1. What is the difference between user-based and item-based collaborative filtering?

Aspect	User-Based Collaborative Filtering	Item-Based Collaborative Filtering
Core Idea	Find users similar to the target user	Find items similar to those the user has already liked
Similarity Computed Between	Users	Items
Recommendation Based On	What similar users liked	What similar items the user has liked
Scalability	Slower and less scalable (more users = more comparisons)	Faster and more scalable (items change less frequently)
Stability	Can vary as user behavior changes frequently	More stable as item properties are fixed
Cold Start Problem	Struggles with new users	Struggles with new items
Example	"Users like you watched X"	"Because you watched X, you might also like Y"

2. What is Collaborative Filtering and How Does It Work?

Collaborative Filtering is a recommendation technique that suggests items to users based on past interactions or preferences of similar users or items — without using content metadata like genre or type. The key Core Principle is "Users who agreed in the past tend to agree again in the future."

Types of Collaborative Filtering:

1. Memory-Based Methods
    - Use similarity scores (cosine, Pearson) between users/items.
    - Examples:
        - User-User CF: Find users similar to the target and recommend what they liked.
        - Item-Item CF: Recommend items similar to what the user liked.
2. Model-Based Methods
    - Use machine learning models (e.g., matrix factorization, SVD, deep learning).
    - Learn latent factors representing users and items.

How It Works:

- Construct a user-item matrix where rows = users, columns = items, and values = ratings (explicit or implicit).
- Find similar users/items using similarity metrics.
- Predict missing ratings or rank items.
- Recommend top-N items with highest predicted score.

Strengths:

- Works well with implicit data (clicks, views).
- Doesn't require metadata.
- Captures complex user-item interaction patterns.

Limitations:

- Cold Start: Doesn't work well for new users/items with no data.
- Sparsity: Many users rate few items → sparse matrix.
- Scalability: Expensive on large datasets (for memory-based methods).