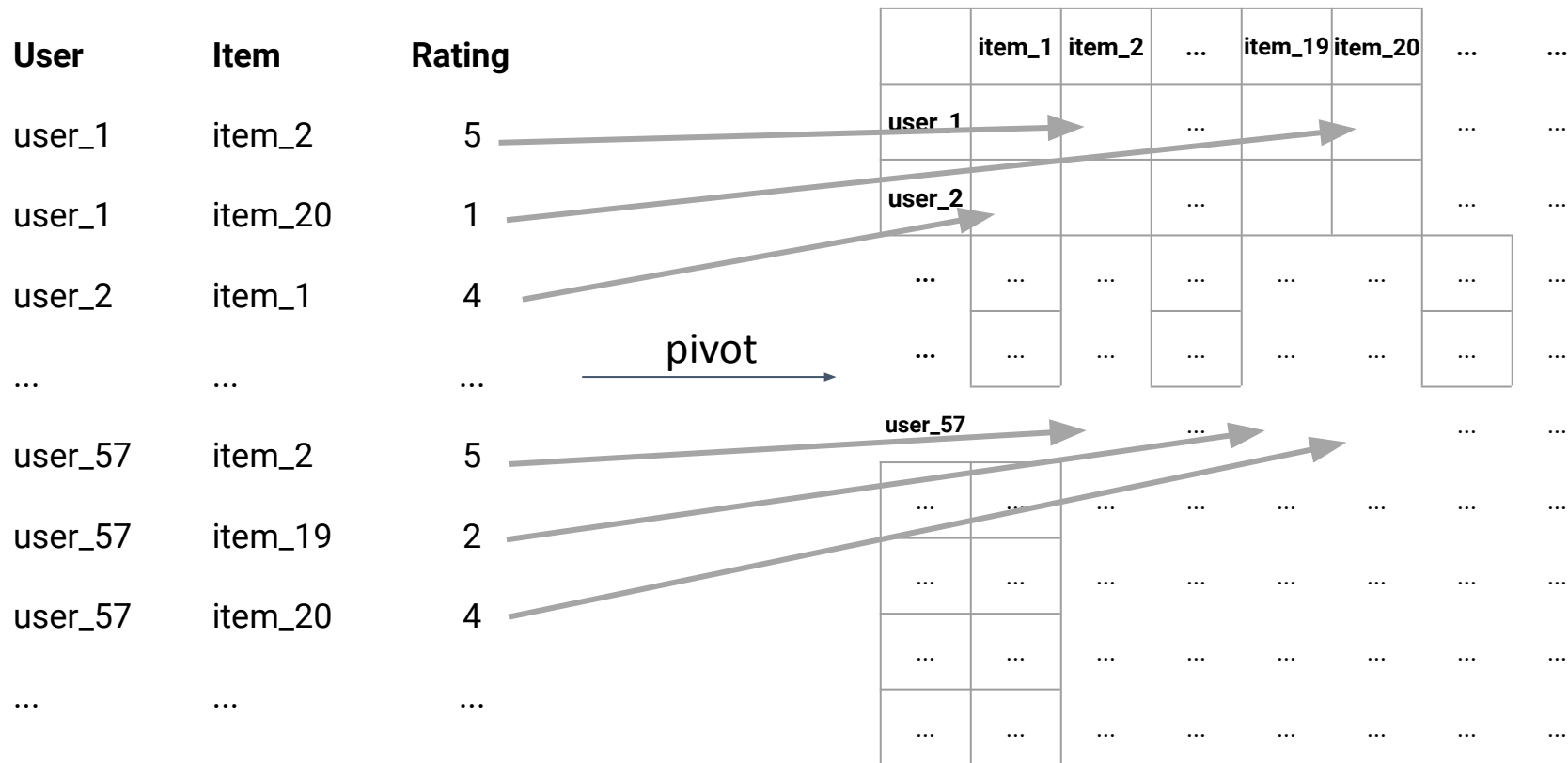


# Basics of User Based Collaborative Filtering

# List of ratings → user-item matrix



## List of ratings $\rightarrow$ user-item matrix

The diagram illustrates the pivot operation in matrix factorization. It shows a user-item rating matrix being decomposed into two matrices,  $U$  and  $V$ , with a pivot value.

**Rating Matrix (Left):**

User	Item	Rating
user_1	item_2	5
user_1	item_20	1
user_2	item_1	4
...	...	...
user_57	item_2	5
user_57	item_19	2
user_57	item_20	4
...	...	...

**Pivot Operation:**

A blue arrow labeled "pivot" points from the rating matrix to the factor matrices.

**Factor Matrix  $U$  (Left):**

User	item_1	item_2	...	item_19	item_20	...	...
user_1	-	5	...	-	1	...	...
user_2	4	-	...	-	-	...	...
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...
user_57	-	5	...	2	4	...	...
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...

**Factor Matrix  $V$  (Right):**

Item	user_1	user_2	...	user_57	...
item_1	...	4	...	...	...
item_2	5	-	...	5	...
...	...	...	...	...	...
item_19	-	-	...	2	...
item_20	1	-	...	4	...
...	...	...	...	...	...
...	...	...	...	...	...

Arrows indicate the mapping of user and item ratings to the corresponding rows and columns in the factor matrices.

# User-based nearest-neighbor collaborative filtering

## The basic technique

Given an "active user" (Alice) and an item  $i$  not yet seen by Alice

- Find a set of users (peers/nearest neighbors) who liked the same items as Alice in the past **and** who have rated item  $i$
- Use, e.g. the average of their ratings to predict, if Alice will like item  $i$
- Do this for all items Alice has not seen and recommend the best-rated

## Basic assumption and idea

- If users had similar tastes in the past they will have similar tastes in the future

# User-based nearest-neighbor collaborative filtering

- Some questions
  - How do we measure similarity?
  - How many neighbors should we consider?
  - How do we generate a prediction from the neighbors' ratings?

