Intro to Recommender Systems

Dr. Rubi Boim

TLDR for today

Recommending by

- Generate a list of candidate items
- **For all items predict the probability of the event**
- Select the top k with the highest score (or apply diversity / …)

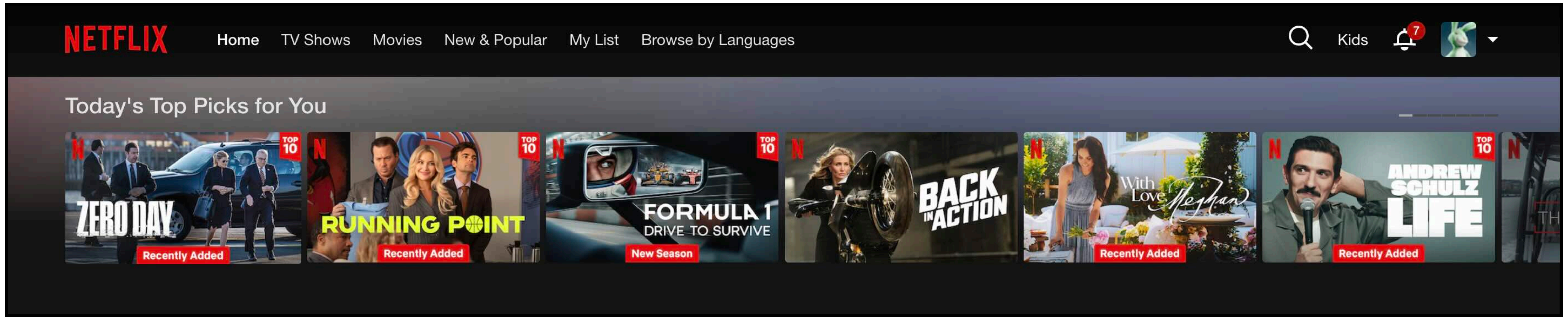
Recommender system types

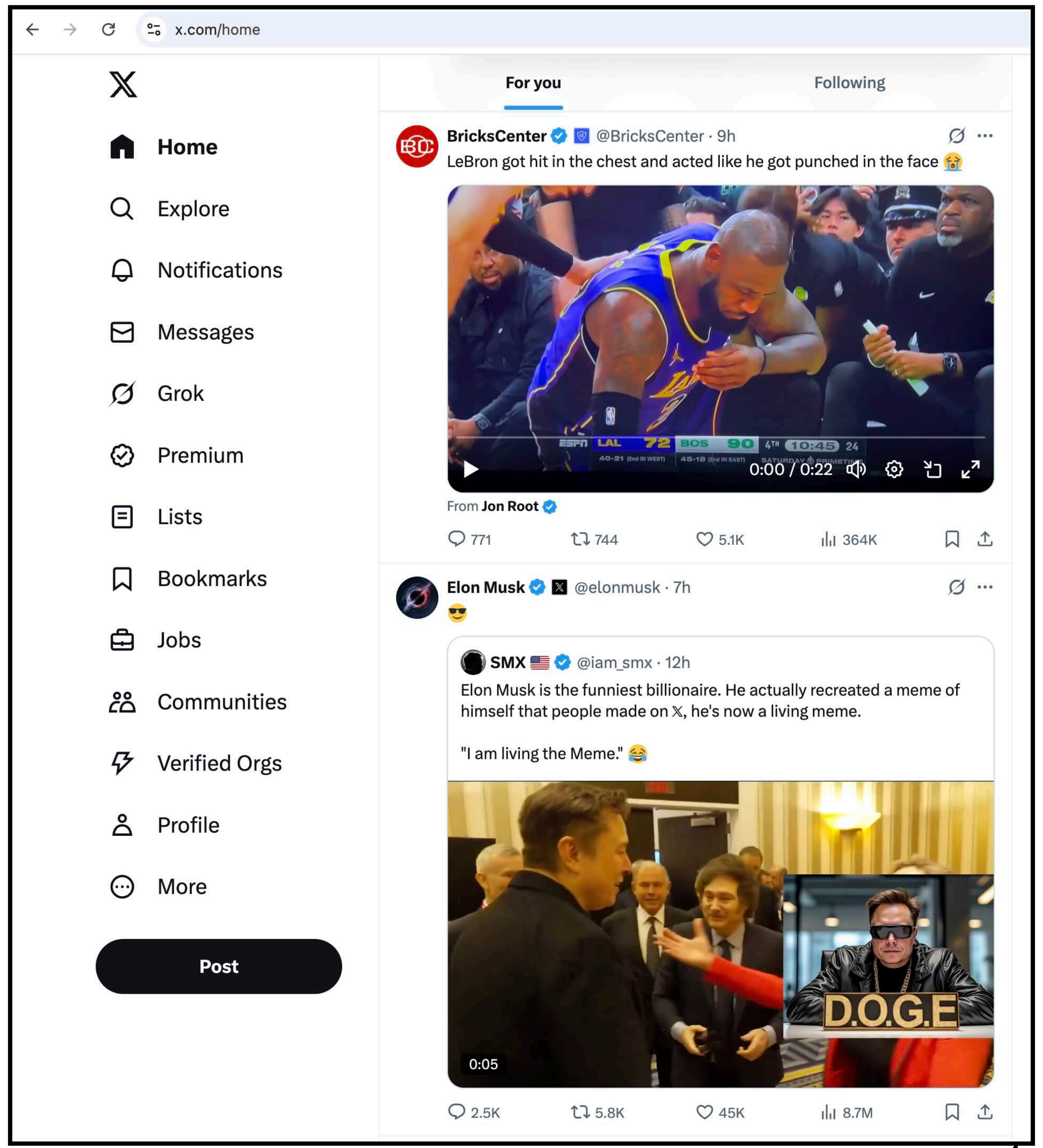
- Content Based
- Collaborative Filtering
- Hybrid

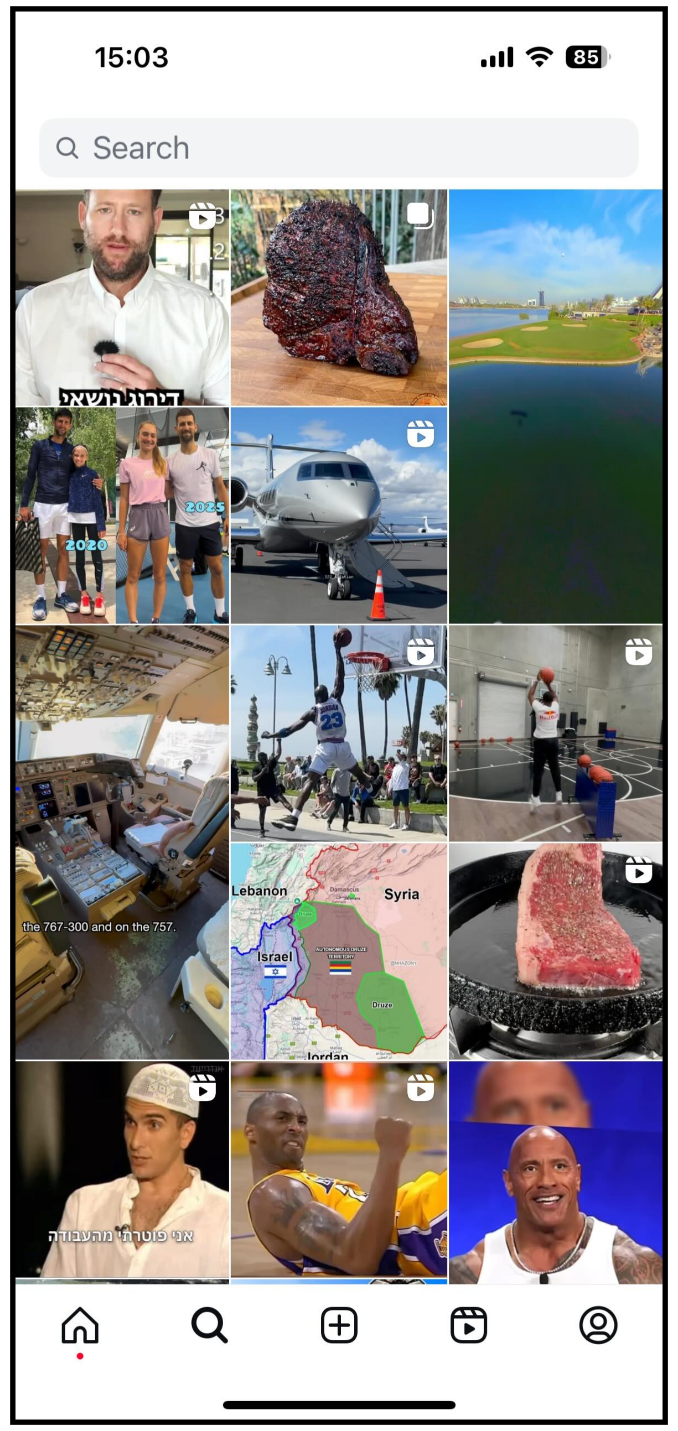
Agenda

- **Intro and Intuition**
- Content Based
- Collaborative Filtering
- Common challenges

How can you build these?

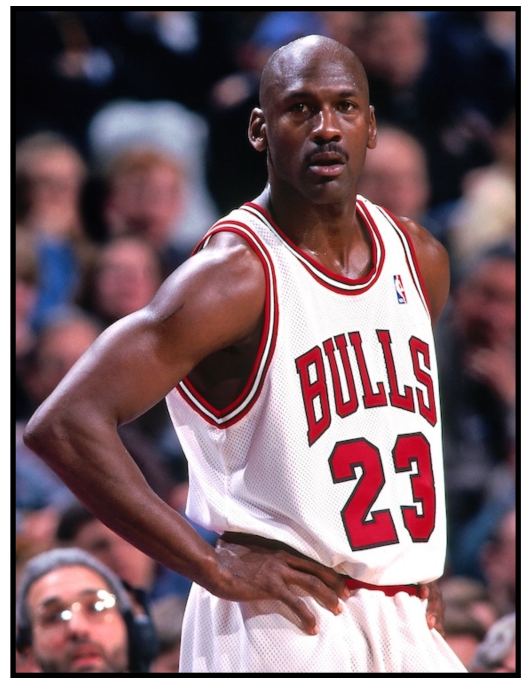


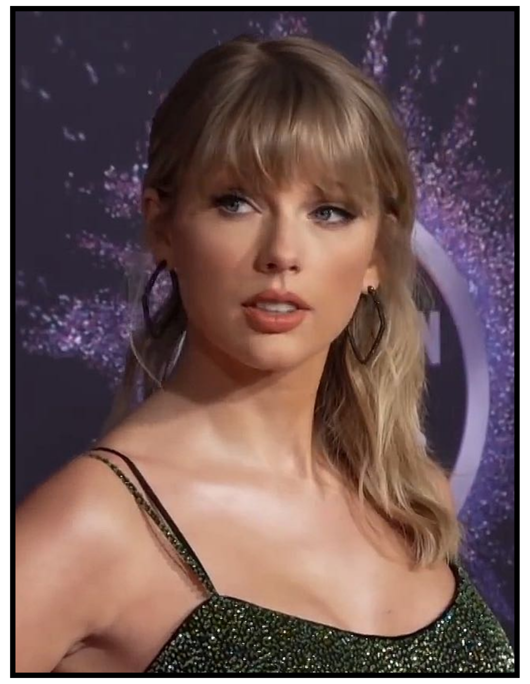




Attempt 1

An editor manually selects the items order





Attempt 1

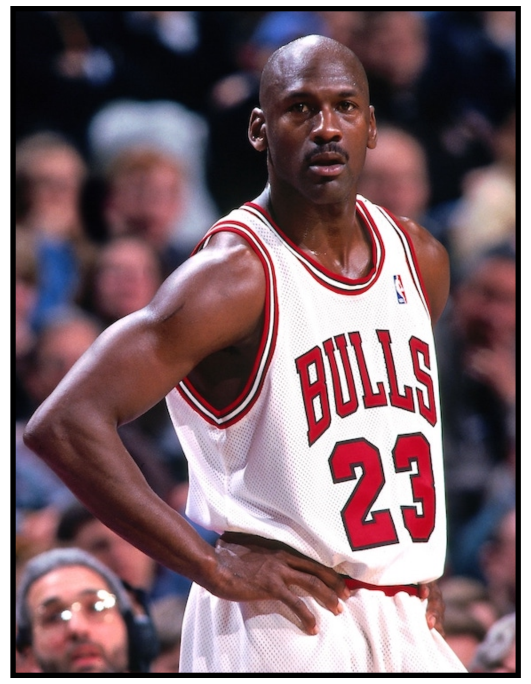
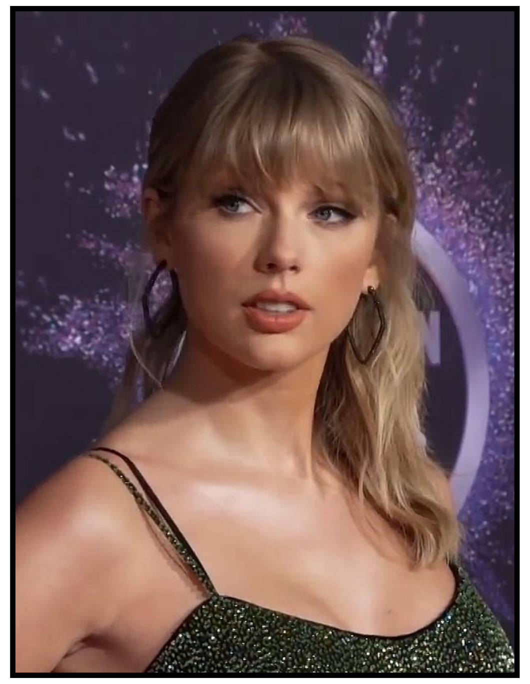
All users gets the same recommendations

