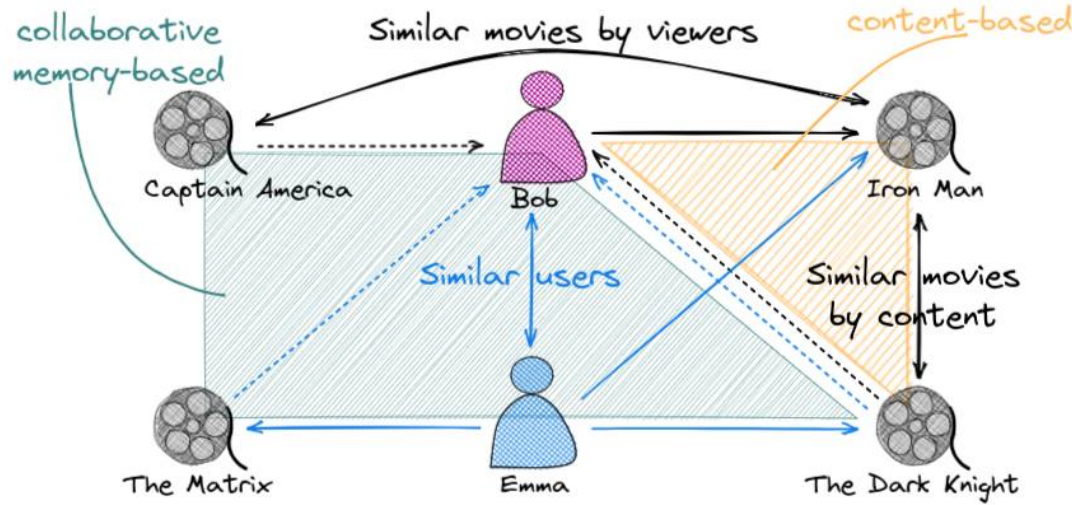


# Recommendation Systems

## Part I

Week 25



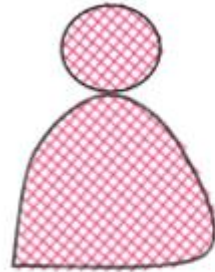
Instructor: Dr. Ivan Reznikov

# Plan

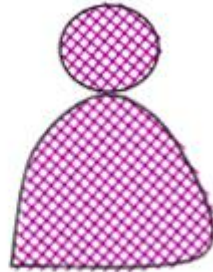
- Statistical approach
- Collaborative-based recommendation systems
- Content-based recommendation systems
- Model-based recommendation systems
- Hybrid recommendation systems
- Graph recommendation systems

# Data

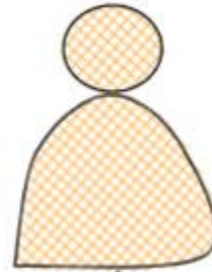
## Users



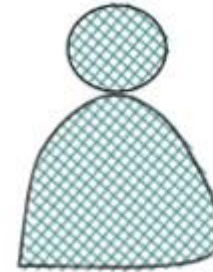
Anna



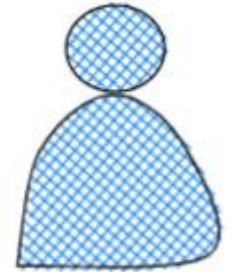
Bob



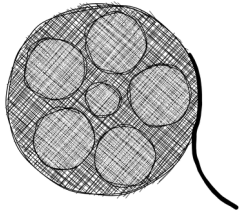
Chloe



David



Emma



## Movies

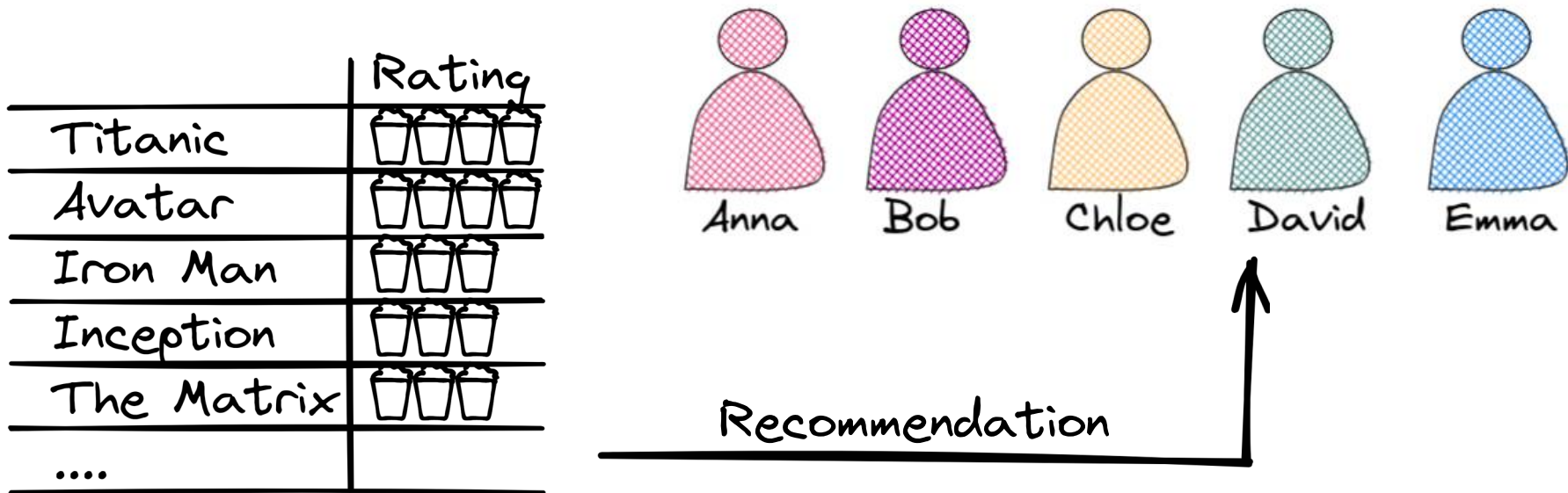
Armageddon  
Avatar  
Black Panther  
Deadpool  
Django Unchained  
Gravity  
Inception

Iron Man  
Justice League  
Logan  
Pretty Woman  
Suicide Squad  
The Avengers  
The Dark Knight

The Great Gatsby  
The Martian  
The Matrix  
The Notebook  
The Wolf of Wall Street  
Thor: Ragnarok  
Titanic  
2001: A Space Odyssey<sup>3</sup>

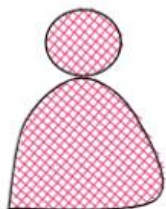
# Recommendation system v 0.1

Recommend most popular movie



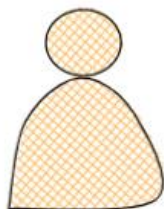
If we suggest a movie based on popularity or statistics, one may call this method **statistic-based**. We can filter movies' ratings by a category, say, age, and recommend popular among teens, for example.

# Users-movies



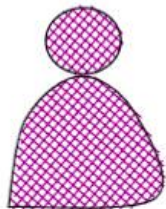
Anna

Inception  
Django Unchained  
The Great Gatsby  
The Wolf of Wall Street  
Titanic



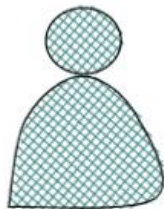
Chloe

Titanic  
Pretty Woman  
Avatar  
Armageddon  
The Great Gatsby  
The Notebook



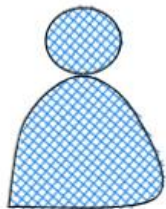
Bob

The Avengers  
Black Panther  
Iron Man  
Thor: Ragnarok  
Deadpool  
Logan



David

2001: A Space Odyssey  
Armageddon  
Avatar  
Gravity  
The Martian  
Thor: Ragnarok

