b — User-User Collaborative Filtering

$$r_{U3,S1} = \frac{\sum_y s_{U3,y} \cdot r_{y,S1}}{\sum_y s_{U3,y}}$$

Using all 3 other users who rated S1:

User y	Rating for S1	Sim to U3	s × r
U1	1	0.50	0.50
U2	0	0.80	0.00
U4	1	0.70	0.70

$$r_{U3,S1}^{\text{user-user}} = \frac{0.50 + 0.00 + 0.70}{0.50 + 0.80 + 0.70} = \frac{1.20}{2.00} = 0.60$$

Note: U2 is the most similar user to U3 (0.80) but disliked S1, dragging the prediction down.

Part 3c — Do We Recommend S1?

$$\text{Average} = \frac{0.70 + 0.60}{2} = 0.65 \geq 0.5$$

(A) Yes — we recommend S1 to U3.

Part 4 (3 points) — Genre-Diverse Recommendations

Task: Find top 3 songs incorporating different genres using item-item CF.

Answer: Use a **greedy re-ranking strategy with genre constraints**:

1. Run item-item CF to produce a large ranked candidate list (e.g., top 20 by predicted rating).
2. Walk through the list from highest to lowest rating. For each song, add it to the final list only if its genre hasn't been included yet. Stop once 3 songs are selected.

This yields the best-rated song from each genre — e.g., the top Rock song, top Jazz song, and top Pop song — combining relevance with variety.