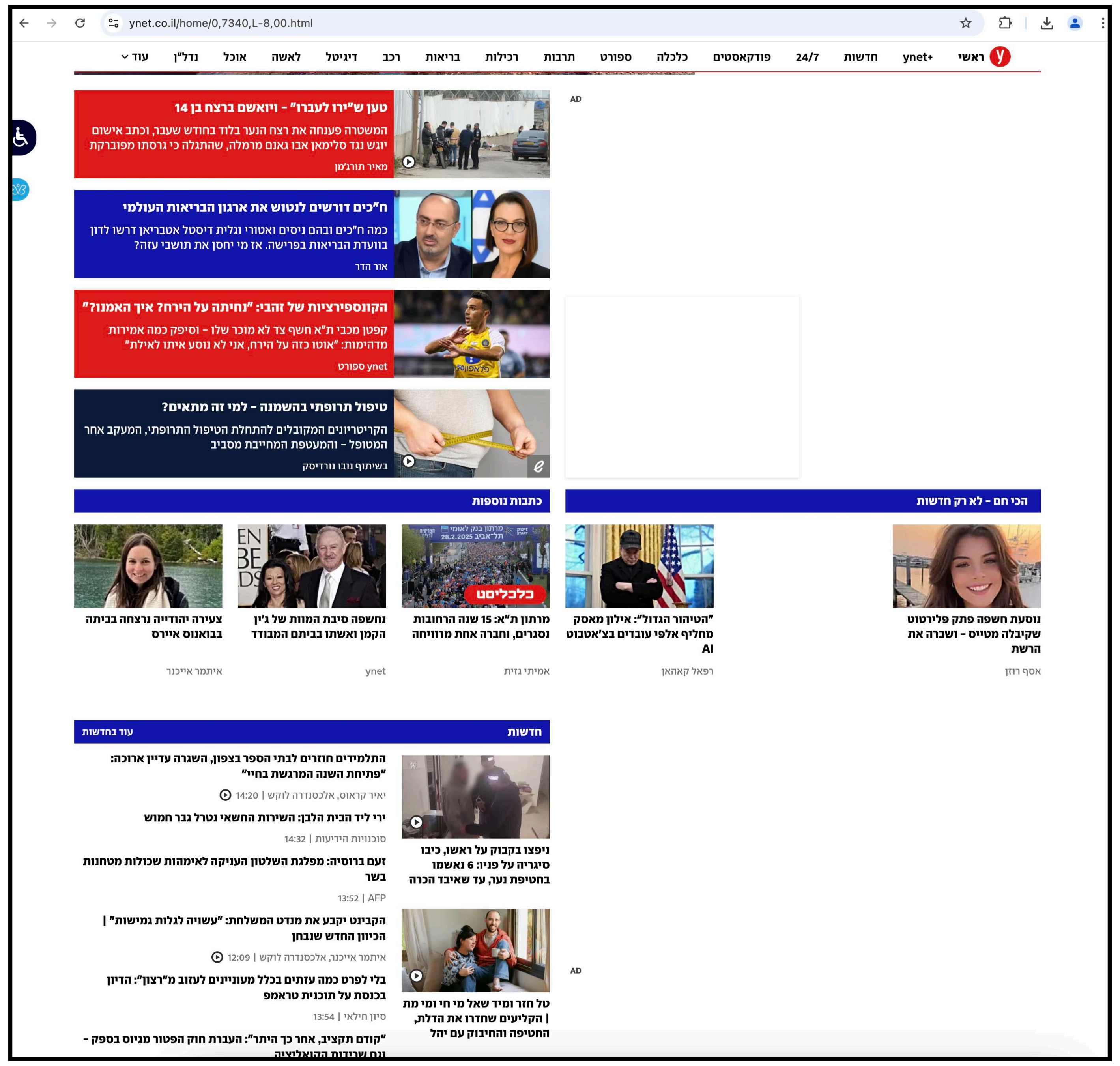
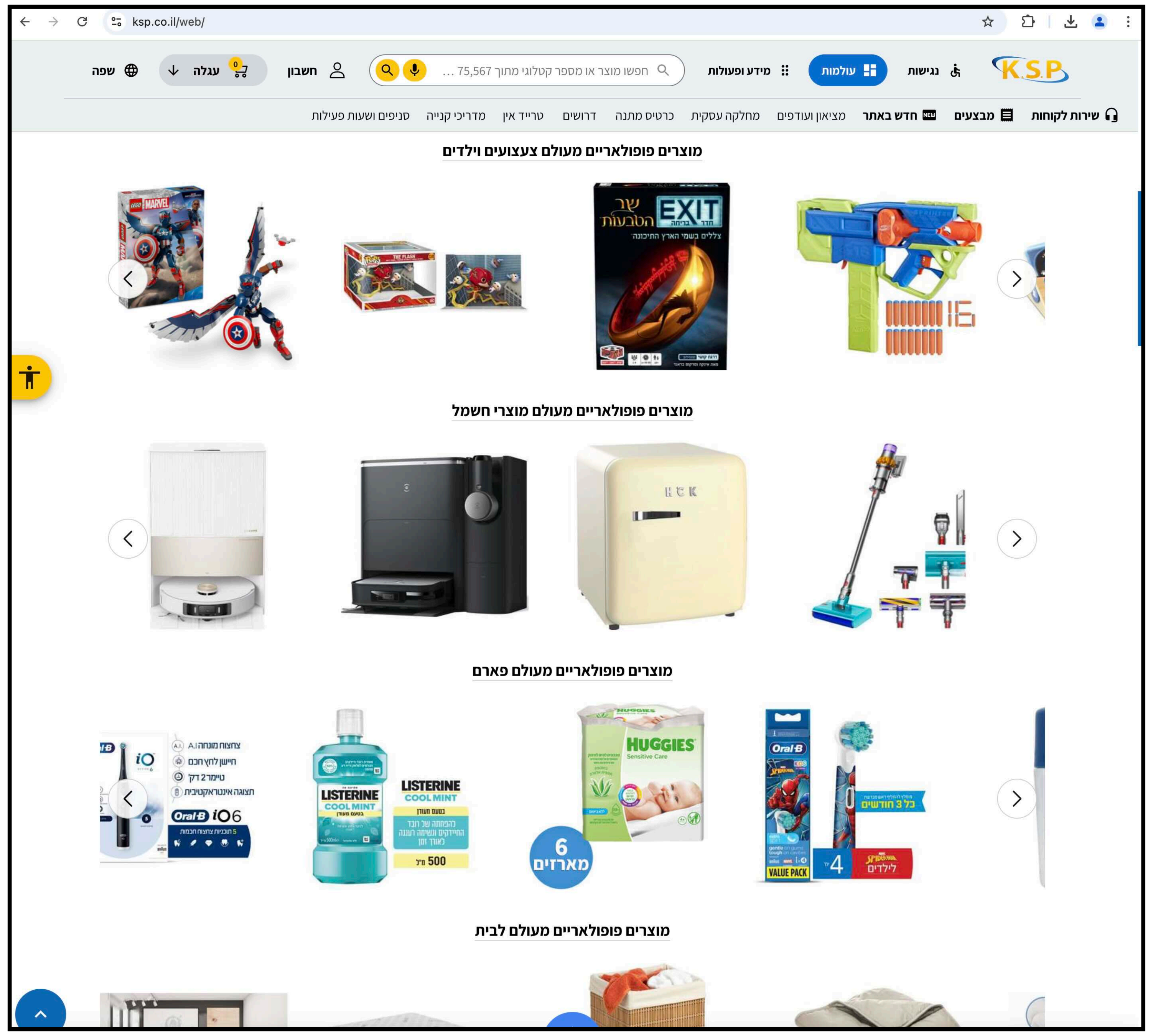
An editor manually selects the items order

Attempt 1

All users gets the same recommendations

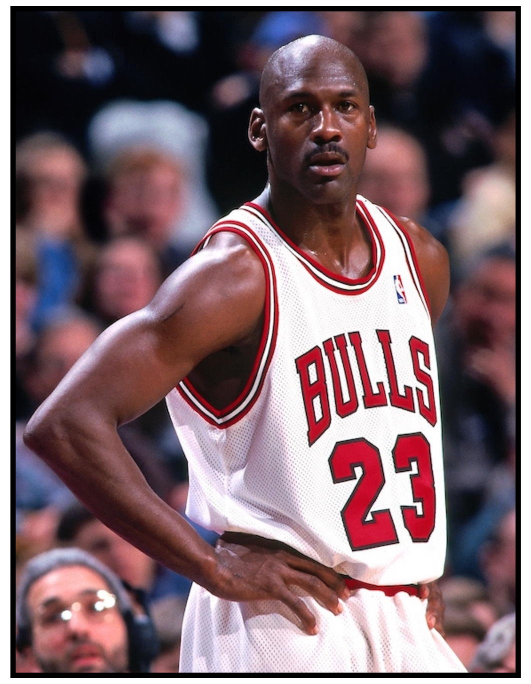
An editor manually selects the items order

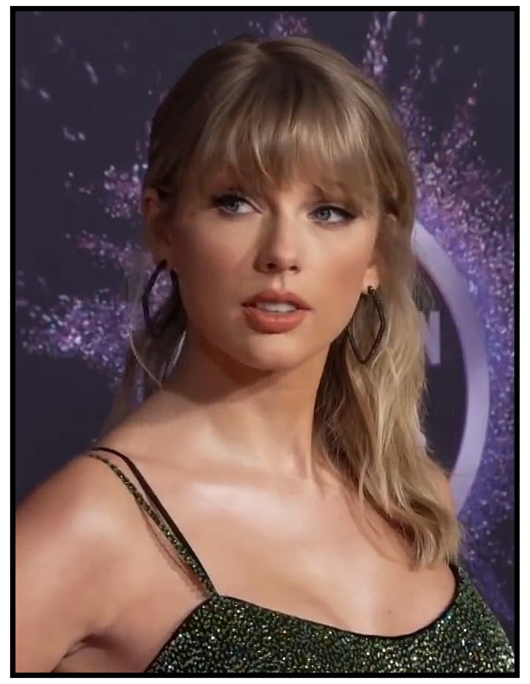



Attempt 2

Recommending weekly trending





Attempt 2

Recommending weekly trending

All users gets the same recommendations

(but it may change each day)

Attempt 2

Recommending weekly trending

How do you calculate it?

Attempt 2

1 - count the number of views for all movies

Top Gun	8232
The Matrix	103294
Inception	32124

...

2 - order the list and select top 4

How do you calculate it?

Attempt 2

1 - count the number of views for all movies

Top Gun	8232
The Matrix	103294
Inception	32124

...

2 - order the list and select top 4

How do you calculate it?

Congratulations!

This is the first “model” you built :)

Attempt 2

1 -
Top
The
Inc
…

2 -

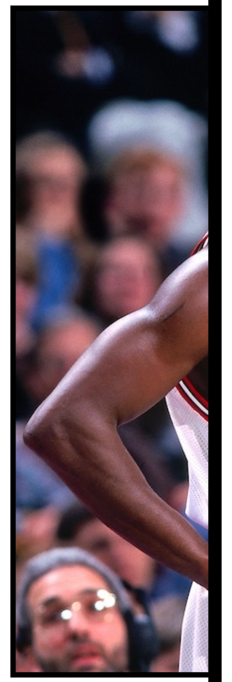
Offline / Training / Building
can take minutes / hours / days…

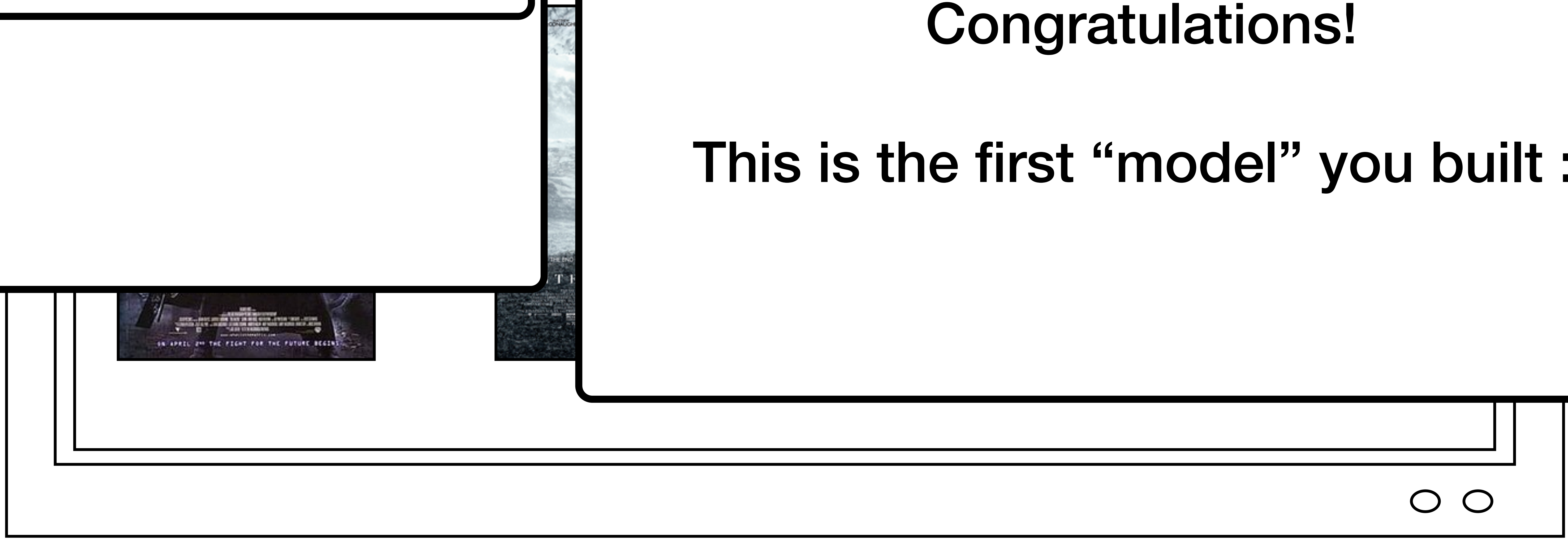
Online / Serving / Inference
in milliseconds

How do you calculate it?

Congratulations!