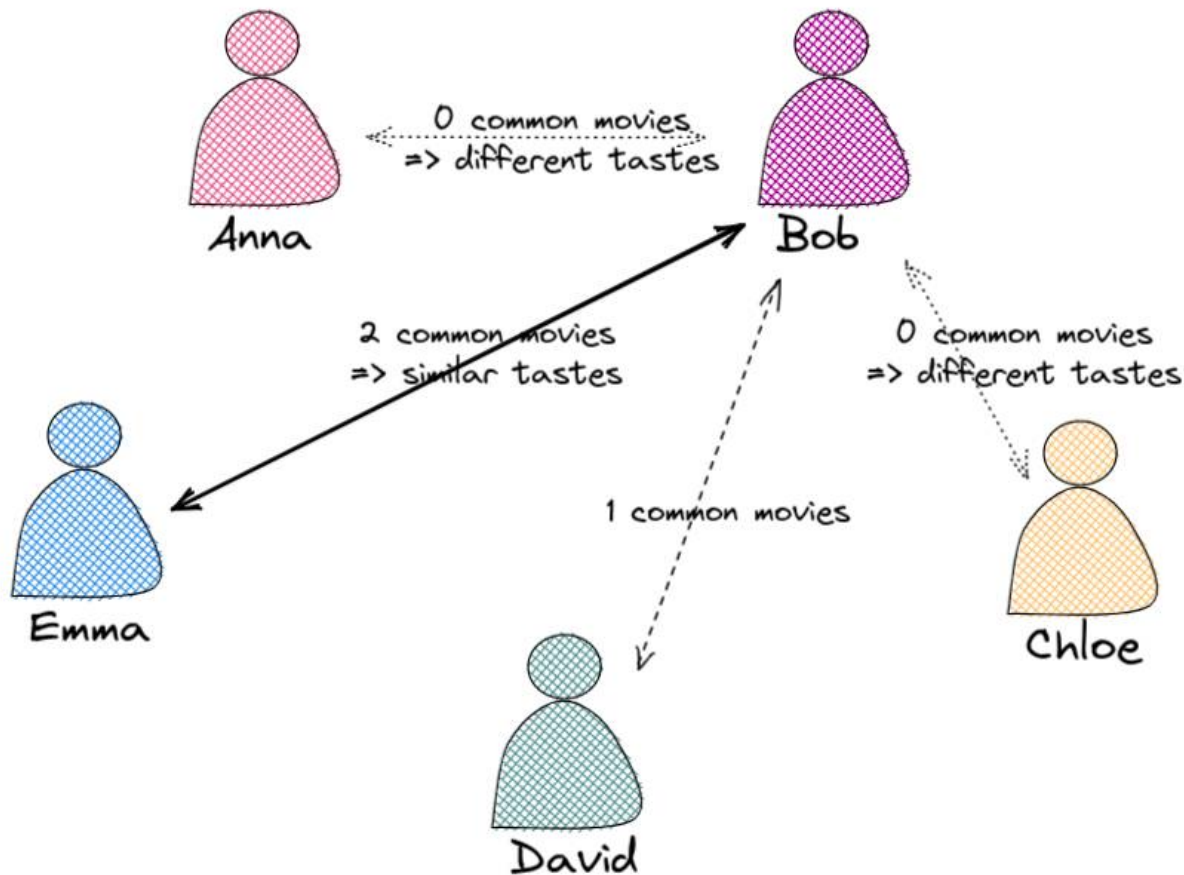


Emma

The Dark Knight  
The Avengers  
Justice League  
Suicide Squad  
The Matrix  
Deadpool

Users  $\leftrightarrow$  movies interactions are most important for building a recommendation system.

# Relationships

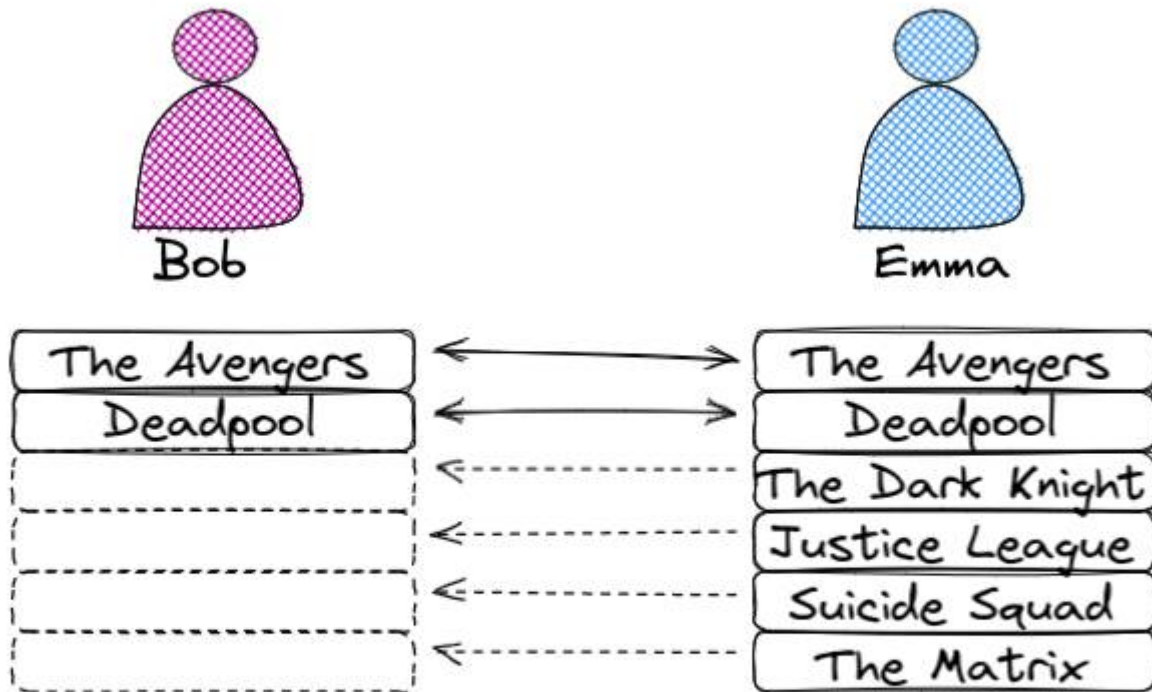


Distance matrix  
of common movies

	A	B	C	D	E
Anna	<del>1</del>	0	1	0	0
Bob		<del>1</del>	0	1	2
Chloe			<del>1</del>	2	0
David				<del>1</del>	0
Emma					<del>1</del>



# Recommendation system v 1.0

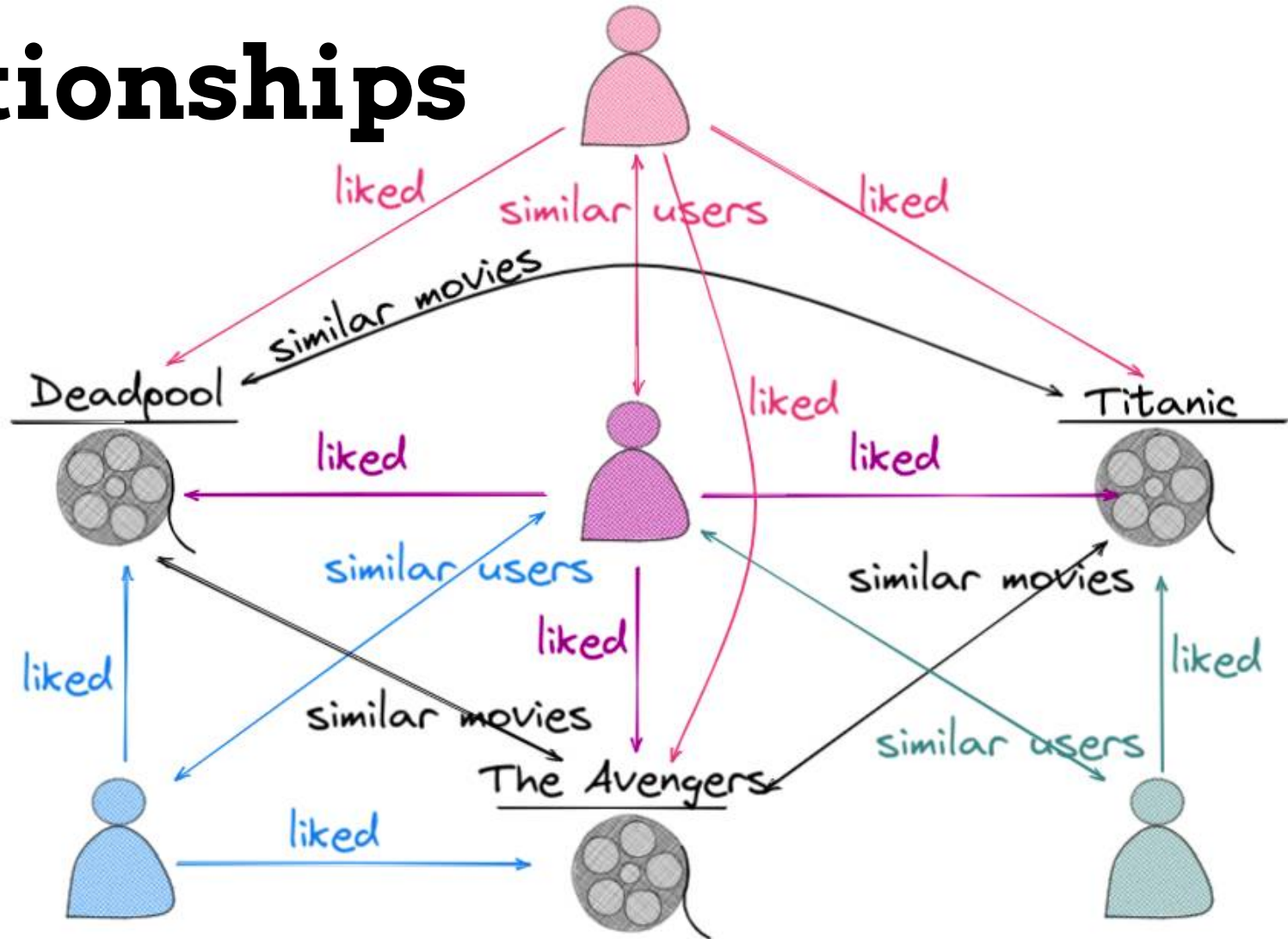


Bob is recommended to watch "The Dark Night", "Justice League", "Suicide Squad" and "The Matrix"

Distance matrix  
of common movies

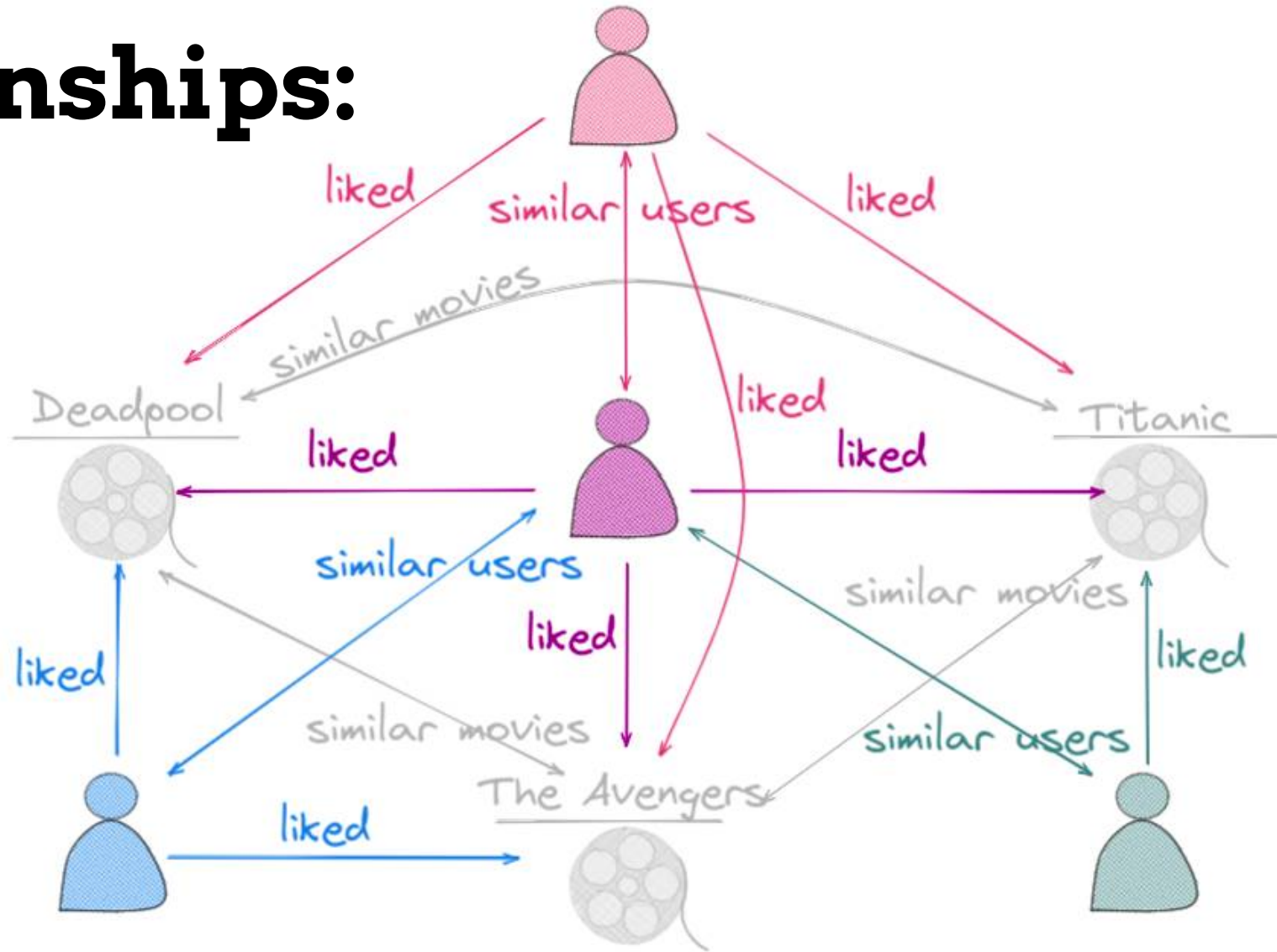
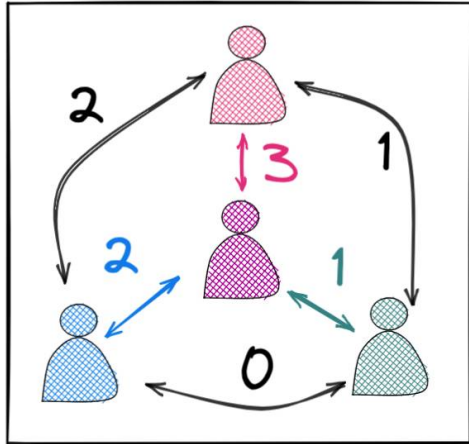
	A	B	C	D	E
Anna	<del>1</del>	0	1	0	0
Bob		<del>1</del>	0	1	2
Chloe			<del>1</del>	2	0
David				<del>1</del>	0
Emma					<del>1</del>