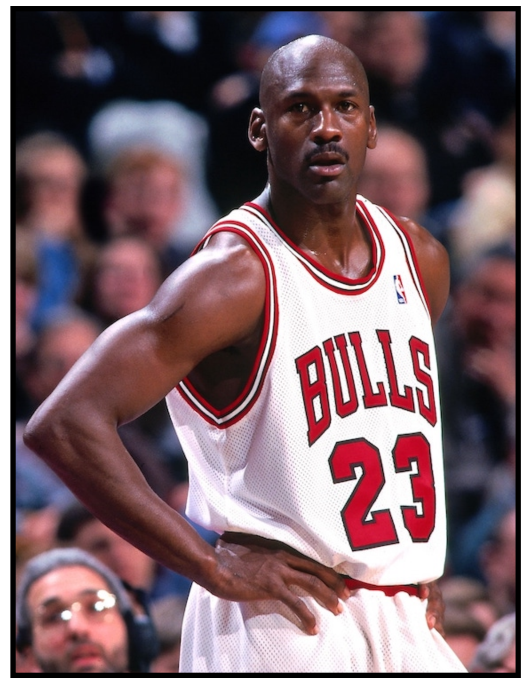
This is the first "model" you built :)

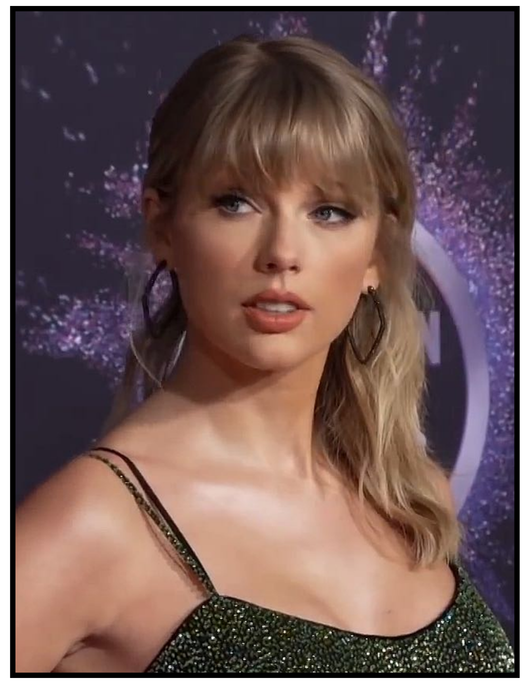




Attempt 3

Recommending weekly trending per country

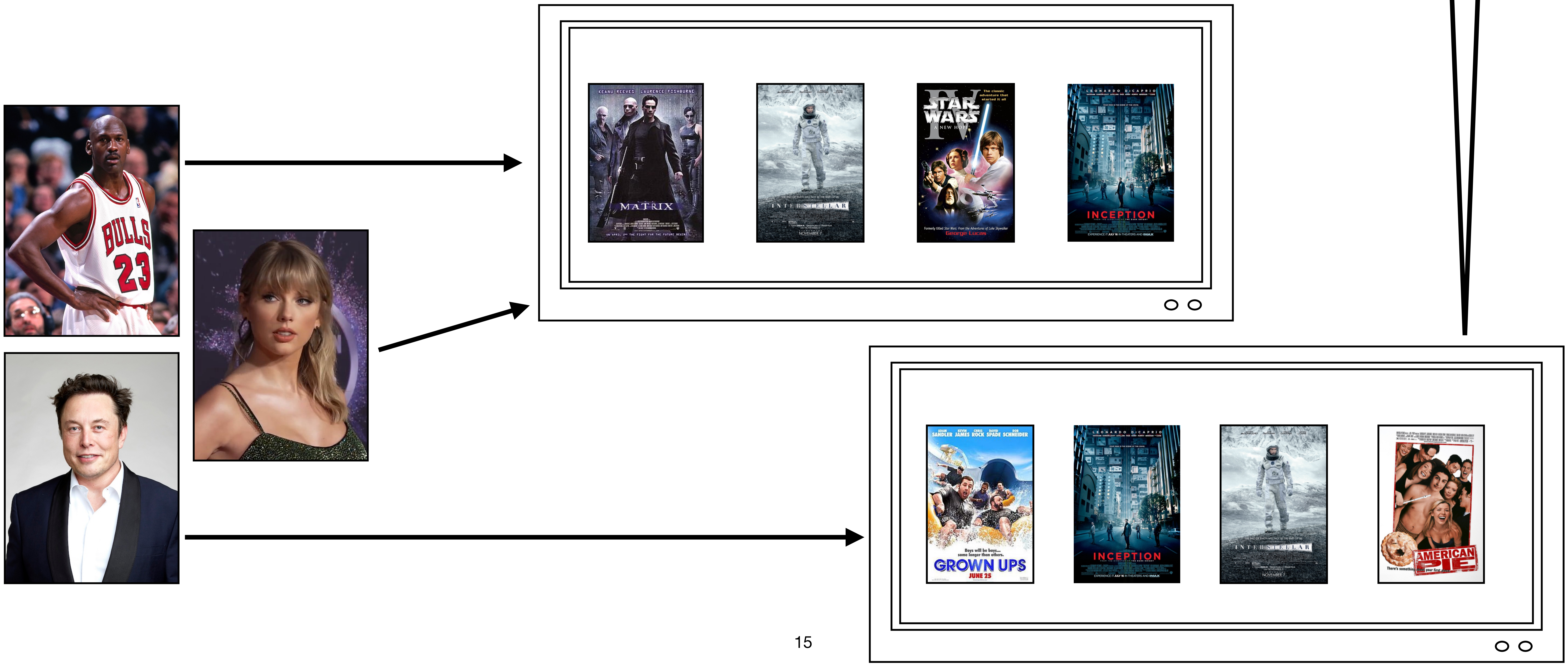




Attempt 3

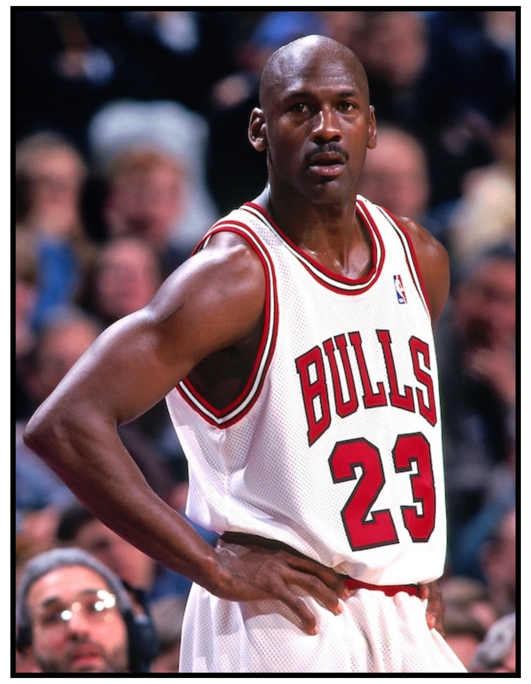
Still not really personalized
(How many people are in India?)

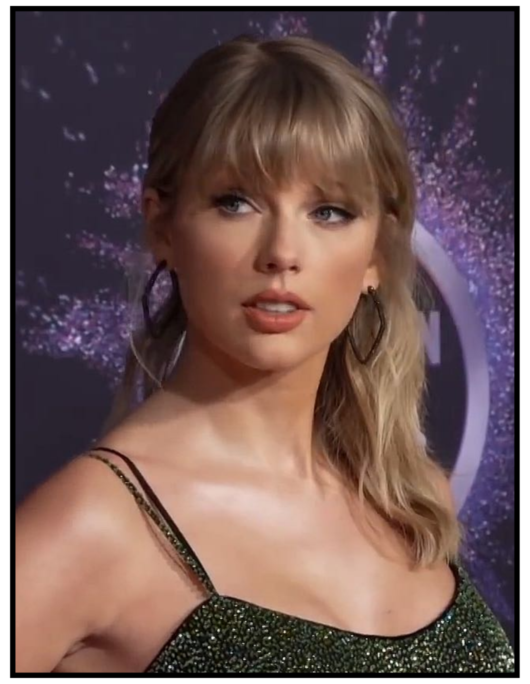
Recommending weekly trending per country



Attempt 4

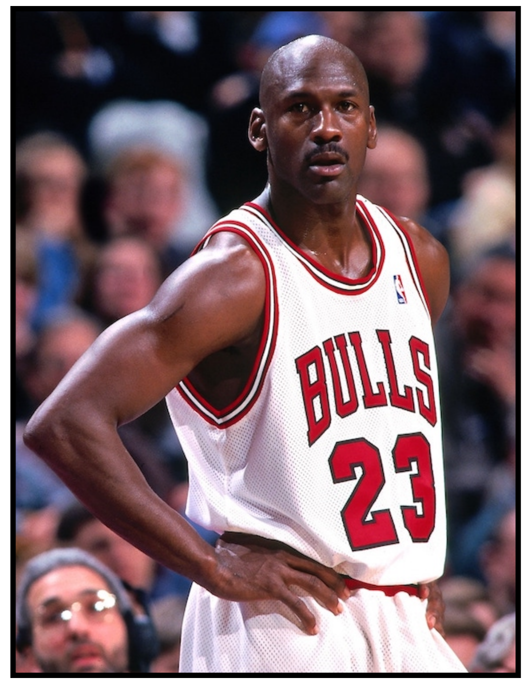
For each user, predict which items they will like

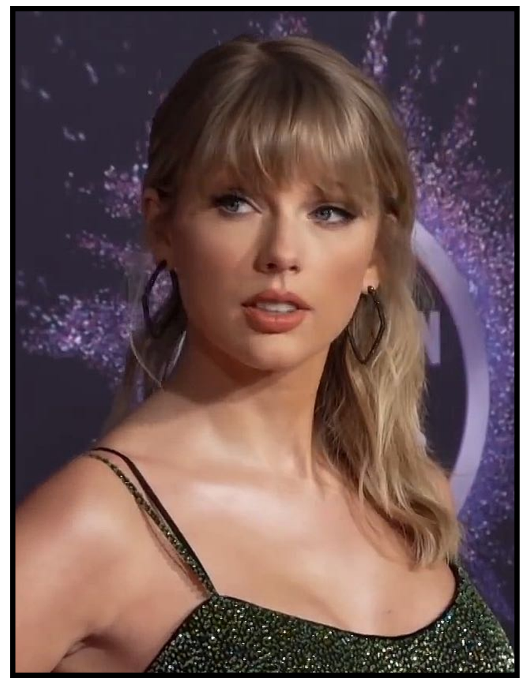




Attempt 4

For each user, predict which items they will like





finally…
BUT HOW DO YOU DO IT???

Recommender System 101

- Generate a list of candidate items
- **For all items predict the probability of the event**
- Select the top k with the highest score
  (or apply diversity / …)

Recommender System 101 (example)

- Generate a list of candidate items
  all the movies with license in Israel which the user had not seen
- **For all items predict the probability of the event**
  Top Gun 2: 0.84, The Matrix: 0.62, American Pie: 0.94
- Select the top k with the highest score
  (or apply diversity / …)
  American Pie, Top Gun 2

Recommender System 101 (example)

- Generate a list of candidate items
  all the movies with license in Israel which the user had not seen
- **For all items predict the probability of the event**
  Top Gun 2: 0.84, The Matrix: 0.62, American Pie: 0.94
- Select the top k with the highest score
  (or apply diversity / …)
  American Pie, Top Gun 2

how do we calculate this?

Recommender System types

- Content based
  based on semantic / static properties
- Collaborative filtering
  based on user behavior
- Hybrid
  A mix of the two

Recommender System types

- Content based
  based on semantic / static properties

- Collaborative filtering
  based on user behavior

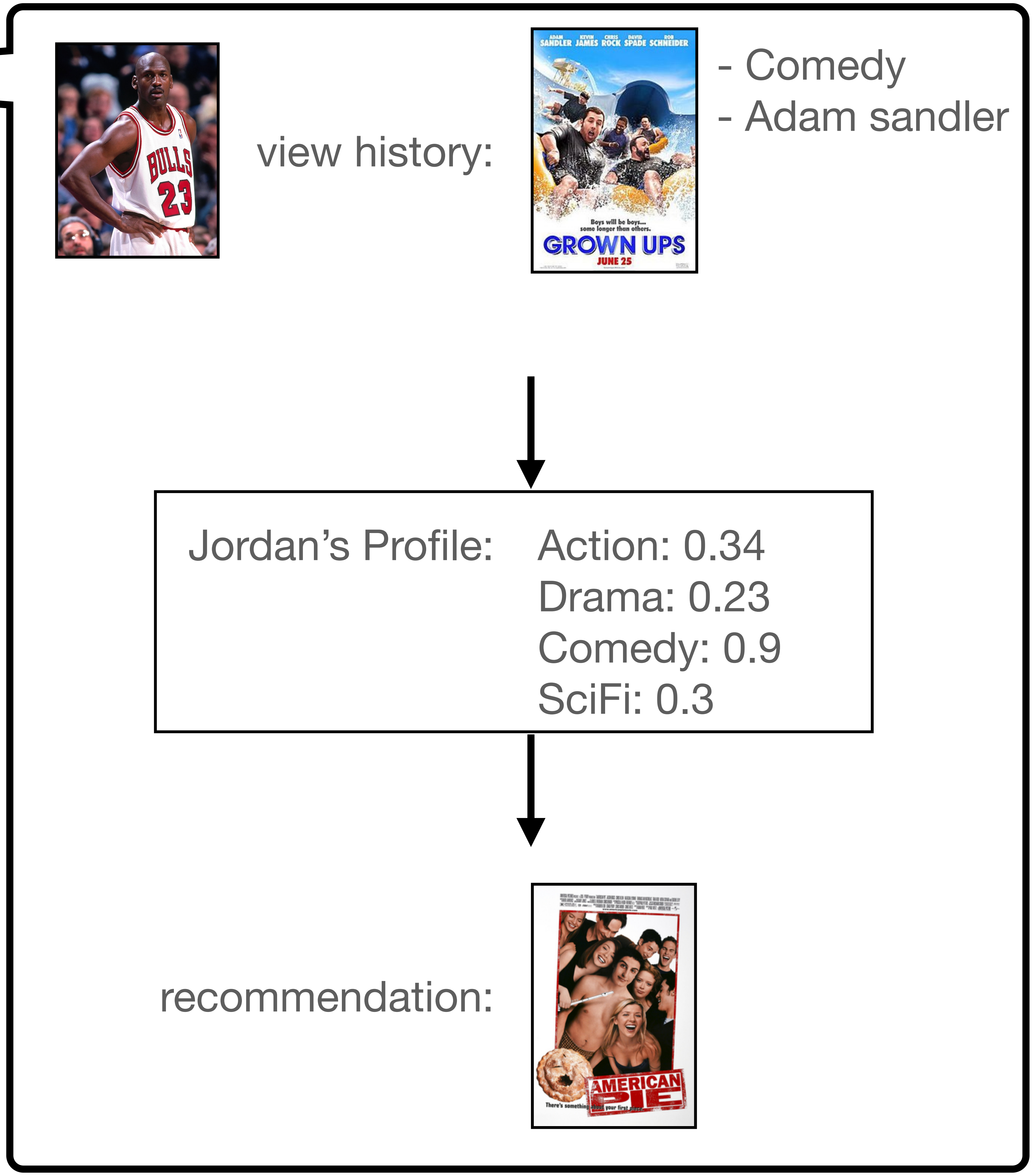
- Hybrid
  A mix of the two

Intuition

because you watch comedy movies with Adam Sandler here are more comedies you might like

Recommender System types

- Content based
  based on semantic / static properties



- Collaborative filtering
  based on user behavior

- Hybrid
  A mix of the two

Recommender System types

- Content based
  based on semantic / static properties
- Collaborative filtering
  based on user behavior
- Hybrid
  A mix of the two

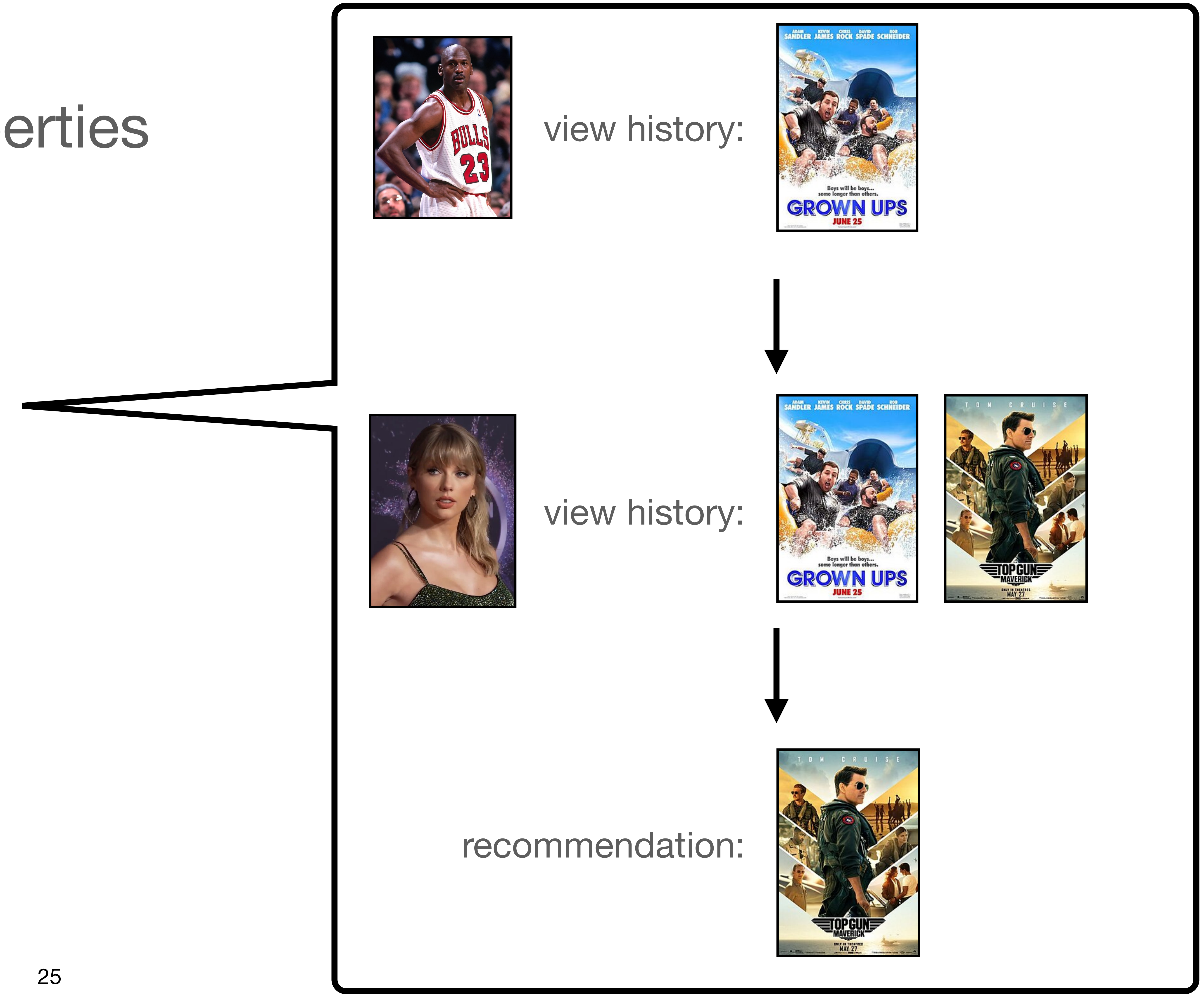
Intuition

Other users who watch “Grown Ups” also watched…

Recommender System types

- Content based
  based on semantic / static properties

- Collaborative filtering
  based on user behavior



- Hybrid
  A mix of the two

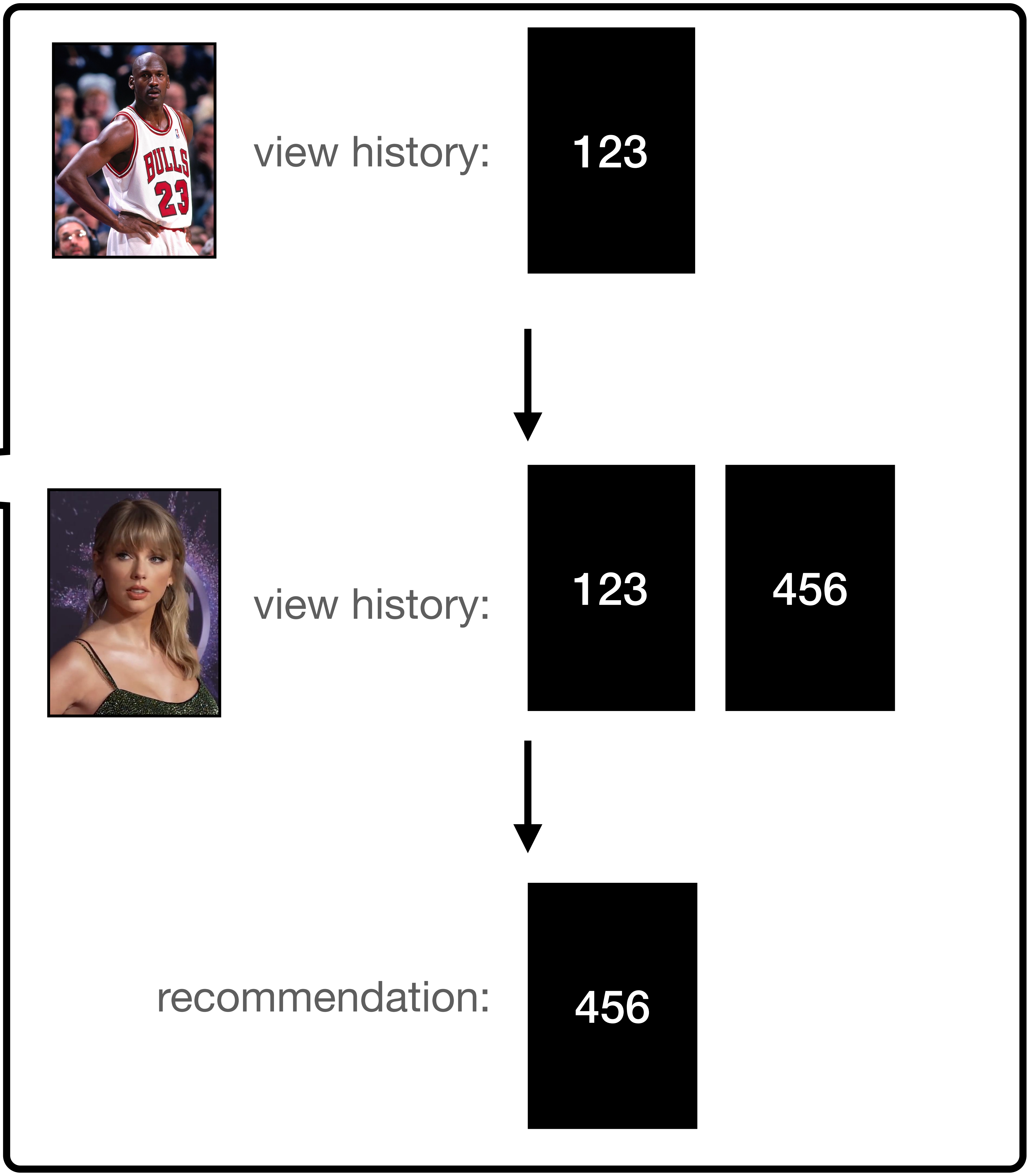
Recommender System types

perties

THIS IS IMPORTANT!

Collaborative Filtering works without any prior semantic knowledge

You can blend any type of items together in the same algorithm (movies, food, books, hotels..)



Recommender System types

- Content based
  based on semantic / static properties

- Collaborative filtering
  based on user behavior

- Hybrid
  A mix of the two

Intuition

Mix the semantics with CF
(there are a lot of ways)

for example, predict new genres the user might like

A short history lesson - “The Netflix Prize”

In 2006 Netflix offer $1M to the team who will improve their recsys by 10% (RMSE error)

- First “big” public data set
  0.5M user, 17k movies, 100M ratings (1-5)

- Hugh leap forward for recommender systems
  and for ML competitions —> Kaggle started in 2010

A short history lesson - “The Netflix Prize”

It was proven that CF is better than CB

THIS IS COUNTER INTUITIVE

Think about it, in order to predict movie ratings you do not need to know *any* semantic information such as actor, genre, year…

In practice btw you would use a hybrid approach

team who will improve

or)

1-5)

mender systems

tarted in 2010

Agenda

- Intro and Intuition
- **Content Based**
- Collaborative Filtering
- Common challenges

Content Based

- Build a profile for each item using its features (attributes)
  genres / categories / actors / release date / price / …

- Build a profile for each user
  but extracting the features from the items she previously interact with (explicit / implicit)

- Recommend items with similar profiles to the user

Content Based

- Build a profile for each item using its features (attributes)
  genres / categories / actors / release date / price / …

- Build a profile for each user
  but extracting the features from the items she previously interact with (explicit / implicit)

- Recommend items with similar profiles to the user

But how do you
(1) create profiles
(2) compare profiles

Feature extraction (by data types)

- Structured metadata
  genres / categories / actors / release date / price / …

- Free text
  description / plot / …

- Visual
  images / videos

Feature extraction (by data types)

- Structured metadata
  genres / categories / actors / release date / price / …

Already extracted (structured)

(You just might want to filter out some)

- Free text
  description / plot / …

- Visual
  images / videos

Feature extraction (by data types)

- Structured metadata
  genres / categories / actors / release date / price / …

- Free text
  description / plot / …

Extract the features from the text (Tokenization, filter stop words, apply stemming…)

- Visual
  images / videos

"Top Gun: Maverick is a 2022 American action drama film directed by Joseph Kosinski and written by Ehren Kruger, Eric Warren Singer, and Christopher McQuarrie. A sequel to the 1986 film Top Gun, Tom Cruise reprises his starring role as the naval aviator Maverick."

"Top Gun: Maverick is a 2022 American action drama film directed by Joseph Kosinski and written by Ehren Kruger, Eric Warren Singer, and Christopher McQuarrie. A sequel to the 1986 film Top Gun, Tom Cruise reprises his starring role as the naval aviator Maverick."

Tokenization

[Top, Gun, Maverick, is, a, 2022, American, action, drama, film, directed, by, Joseph, Kosinski, and, written, by, Ehren, Kruger, Eric, Warren, Singer, and, Christopher, McQuarrie, A, sequel, to, the, 1986, film, Top, Gun, Tom, Cruise, reprises, his, starring, role, as, the, naval, aviator, Maverick]

"Top Gun: Maverick is a 2022 American action drama film directed by Joseph Kosinski and written by Ehren Kruger, Eric Warren Singer, and Christopher McQuarrie. A sequel to the 1986 film Top Gun, Tom Cruise reprises his starring role as the naval aviator Maverick."

Tokenization

[Top, Gun, Maverick, is, a, 2022, American, action, drama, film, directed, by, Joseph, Kosinski, and, written, by, Ehren, Kruger, Eric, Warren, Singer, and, Christopher, McQuarrie, A, sequel, to, the, 1986, film, Top, Gun, Tom, Cruise, reprises, his, starring, role, as, the, naval, aviator, Maverick]

Removing Stop Words

[Top, Gun, Maverick, 2022, American, action, drama, film, directed, Joseph, Kosinski, written, Ehren, Kruger, Eric, Warren, Singer, Christopher, McQuarrie, sequel, 1986, film, Top, Gun, Tom, Cruise, reprises, starring, role, naval, aviator, Maverick]

"Top Gun: Maverick is a 2022 American action drama film directed by Joseph Kosinski and written by Ehren Kruger, Eric Warren Singer, and Christopher McQuarrie. A sequel to the 1986 film Top Gun, Tom Cruise reprises his starring role as the naval aviator Maverick."

Tokenization

[Top, Gun, Maverick, is, a, 2022, American, action, drama, film, directed, by, Joseph, Kosinski, and, written, by, Ehren, Kruger, Eric, Warren, Singer, and, Christopher, McQuarrie, A, sequel, to, the, 1986, film, Top, Gun, Tom, Cruise, reprises, his, starring, role, as, the, naval, aviator, Maverick]

Removing Stop Words

[Top, Gun, Maverick, 2022, American, action, drama, film, directed, Joseph, Kosinski, written, Ehren, Kruger, Eric, Warren, Singer, Christopher, McQuarrie, sequel, 1986, film, Top, Gun, Tom, Cruise, reprises, starring, role, naval, aviator, Maverick]

Stemming

[top, gun, maverick, 2022, american, action, drama, film, direct, joseph, kosinski, writ, ehren, kruger, eric, warren, singer, christopher, mcquarri, sequel, 1986, film, top, gun, tom, cruise, reprais, starr, role, naval, aviator, maverick]

Feature extraction (by data types)

- Structured metadata
  genres / categories / actors / release date / price / …

- Free text
  description / plot / …

- Visual
  images / videos

Convert to free text, then extract…

s)

OK - so we have these features for Top Gun 2

[top, gun, maverick, 2022, american, action, drama, film, direct, joseph, kosinski, writ, ehren, kruger, eric, warren, singer, christopher, mcquarri, sequel, 1986, film, top, gun, tom, cruise, reprais, starr, role, naval, aviator, maverick]

What do we do next?