# Relationships

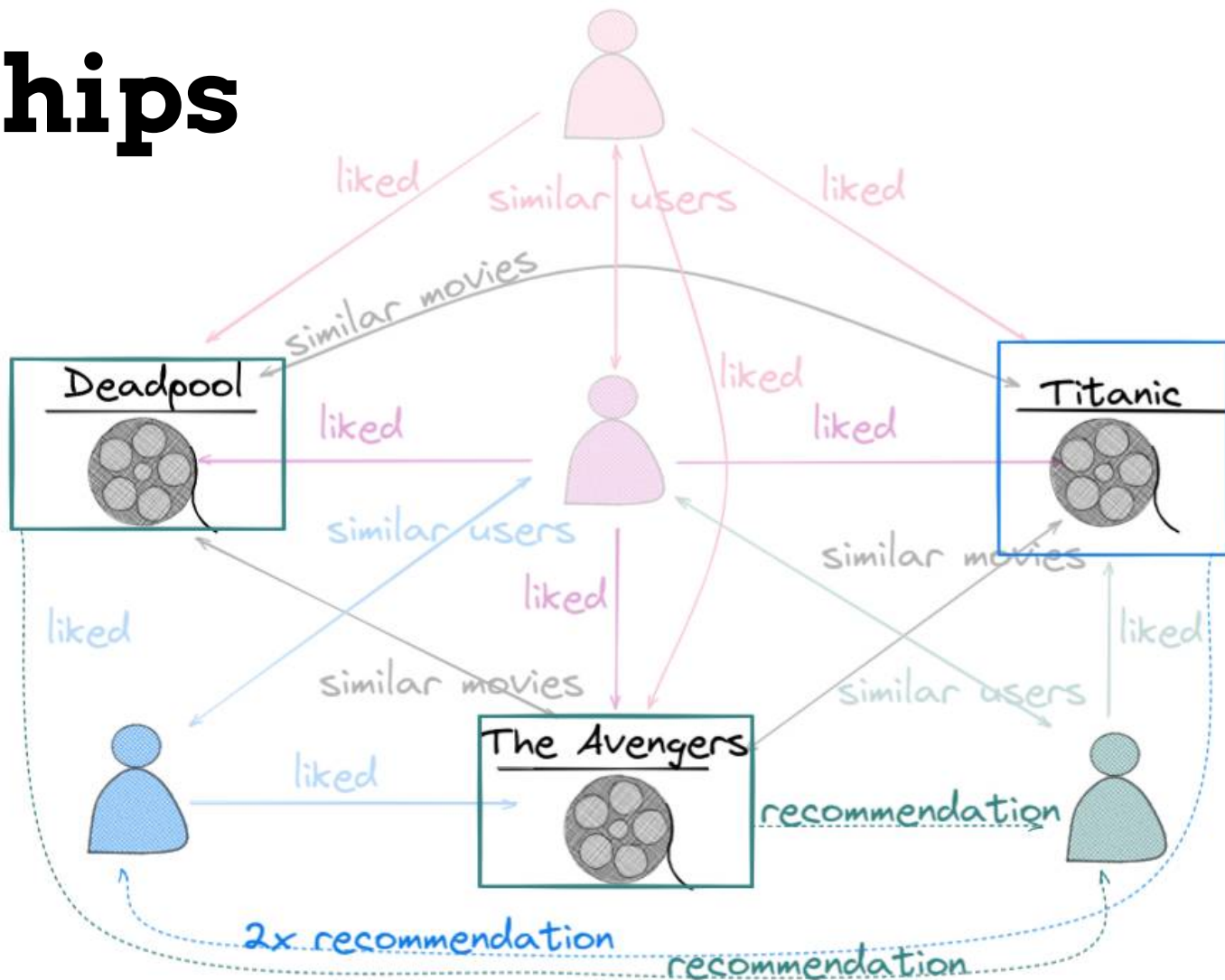




# Relationships: *users*



# Relationships *users*

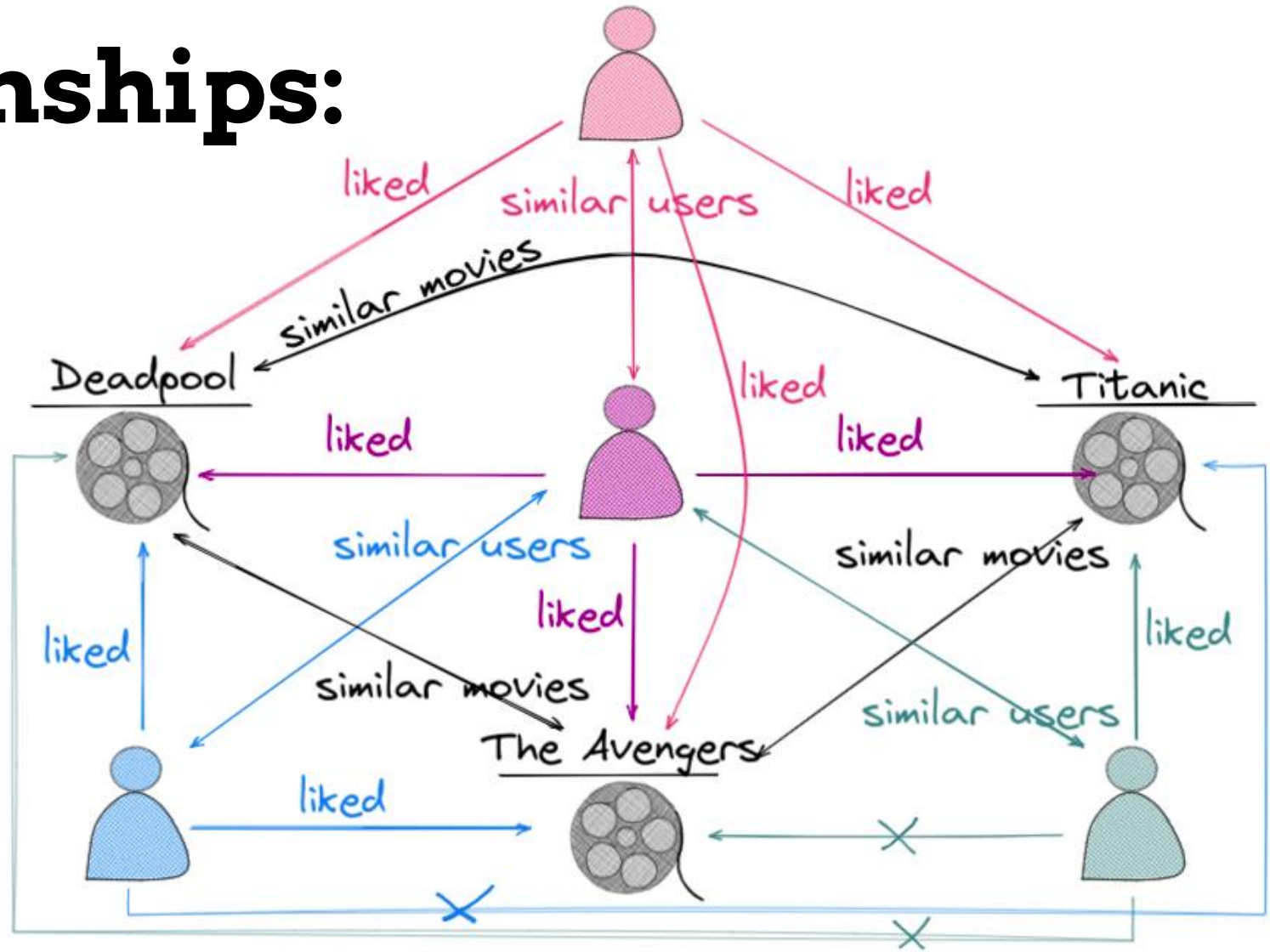


We make the following recommendations based on "users" similarity:

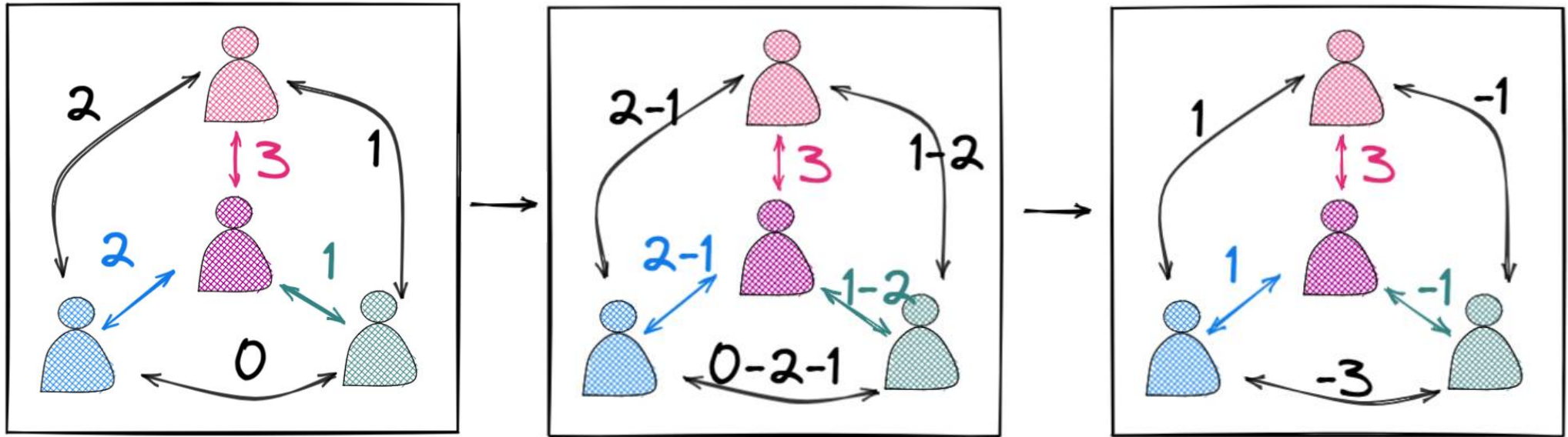
- Titanic (2x)
- Deadpool (1x)
- The avengers (1x)

# Relationships: *users*

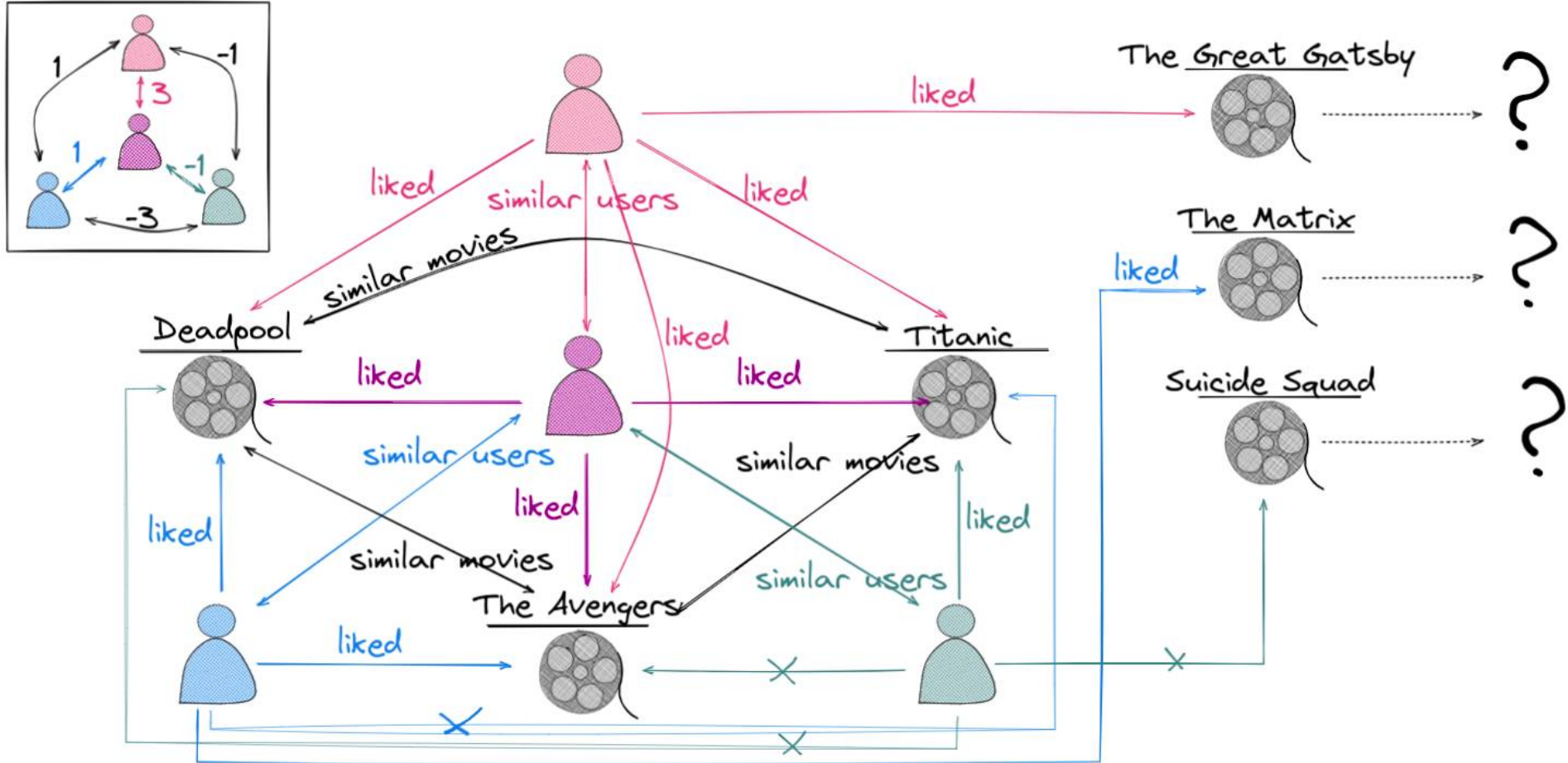
Let's imagine none of our predictions were made correctly.  
How will this change our distance matrix?