Vectorization

Converting tokens into numerical representations
so they could be processed by ML

- BoW / TF-IDF
  simple and effective

- Embeddings
  complex models to build,
  but superior results,
  and (maybe) easier “inference”

Vectorization

Converting tokens into numerical representations
so they could be processed by ML

- BoW / TF-IDF
  simple and effective

- Embeddings
  complex models to build,
  but superior results,
  and (maybe) easier “inference”

Can “handle” for typos like
“tel-aviv” and “tel aviv”

Can also understand “meaning”

i want to eat an apple
vs
apple just released a new phone

BoW / TF-IDF

Reflect how important a word is to a document in a collection

- Create a dictionary of all words
  the space

* without word order

	id	weight
action	1	X
drama	2	X
animated	3	
…		
jordan	594	
cruise	595	X
…		

Bag of Words

- Counts the number of appearances in the text

  If the data is structured — > 1

	id	weight
action	1	1
drama	2	1
animated	3	
…		
jordan	594	
cruise	595	3
…		

TF-IDF

- Term Frequency
  weight by the frequency in the document

- Inverse Document Frequency
  measures how common or rare a term is across all documents D in the corpus:

$$\mathrm{TF}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$$

$$\mathrm{IDF}(t,D) = \log\left(\frac{N}{|\{d \in D : t \in d\}|}\right)$$

$$\text{TF-IDF}(t,d,D) = \mathrm{TF}(t,d) \times \mathrm{IDF}(t,D)$$

	id	weight
action	1	1.42
drama	2	0.2
animated	3	
…		
jordan	594	
cruise	595	9.23
…		

TF-IDF - Notes

What is the “size" of the vector?

TF-IDF - Notes

What is the “size" of the vector?

- the dictionary size
  —> high dimension

- But most values are unknown
  —> sparse data

- Requires exact text match
  “tel aviv” != “tel-aviv”,   “car” != “automobile”

Embeddings

Dense vector representations where each feature is mapped into a continuous vector space

- Often generated using neural network models

- Capture semantic meaning by placing similar words or documents closer together in the embedding space

Embeddings

Dense vect ature is
mapped int

There are a lot of options:

Word2Vec
Sentence-BERT
embed-english-v3.0 (Cohere)
text-embedding-ada-002 (OpenAI)

- Often gen lels

- Capture semantic meaning by placing similar words or documents closer together in the embedding space

Embeddings - Notes

What is the “size" of the vector?

- A fixed (relatively small) size
  300-2000 in most of today’s models

- Most values are known
  —> dense

- Captures semantics
  “tel aviv” ~= “tel-aviv”,   “car” ~= “automobile”

TD-IDF vs Embeddings

Aspect	TF-IDF	Embeddings
Representation	Sparse, high-dimensional	Dense, low-dimensional
Interpretability	Highly interpretable (words are explicit)	Less interpretable (dimensions are abstract)
Semantic Capture	Limited semantic understanding	Rich semantic understanding
Computational Cost	Relatively low	Can be higher (especially during training)
Usage	Suitable for smaller datasets or when interpretability is key	Ideal for larger datasets and when capturing nuance is critical

Vectorization - DONE!

We have a vector representation for each item

- What is the next move?

Cosine Similarity

Used to compare 2 vectors

popular choice for TF-IDF and Embeddings

- calculates the cosine of the angle between two vectors

  If the vectors point in the same direction (angle is 0°),
  the cosine similarity is 1, indicating maximum similarity

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}},$$

Agenda

- Intro and Intuition
- Content Based
- **Collaborative Filtering**
- Common challenges

Collaborative Filtering

Recommend based on the behavior and preferences of other users

Memory based CF

- User-based CF
- Item-based CF

Model based CF

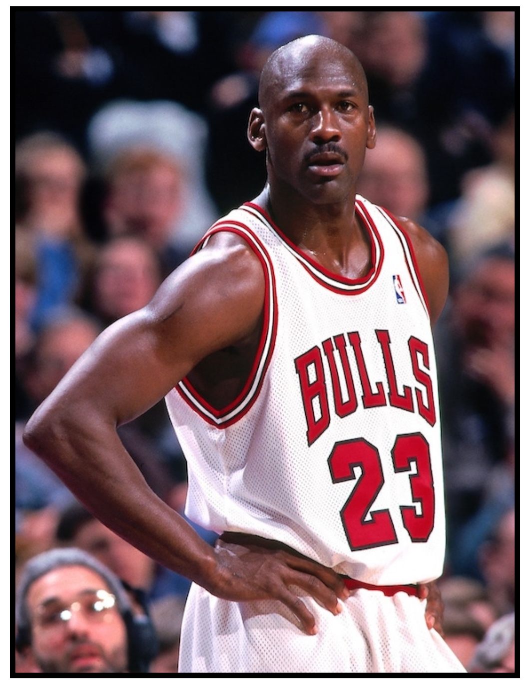
- Matrix factorization

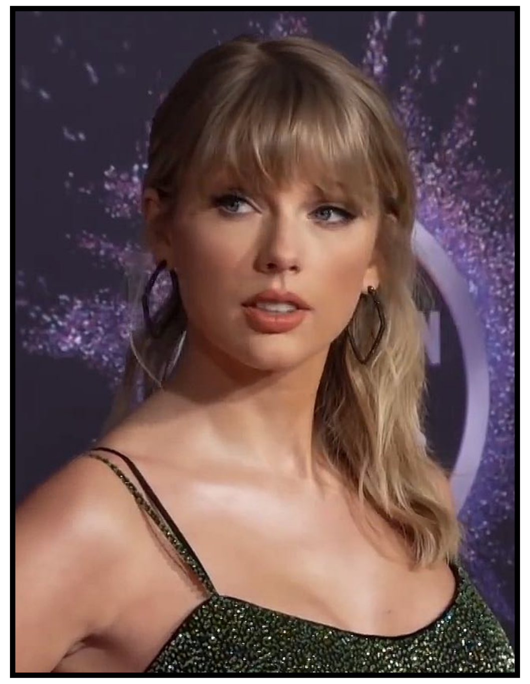
Users

Items

Events

Users






MOANA 2
ONLY IN THEATERS
NOVEMBER 27

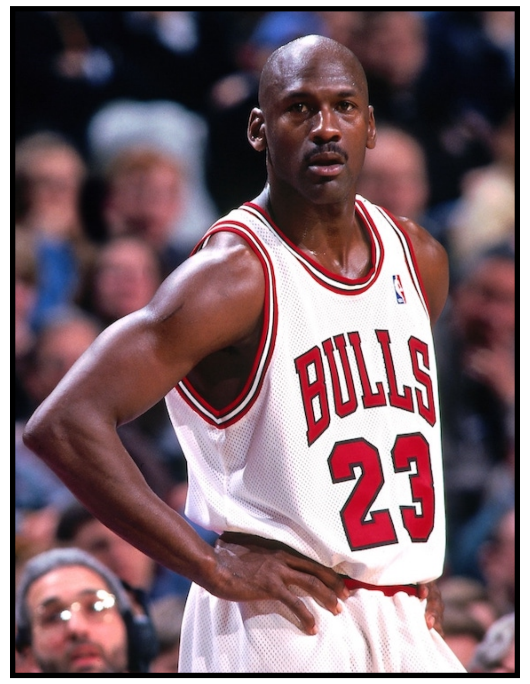
TOM CRUISE
TOP GUN
MAVERICK
ONLY IN THEATRES
MAY 27

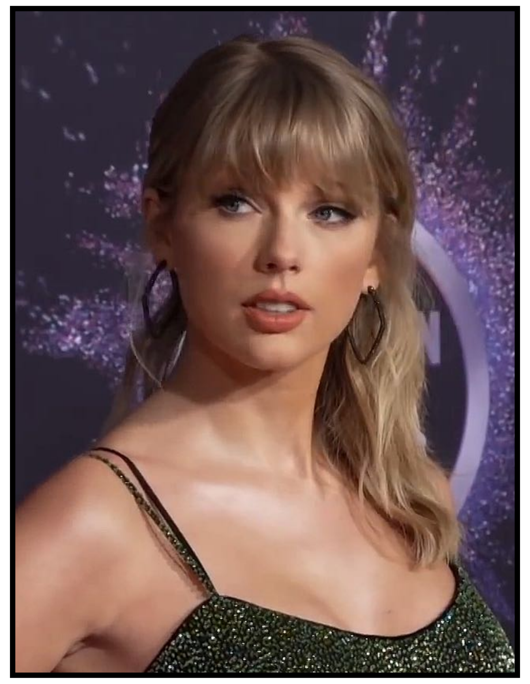
ADAM SANDLER
KEVIN JAMES
CHRIS ROCK
DAVID SPADE
ROB SCHNEIDER
Boys will be boys...
some longer than others.
GROWN UPS
JUNE 25

AMERICAN PIE

Items

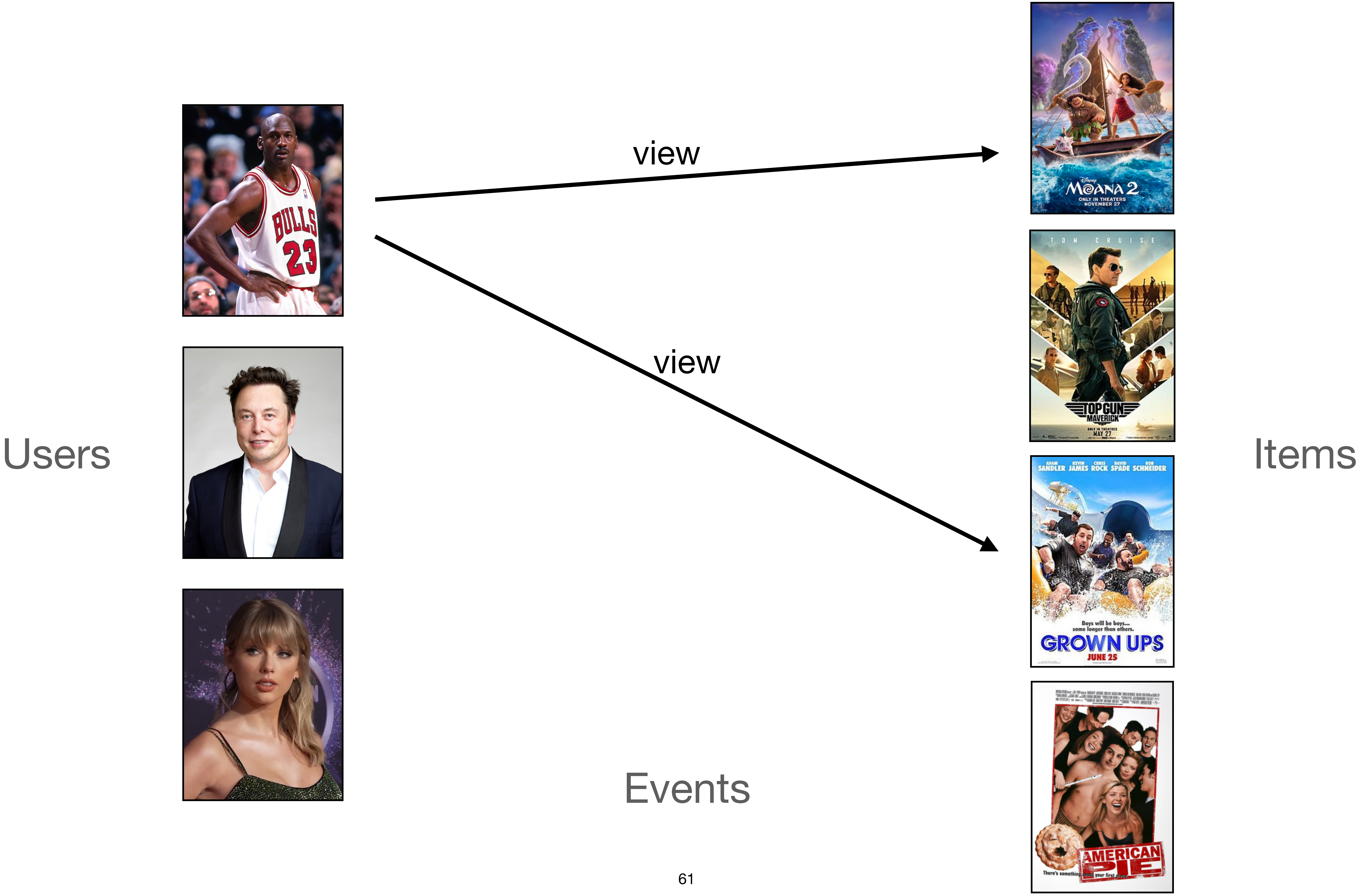
Users



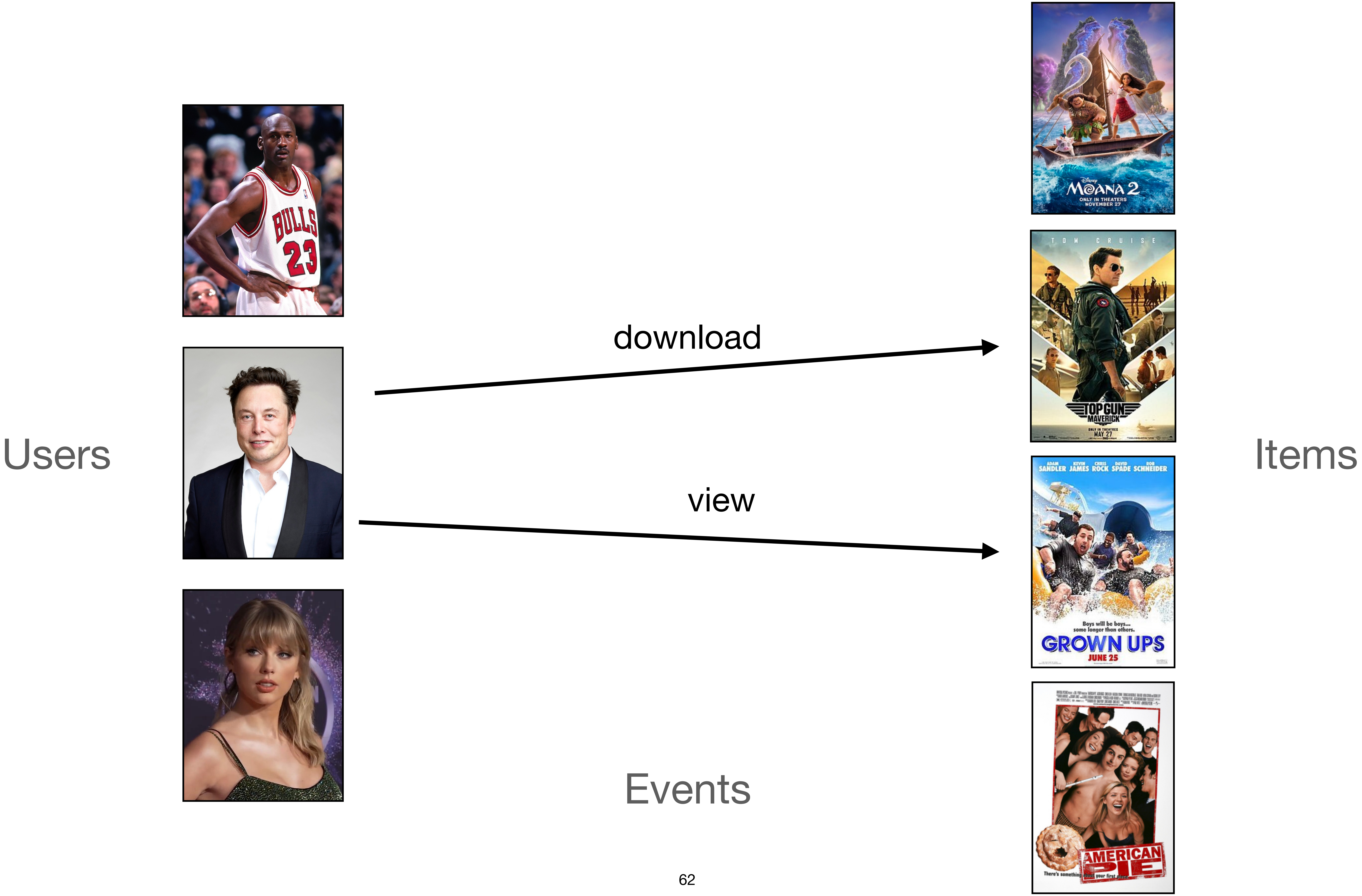


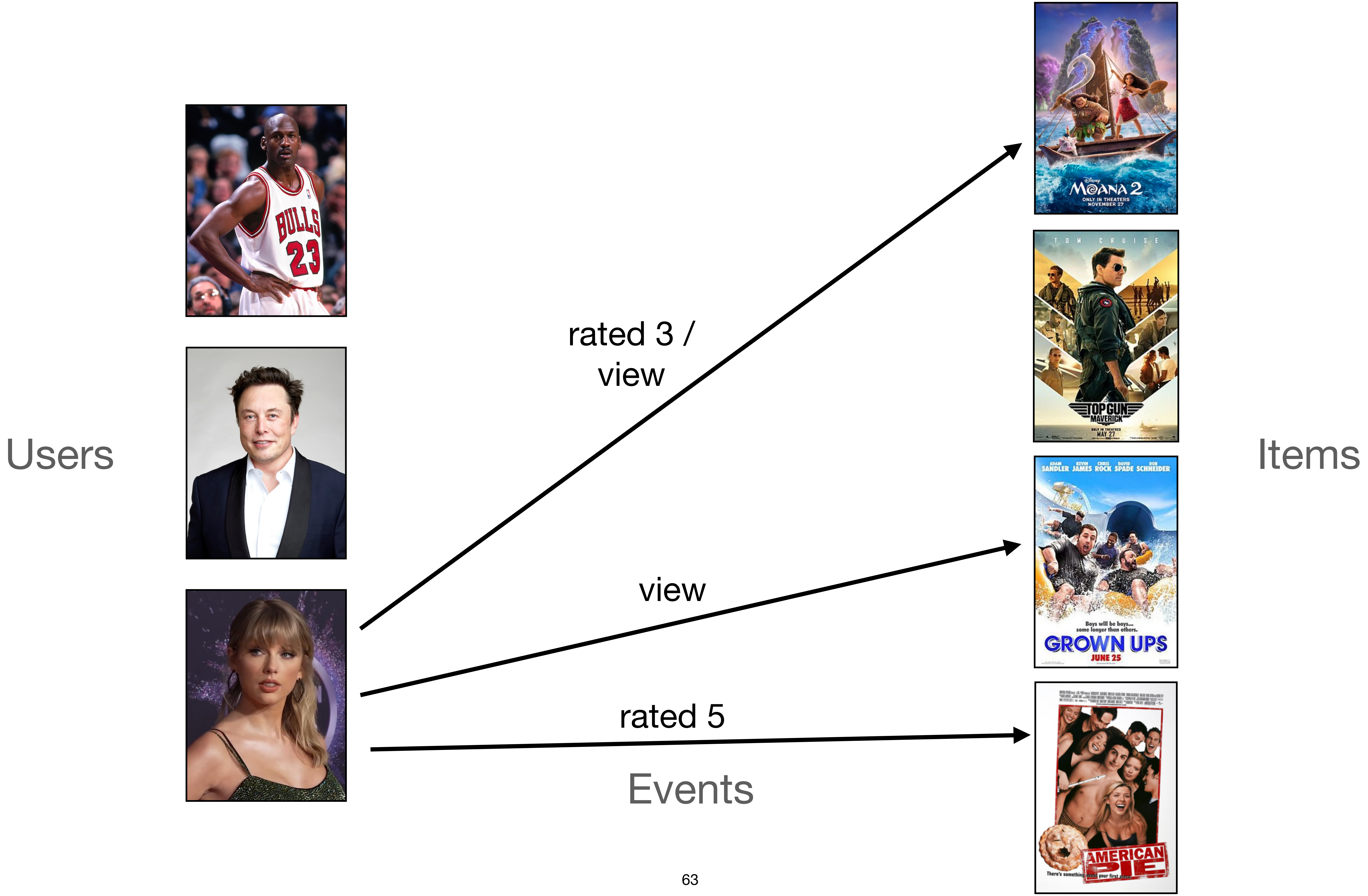
Items

Events



Users
view
view
Items
Events
BULLS
23
MOANA 2
ONLY IN THEATERS
NOVEMBER 27
TOM CRUISE
TOP GUN
MAVERICK
ADAM SANDLER KEVIN JAMES CHRIS ROCK DAVID SPADE ROB SCHNEIDER
Boys will be boys...
some longer than others.
GROWN UPS
JUNE 25
AMERICAN PIE

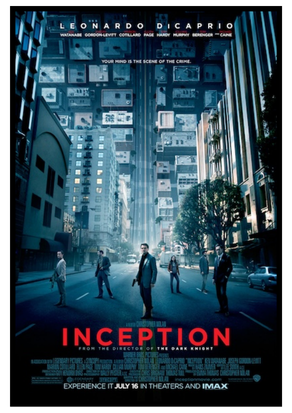
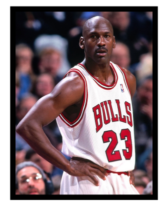
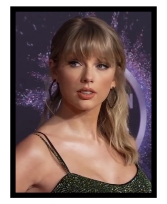

Users
download
view
Items
Events
MOANA 2
ONLY IN THEATERS
NOVEMBER 27
T O M C R U I S E
TOP GUN
MAVERICK
MAY 27
ADAM SANDLER KEVIN JAMES CHRIS ROCK DAVID SPADE ROB SCHNEIDER
Boys will be boys...
some longer than others.
GROWN UPS
JUNE 25
AMERICAN PIE


Users
Items
rated 3 / view
view
rated 5
Events
BULLS 23
MOANA 2
ONLY IN THEATERS NOVEMBER 27
TOM CRUISE
TOP GUN MAVERICK
ADAM SANDLER KEVIN JAMES CHRIS ROCK DAVID SPADE ROB SCHNEIDER
Boys will be boys... some longer than others.
GROWN UPS
JUNE 25
AMERICAN PIE

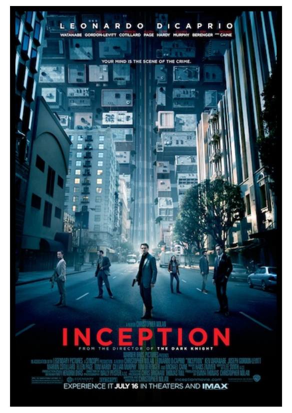
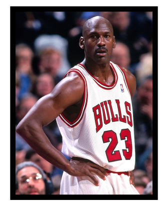
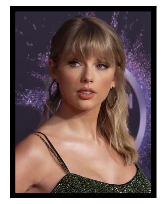
User-Item Matrix

	51235	2312	5215	232	…	987233	4124
3234	X		X				
41232		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
99283	X		X			X	

User-Item Matrix

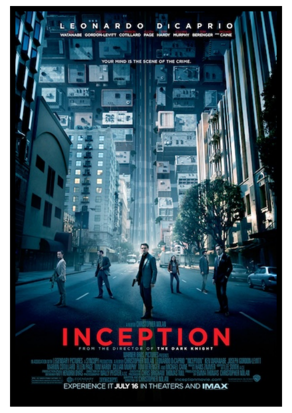
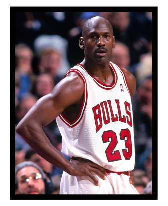
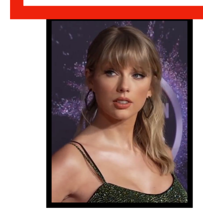
					…		
	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

User-Item Matrix

					…		
	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

Represents a single event - lets assume this is "VIEW"
So Jordan viewed Moana 2 and Grown Ups

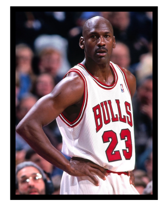
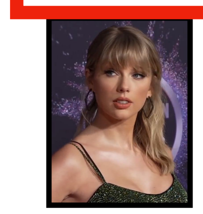
User-Item Matrix

					…		
	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

Represents a single event - lets assume this is "VIEW"
So Jordan viewed Moana 2 and Grown Ups

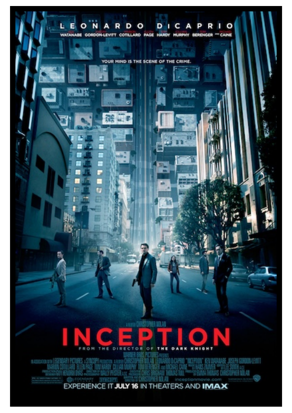
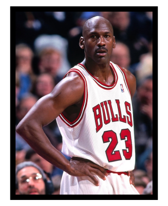
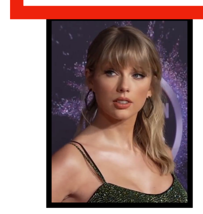
User-Item Matrix

To predict what Jordan will like to view, we will look for similar vectors (users)

					...		
	X		X				
		X		X		X	
...			X			X	
...				X			
...							
...		X				X	
	X		X			X	

User-Item Matrix

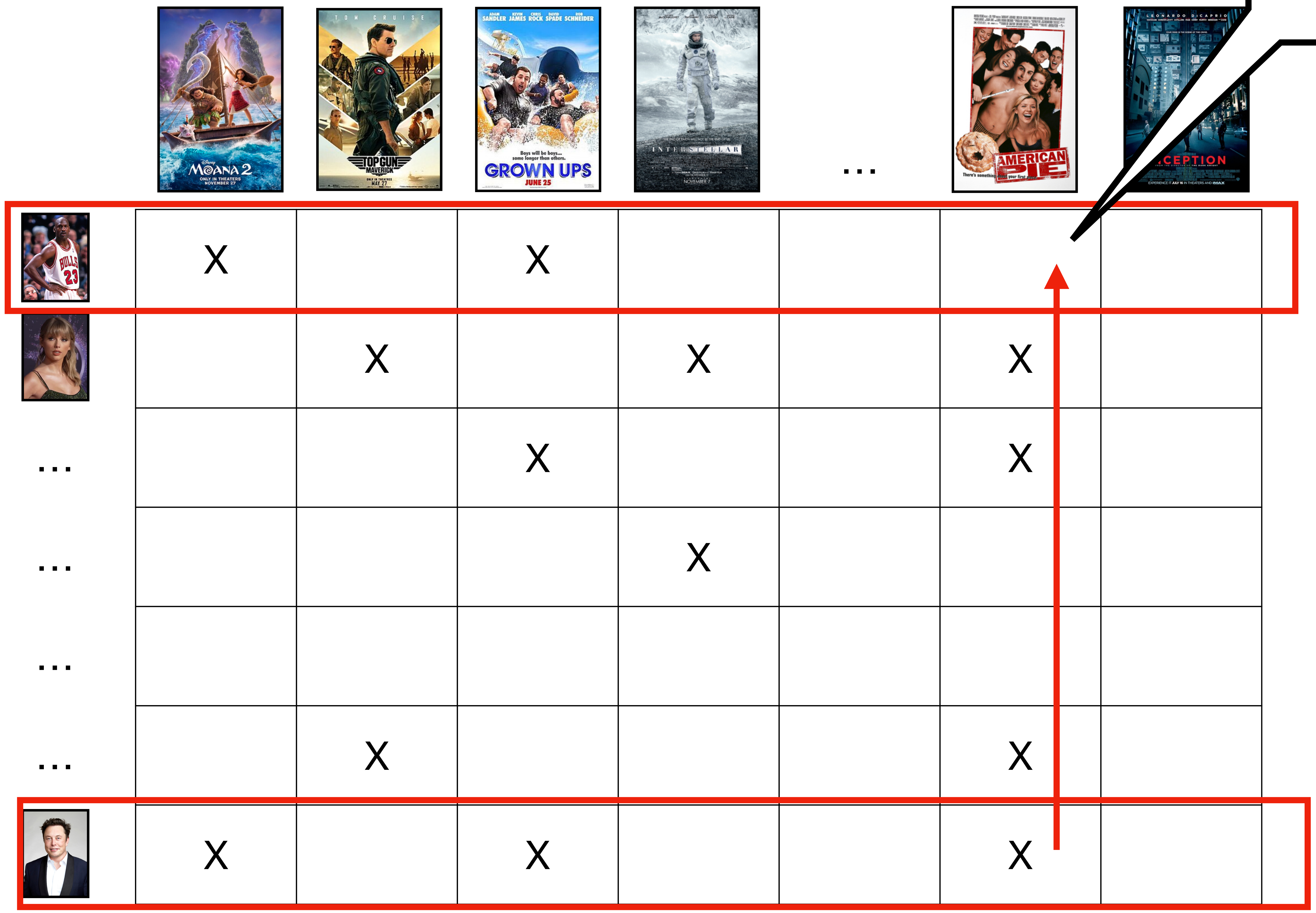
To predict what Jordan will like to view, we will look for similar vectors (users)