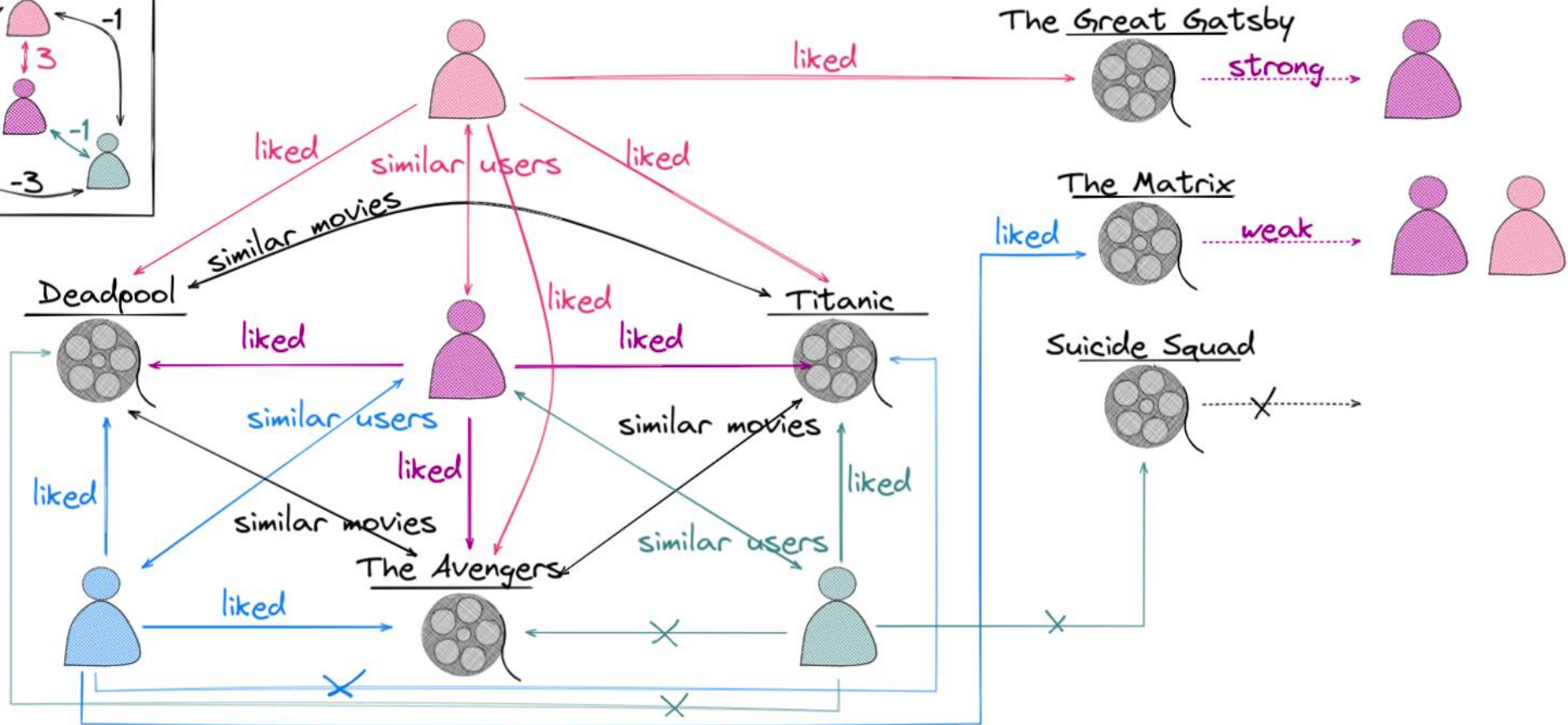
# Relationships: *users*



# Recommendation system v 1.0

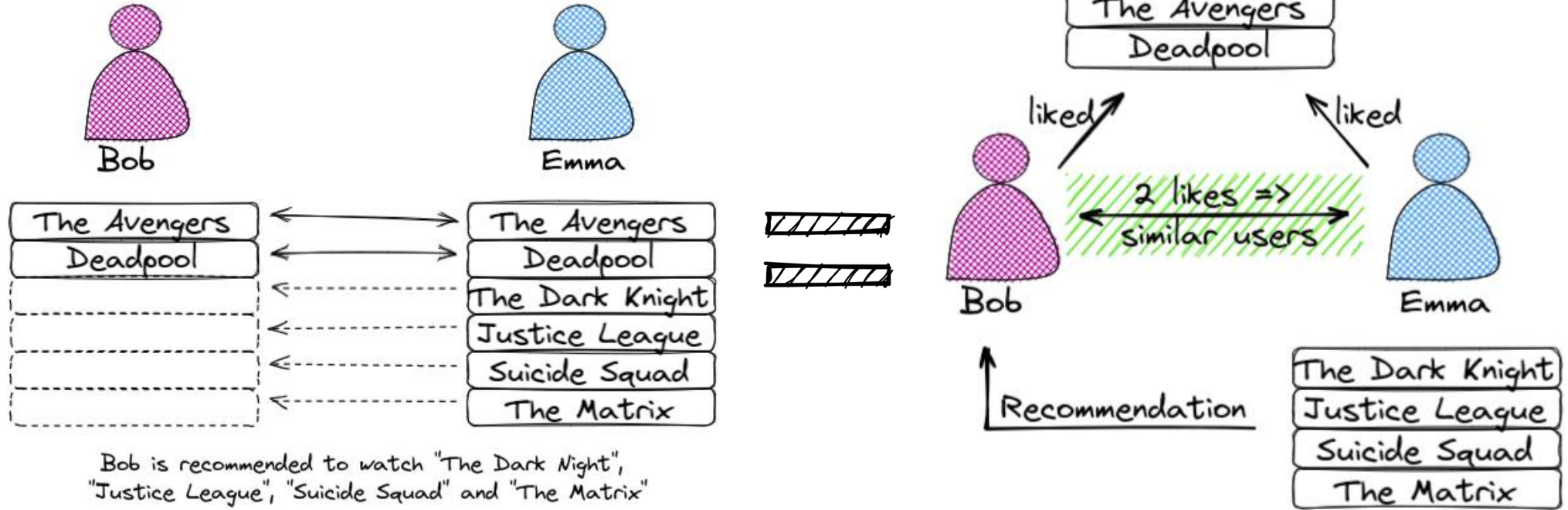








# Collaborative



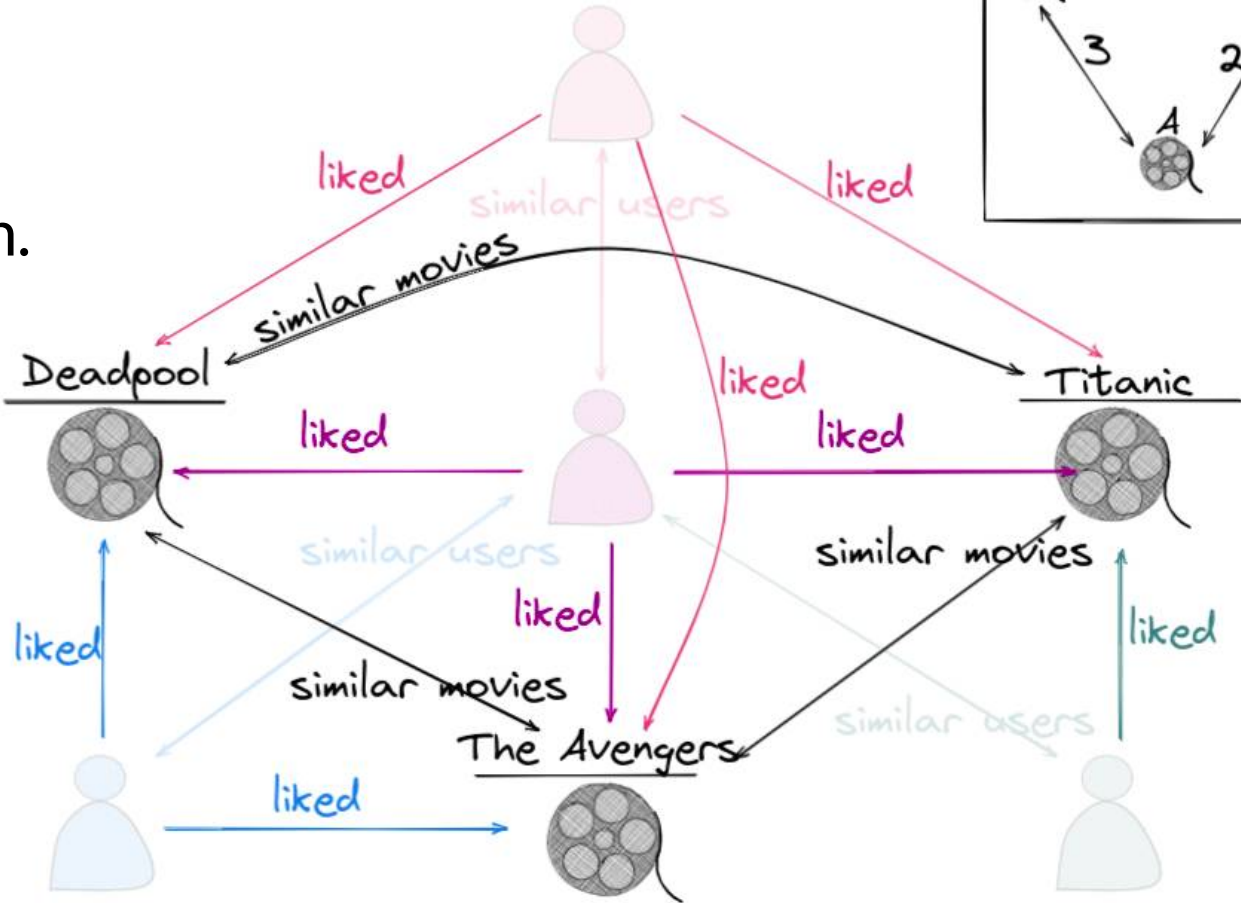
If users similarity is defined through other types (items, movies, etc) then we're dealing with **collaborative recommendation systems**.