					…		
	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

User-Item Matrix

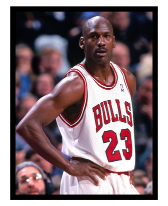
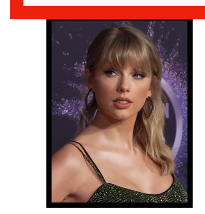
To predict what Jordan will like to view, we will look for similar vectors (users)

Recommend American Pie



User-Item Matrix

To predict what Jordan will like to view, we will look for similar vectors (users)

					…		
	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

Recommend American Pie

This algorithm is "User Based CF"

User-based CF

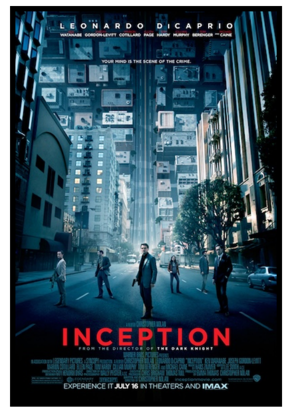
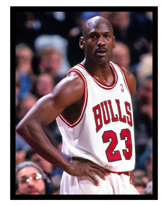
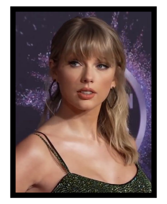
The preferences of similar users (neighbors) can be used to predict the preference for an item

Steps

- Similarity calculation
  calculates similarities with all other users (Cosine / Pearson / Jaccard …)
- Neighborhood formation
  identify a set of similar users (neighbors)
- Prediction
  take a weighted average of the preference by the neighbors

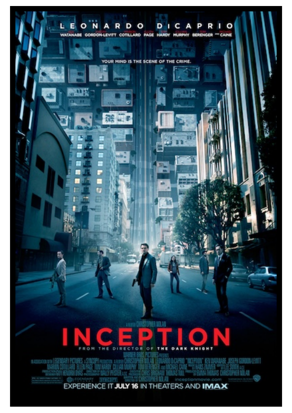
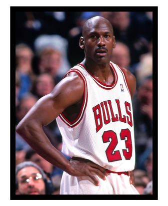
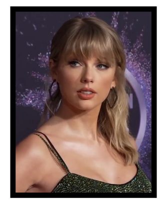
$$\hat{v}_{u,i} = \frac{\sum_{v \in N(u)} s(u,v) \times v_{v,i}}{\sum_{v \in N(u)} s(u,v)}$$

User-Item Matrix

					…		
	X		X				
		X		X		X	
…			X			X	
…	X			X			
…	X			X			
…		X				X	
	X		X	X		X	

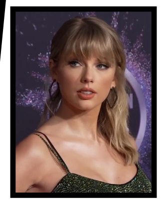
So what is “Item Based CF”?

User-Item Matrix

					…		
	X		X				
		X		X		X	
…			X			X	
…	X			X			
…	X			X			
…		X				X	
	X		X	X		X	

So what is
“Item Based CF”?

User-Item Matrix

					…		
	X		X				
		X		X		X	
…			X			X	
…	X			X			
…	X			X			
…		X				X	
	X		X	X		X	

Recommend Interstellar because you watched Moana 2

So what is "Item Based CF"?

User-Item Matrix

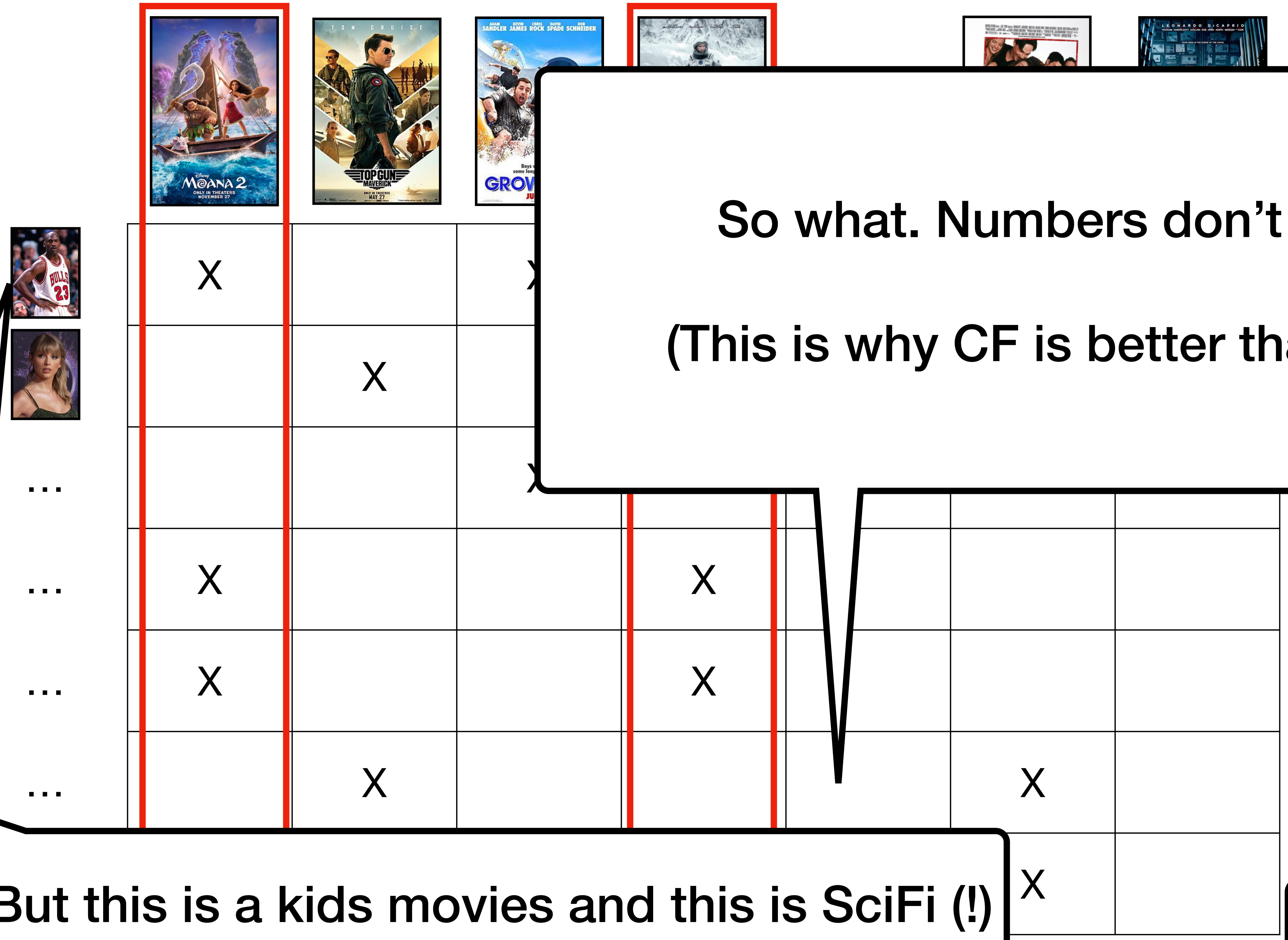
					…		
	X		X				
		X		X		X	
…			X			X	
…	X			X			
…	X			X			
…		X				X	
						X	

Recommend Interstellar because you watched Moana 2

But this is a kids movies and this is SciFi (!)

So what is “Item Based CF”?

User-Item Matrix



Recommend Interstellar because you watched Moana 2

So what. Numbers don't lie :)

(This is why CF is better than CB)

But this is a kids movies and this is SciFi (!)

So what is "Item Based CF"?

Item-based CF

If two items are “similar", then a user who liked one item will likely like the other

Steps

- Similarity calculation
  calculates similarities with all other items (Cosine / Person / Jaccard …)
- Neighborhood formation
  identify a set of similar items (neighbors)
- Prediction
  take a weighted average of the preference by the neighbors

$$\hat{v}_{u,i} = \frac{\sum_{j \in N(i)} s(i,j) \times v_{u,j}}{\sum_{j \in N(i)} s(i,j)}$$

Item-based CF

If two items are "similar", then a user who liked one item will likely like the other

Similarity is "different" for CB and CF

Steps

- Similarity calculation
  calculates similarities with all other items (Cosine / Person / Jaccard …)
- Neighborhood formation
  identify a set of similar items (neighbors)
- Prediction
  take a weighted average of the preference by the neighbors

$$\hat{v}_{u,i} = \frac{\sum_{j \in N(i)} s(i,j) \times v_{u,j}}{\sum_{j \in N(i)} s(i,j)}$$

User based vs Item-based

?

User based vs Item-based

TLDR;

- Item based is preferred most times

$$\hat{v}_{u,i} = \frac{\sum_{v \in N(u)} s(u,v) \times v_{v,i}}{\sum_{v \in N(u)} s(u,v)}$$

User based

$$\hat{v}_{u,i} = \frac{\sum_{j \in N(i)} s(i,j) \times v_{u,j}}{\sum_{j \in N(i)} s(i,j)}$$

Item based

User based vs Item-based

Aspect	User-Based CF	Item-Based CF
Similarity Calculation	Between users	Between items
Scalability	More computationally intensive with many users	Typically more scalable due to fewer items
Stability	Can be volatile as user preferences change	Generally more stable since item properties change slowly
Data Sparsity	Can struggle when users have few interactions	Often more robust as items tend to accumulate more interactions
Recommendation Focus	Recommends based on similar users' behavior	Recommends items similar to those the user has engaged with

Collaborative Filtering

Recommend based on the behavior and preferences of other users

Memory based CF

- User-based CF
- Item-based CF

Model based CF

- Matrix factorization

Reminder from a few slides ago…

Matrix Factorization

One of the most popular techniques for CF

- A technique to discover latent factors that explain user-item interactions
  latent factors ~ hidden dimension ~ "embeddings"

- Decomposes the large, sparse user-item matrix into **two** (or three) lower-dimensional matrices

Matrix Factorization

					…		
	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

Original matrix
R

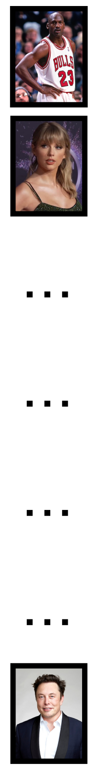
Matrix Factorization

	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

Original matrix
R

Intuition
We want to create 2 (or 3) matrixes that if we multiply them we will get the original matrix

Matrix Factorization

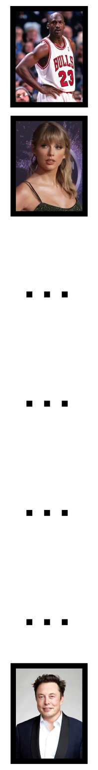


				...		
X		X				
	X		X		X	
		X			X	
			X			
	X				X	
X		X			X	

Rows: (user), (user), ..., ..., ..., ..., (user)

Original matrix
R

Latent space



	0.32	0.69	0.88	0.32
	0.84	0.35	0.51	0.32
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
	0.25	0.32	0.44	0.24

User features matrix
U

*

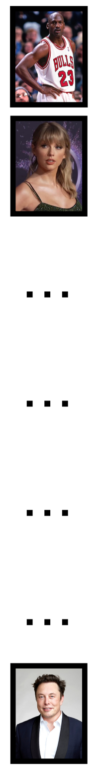
Latent space

				...		
0.23	0.12	0.3	0.66	...	0.09	0.23
0.43	0.33	0.12	0.24	...	0.23	0.42
0.51	0.83	0.124	0.14	...	0.34	0.51
0.98	0.15	0.82	0.92	...	0.53	0.23

Item features matrix
V

Matrix Factorization

How can we get this value?

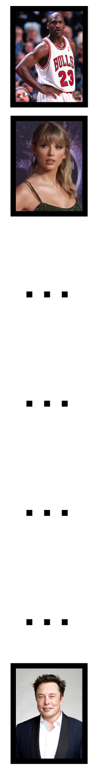


				...		
X		X				
	X		X		X	
		X			X	
			X			
	X				X	
X		X			X	

Rows 3–6 are labelled "..." (rows 1, 2 and 7 are labelled with photos).

Original matrix
R

Latent space



0.32	0.69	0.88	0.32
0.84	0.35	0.51	0.32
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
0.25	0.32	0.44	0.24

Rows 3–6 are labelled "…" (rows 1, 2 and 7 are labelled with photos).

User features matrix
U

*

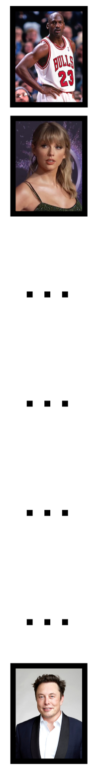
Latent space

0.23	0.12	0.3	0.66	…	0.09	0.23
0.43	0.33	0.12	0.24	…	0.23	0.42
0.51	0.83	0.124	0.14	…	0.34	0.51
0.98	0.15	0.82	0.92	…	0.53	0.23

Item features matrix
V

Matrix Factorization

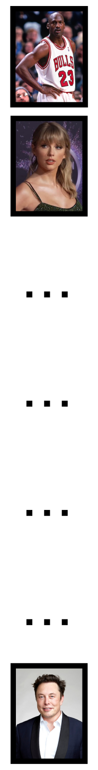
How can we get this value?



				...		
X		X				
	X		X		X	
		X			X	
			X			
	X				X	
X		X			X	

(rows 3–6 labelled ...)

Original matrix
R



Latent space

	0.32	0.69	0.88	0.32
	0.84	0.35	0.51	0.32
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
	0.25	0.32	0.44	0.24

User features matrix
U

*

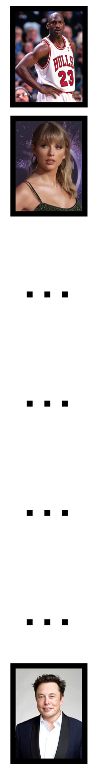
Latent space

				...		
0.23	0.12	0.3	0.66	...	0.09	0.23
0.43	0.33	0.12	0.24	...	0.23	0.42
0.51	0.83	0.124	0.14	...	0.34	0.51
0.98	0.15	0.82	0.92	...	0.53	0.23

Item features matrix
V

Matrix Factorization - prediction

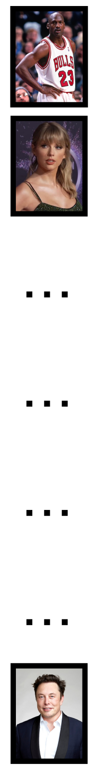
And how to can we get this value?