# Collaborative: *movies*

Movies' similarity can also be defined through collaboration.

For example theaters can target new film ads more effectively.

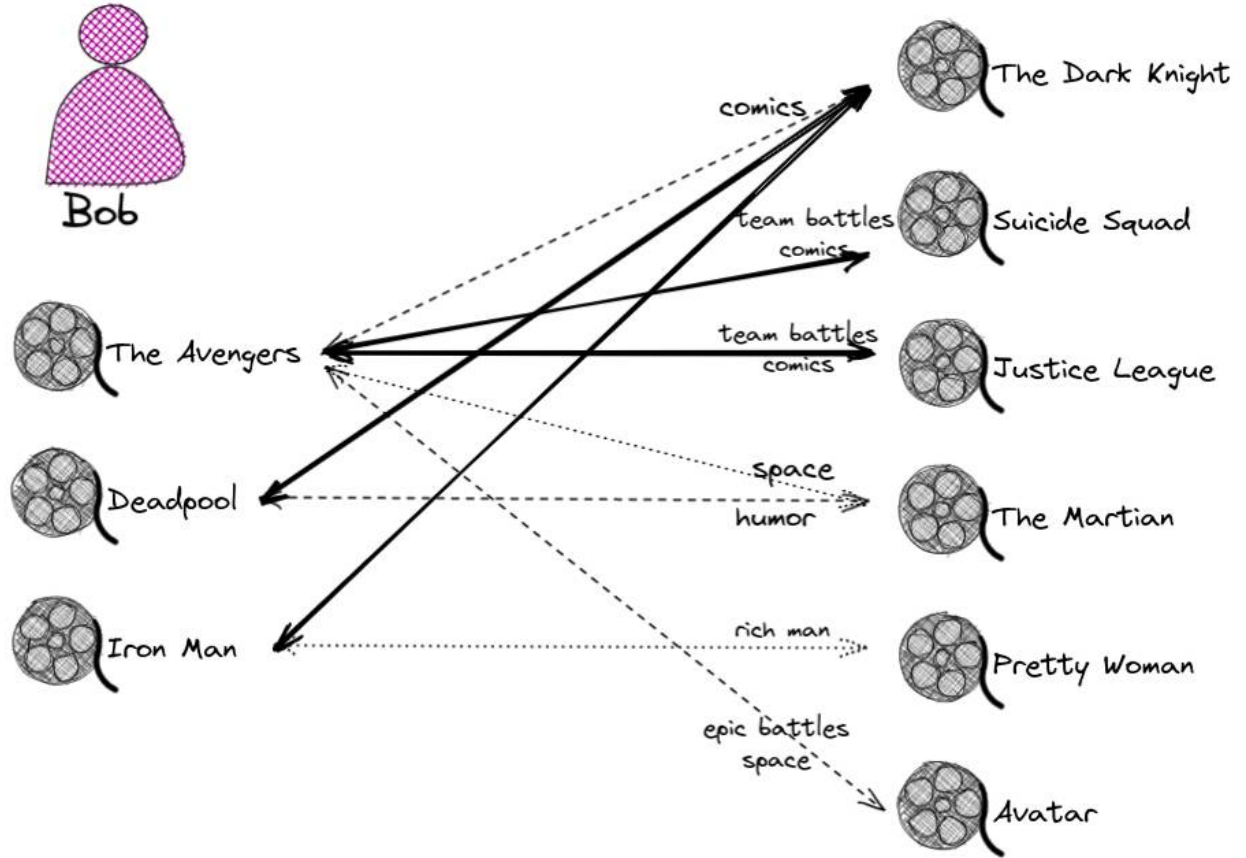
Building similarity spaces is key.



# Content

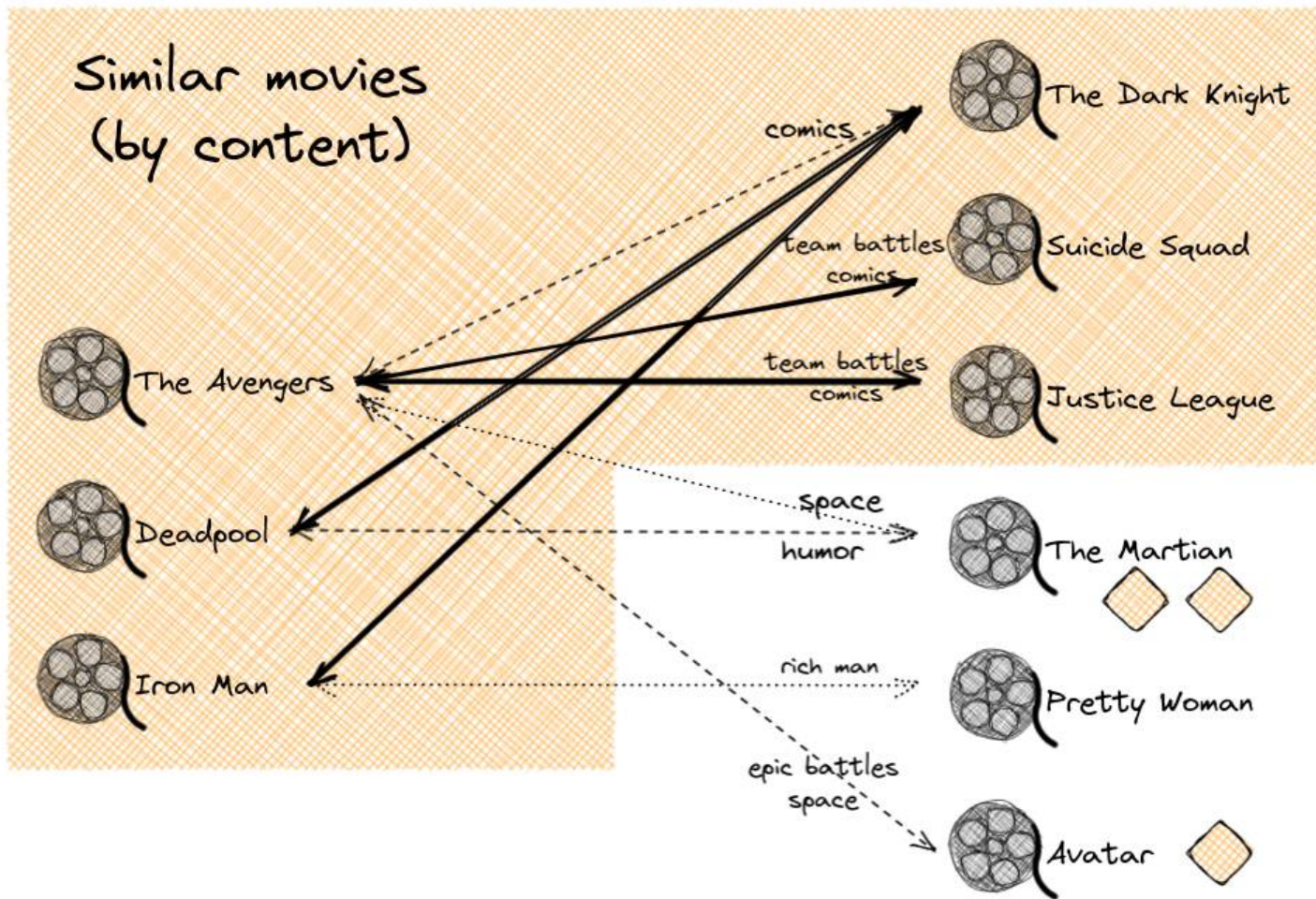
Similarities can also be defined without using other types.

One can define how similar are movies based on their content.



# Content

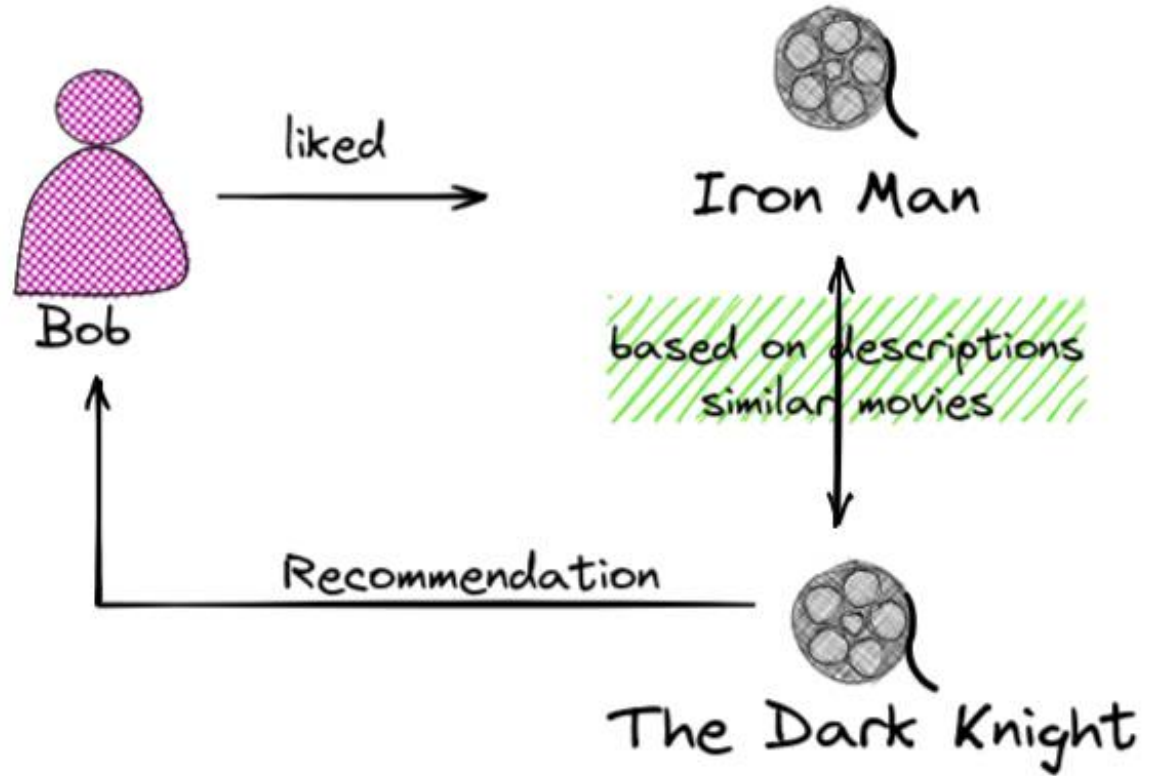
On practice movies descriptions may be used to build the similarity matrix.





# Content

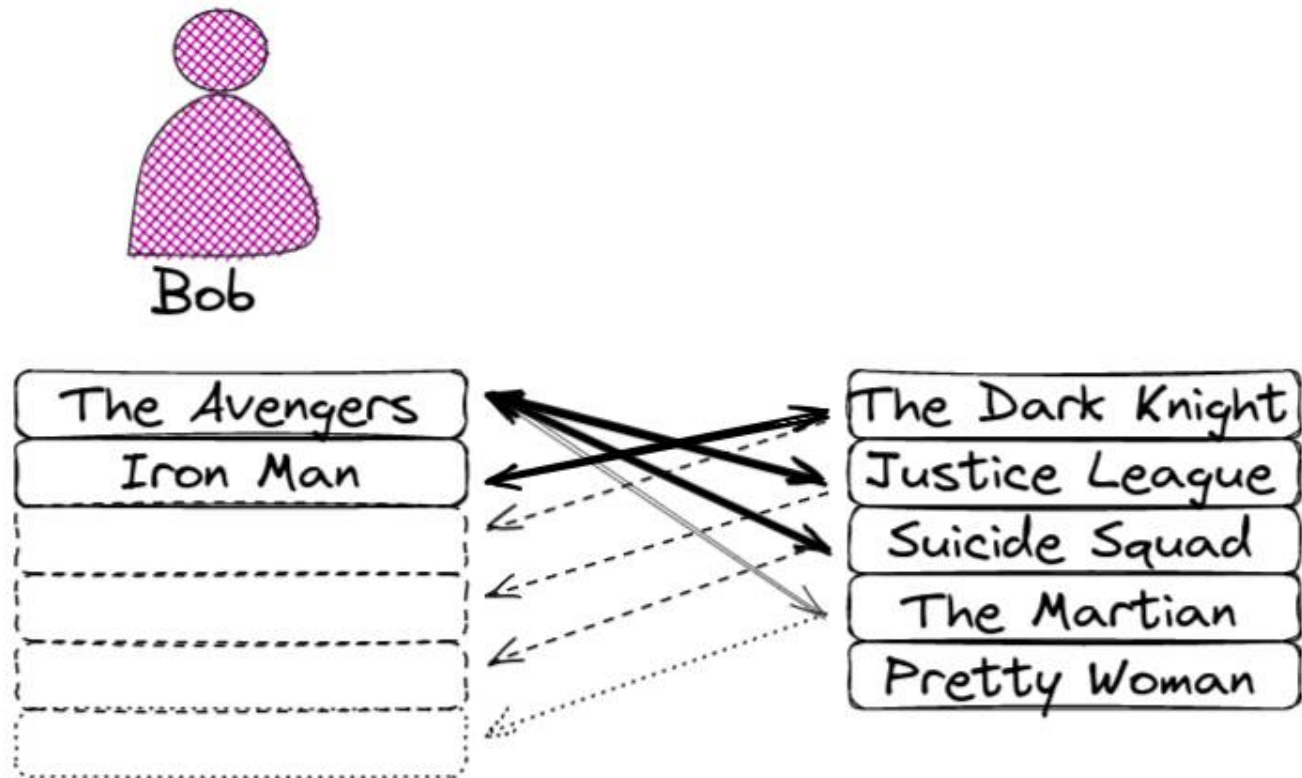
Such recommendation systems are called **content-based**.



# Content

Can you name cases where content-based are preferred to collaborative?

What about vice versa?



Bob is recommended to watch "The Dark Night",  
"Justice League", "Suicide Squad".  
"The Martian" should also be a good choice



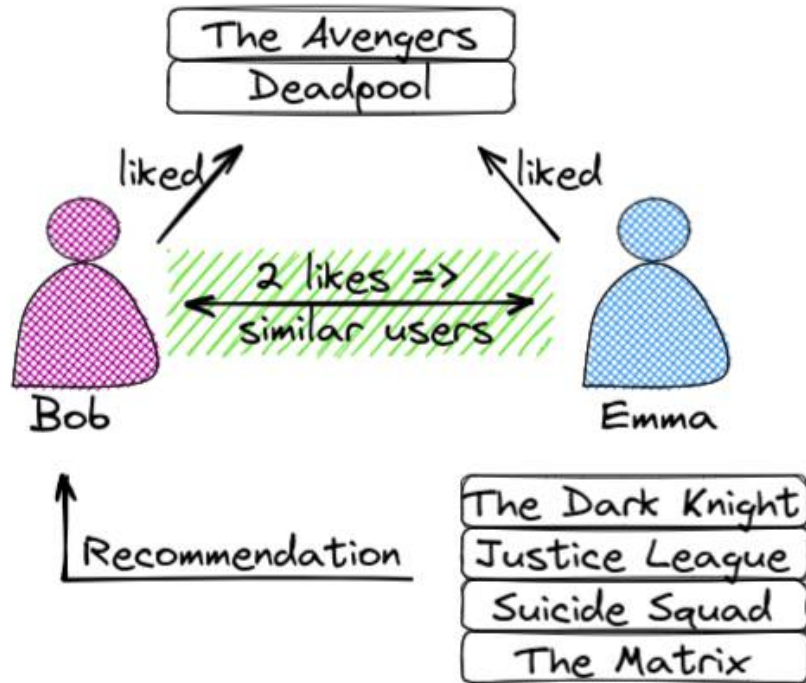
# Model

	gender	age	....	Movie1	Movie2	Movie1143
Anna	f	32	...			
Bob	m	24	...			
Chloe	f	40	...			
David	m	22	...			
Emma	f	25	...			
features				outputs		

One can also make a feature table with users parameters. This data might be used for creating embeddings (numeric characteristic) of users and build a ML model above it, to predict scores. This type of recommendations are called **model-based**.

# Collaborative

## Memory-based



## Model-based

	gender	age	....	Movie1	Movie2	Movie1143
Anna	f	32	...			
Bob	m	24	...			
Chloe	f	40	...			
David	m	22	...			
Emma	f	25	...			

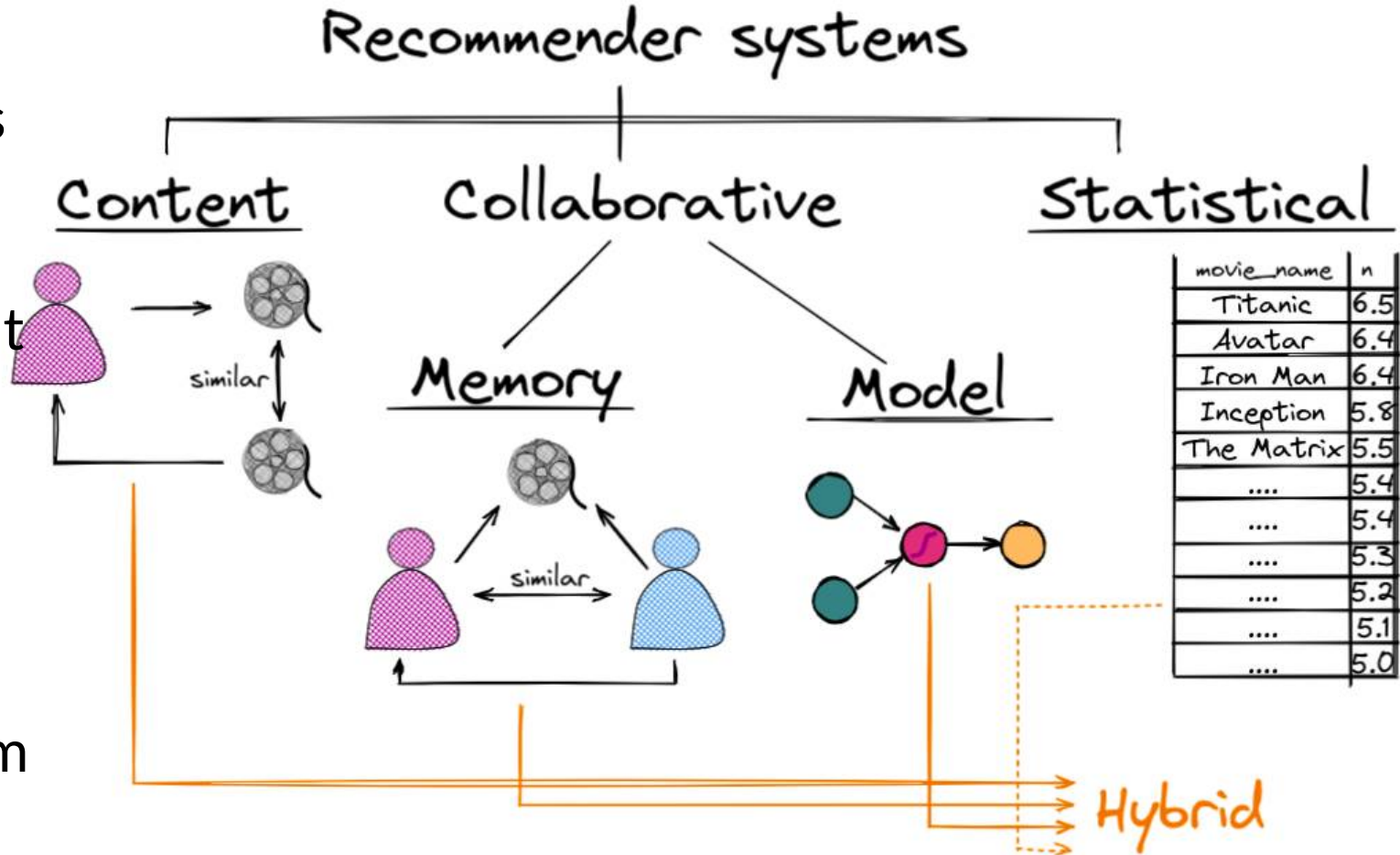
features outputs

As both methods use user <-> movie relationships, both of them are considered collaborative. They are distinguished as **model** and **memory** based respectively.

# Hybrid systems

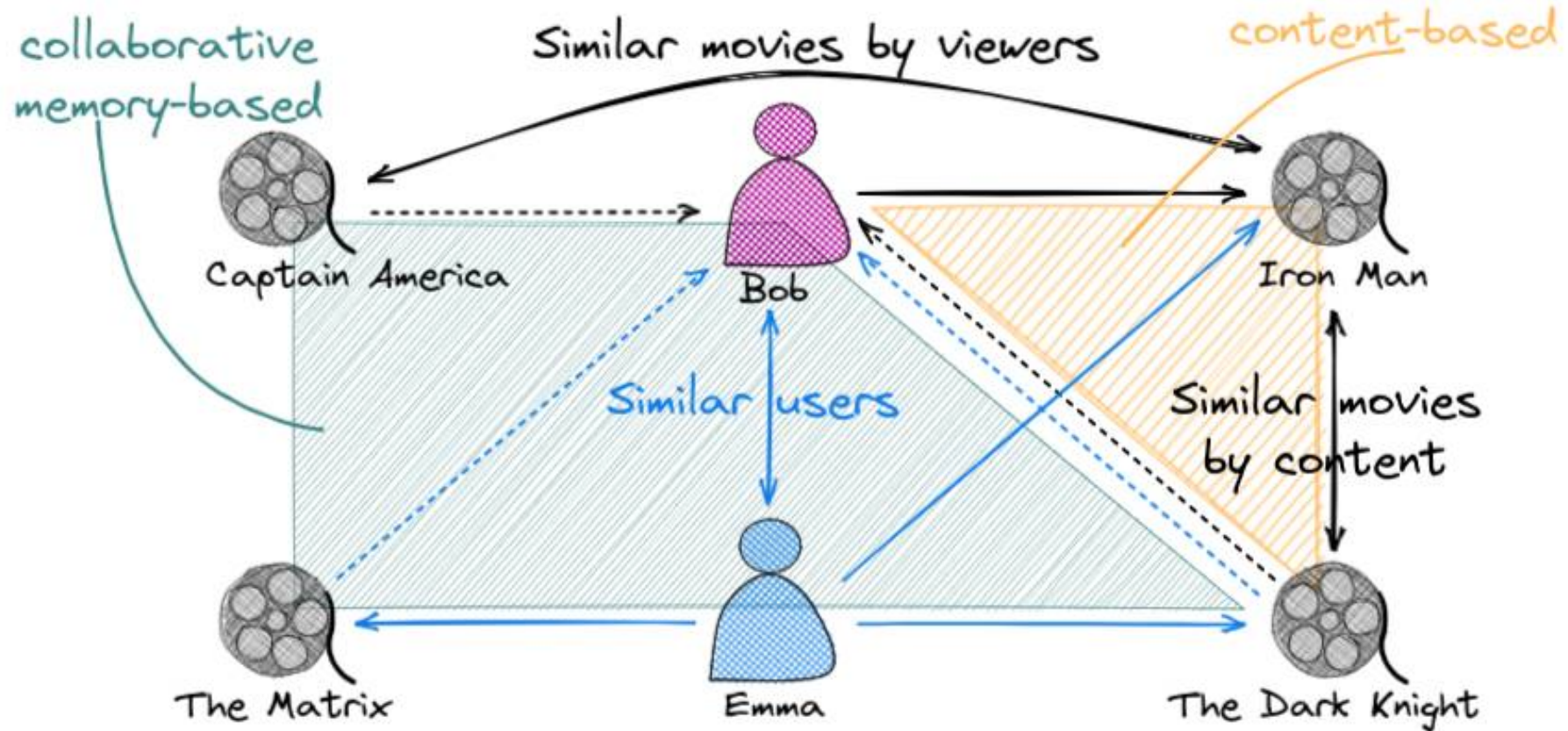
Hybrid systems are built to combine the pros of different approaches.

The resulting solution can be considered as a ranking problem





# Graph systems



# Graph systems

Graph recommendation systems can outperform traditional hybrid systems due to the synergy gained by combining content and collaborative approaches for multiple types of nodes.

Much more information might be used describing nodes and we are not limited by the number of types