					...		
	X		X				
		X		X		X	
...			X			X	
...				X			
...							
...		X				X	
	X		X			X	

Original matrix
R

Latent space



	0.32	0.69	0.88	0.32
	0.84	0.35	0.51	0.32
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
	0.25	0.32	0.44	0.24

User features matrix
U

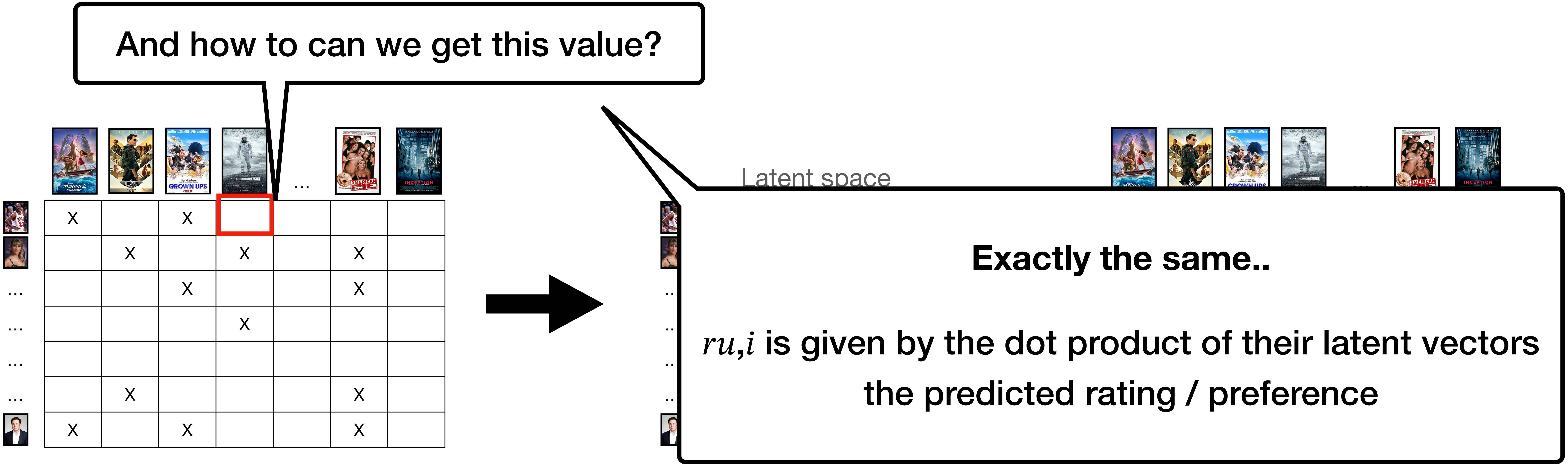
*

Latent space

				...		
0.23	0.12	0.3	0.66	...	0.09	0.23
0.43	0.33	0.12	0.24	...	0.23	0.42
0.51	0.83	0.124	0.14	...	0.34	0.51
0.98	0.15	0.82	0.92	...	0.53	0.23

Item features matrix
V

Matrix Factorization - prediction



Original matrix
R

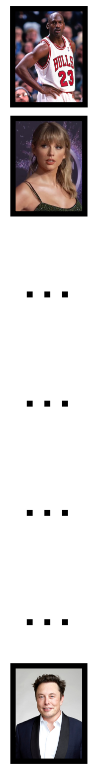
User features matrix
U

Item features matrix
V

Matrix Factorization - prediction

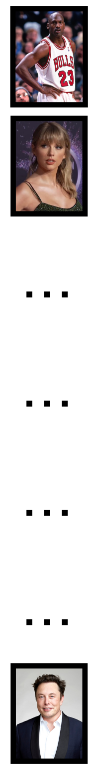
And how to can we get this value?

$$\hat{r}_{ui} = \mathbf{u} \cdot \mathbf{v}$$



X		X				
	X		X		X	
		X			X	
			X			
	X				X	
X		X			X	

Original matrix
R



Latent space

0.32	0.69	0.88	0.32
0.84	0.35	0.51	0.32
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
0.25	0.32	0.44	0.24

User features matrix
U

*

Latent space

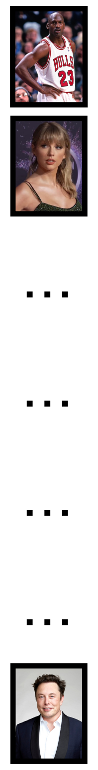
0.23	0.12	0.3	0.66	…	0.09	0.23
0.43	0.33	0.12	0.24	…	0.23	0.42
0.51	0.83	0.124	0.14	…	0.34	0.51
0.98	0.15	0.82	0.92	…	0.53	0.23

Item features matrix
V

Matrix Factorization - prediction

And how to can we get this value?

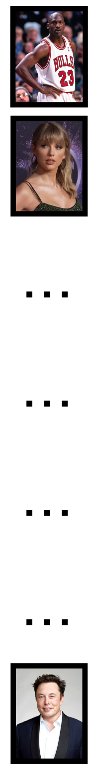
					...		
	X		X				
		X		X		X	
...			X			X	
...				X			
...							
...		X				X	
	X		X			X	



Original matrix
R

Latent space

	0.32	0.69	0.88	0.32
	0.84	0.35	0.51	0.32
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...
	0.25	0.32	0.44	0.24



User features matrix
U

*

$$\hat{r}_{ui} = \mathbf{u} \cdot \mathbf{v}$$

Including bias values
(A popular technique from the Netflix prize)

$$\hat{r}_{ui} = \mu + b_u + b_i + \mathbf{u} \cdot \mathbf{v}$$

Latent space

0.						
0.						
0.51	0.83	0.124	0.14	...	0.34	0.51
0.98	0.15	0.82	0.92	...	0.53	0.23

Item features matrix
V

Matrix Factorization - techniques

- **SVD**
  Full decomposition using orthogonal matrices and singular values
- **Funk SVD**
  An optimization approach using gradient descent
- **PMF**
  A probabilistic approach to factorization
- **NMF**
  Ensures all elements are non-negative for interpretability
- **ALS**
  Alternates solving for user and item matrices using least squares

Agenda

- Intro and Intuition
- Content Based
- Collaborative Filtering
- **Common challenges**

Implicit vs Explicit data

- If I asked you to **explicitly** provide 3 movies or artists you liked what would you say?

Implicit vs Explicit data

- If I asked you to **explicitly** provide 3 movies or artists you liked what would you say?

  is it “cool” to say Moana 2 or Taylor Swift if you are a grown man?

- You can think of a similar example for Woman :)

- **Implicit** capture your TRUE preference

  —> always works better :)

Implicit vs Explicit data

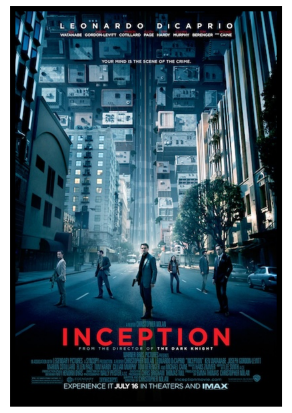
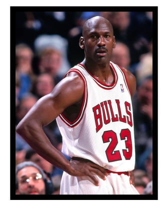
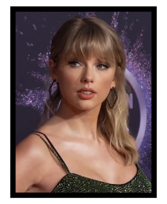
- If I asked you to **explici** sts
  you liked what would yo
  is it "cool" to say Moana 2 or T

Click on something, view 50% of video, search, share, copy link, like…

- You can think of a similar examp e for Woman :)

- **Implicit** capture your TRUE preference
  —> always works better :)

How much values are available?

					…		
	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

How much values are available?

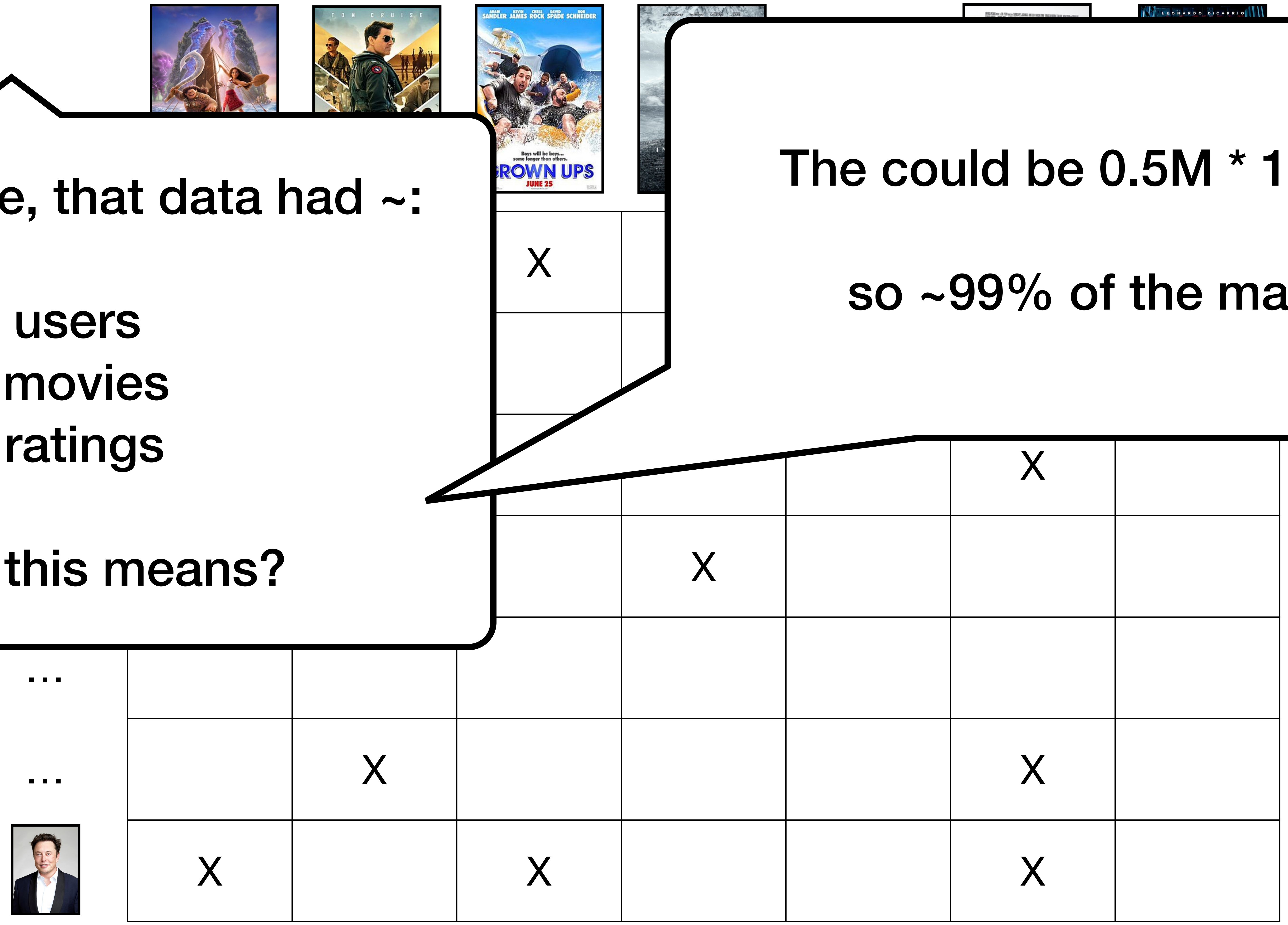
In the Netflix prize, that data had ~:

0.5M users
17.7K movies
100M ratings

What does this means?

					...		
			X				
				X		X	
			X			X	
				X			
...							
...		X				X	
	X		X			X	

How much values are available?



In the Netflix prize, that data had ~:

0.5M users
17.7K movies
100M ratings

What does this means?

The could be 0.5M * 17.7K = ~9B values

so ~99% of the matrix is unknown

How much values are available?

In the Netflix prize, that data had ~:

0.5M users
17.7K movies
100M ratings

What does this means?

The could be 0.5M * 17.7K = ~9B values

so ~99% of the matrix is unknown

Recommeder system data sets are:
- Big Data (endless implicit events)
- BUT the data is SPARSE

How much values are available?

In the N

100M ratings

What does this means?

= ~9B values

unknown

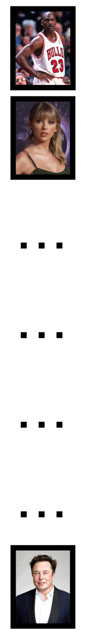
Traditional ML algorithms does NOT work on sparse data.

—> this is why Recommender Systems are unique

Recommeder system data sets are:
- Big Data (endless implicit events)
- BUT the data is SPARSE

Multi event data matrix

- Can a matrix represent multi event data?



					...		
	X		X				
		X		X		X	
...			X			X	
...				X			
...							
...		X				X	
	X		X			X	

VIEW

Multi event data matrix

- Can a matrix represent multi event data?

					…		
	X		X				
		X		X		X	
…			X			X	
…				X			
…							
…		X				X	
	X		X			X	

VIEW

					…		
						X	X
	X	X		X			X
…			X				
…	X		X	X			
…					X		
…		X					
			X	X		X	X

DOWNLOAD

Multi event data matrix

- Can a matrix represent multi event data?

					...		
	X		X				
		X		X		X	
...			X			X	
...				X			
...							
...		X				X	
	X		X			X	

VIEW

					...		
						X	X
	X	X		X			X
...			X				
...	X		X	X			
...					X		
...		X					
			X	X		X	X

DOWNLOAD

					...		
			1				1
	3	5					
...			2				
...					5		
...				4			
...		2					
		4		3			5

RATING (1-5)

Multi event data matrix

There are techniques to “blend” or “merge” - for example, add 1 for “view” 2 for “share”, -1 for “dislike”…

data?

					...		
	X		X				
				X		X	
...			X			X	
...				X			
...							
...		X				X	
	X		X			X	

VIEW

					...		
						X	X
	X	X		X			X
...			X				
...	X		X	X			
...					X		
...		X					
			X	X		X	X

DOWNLOAD

					...		
			1				1
	3	5					
...			2				
...					5		
...				4			
...		2					
		4		3			5

RATING (1-5)

Multi event data matrix

There are techniques to “blend” or “merge” - for example, add 1 for “view” 2 for “share”, -1 for “dislike”…

data?

	X		X				
				X		X	
…			X			X	
…				X			
…							
…		X					
	X		X				

VI

KISS

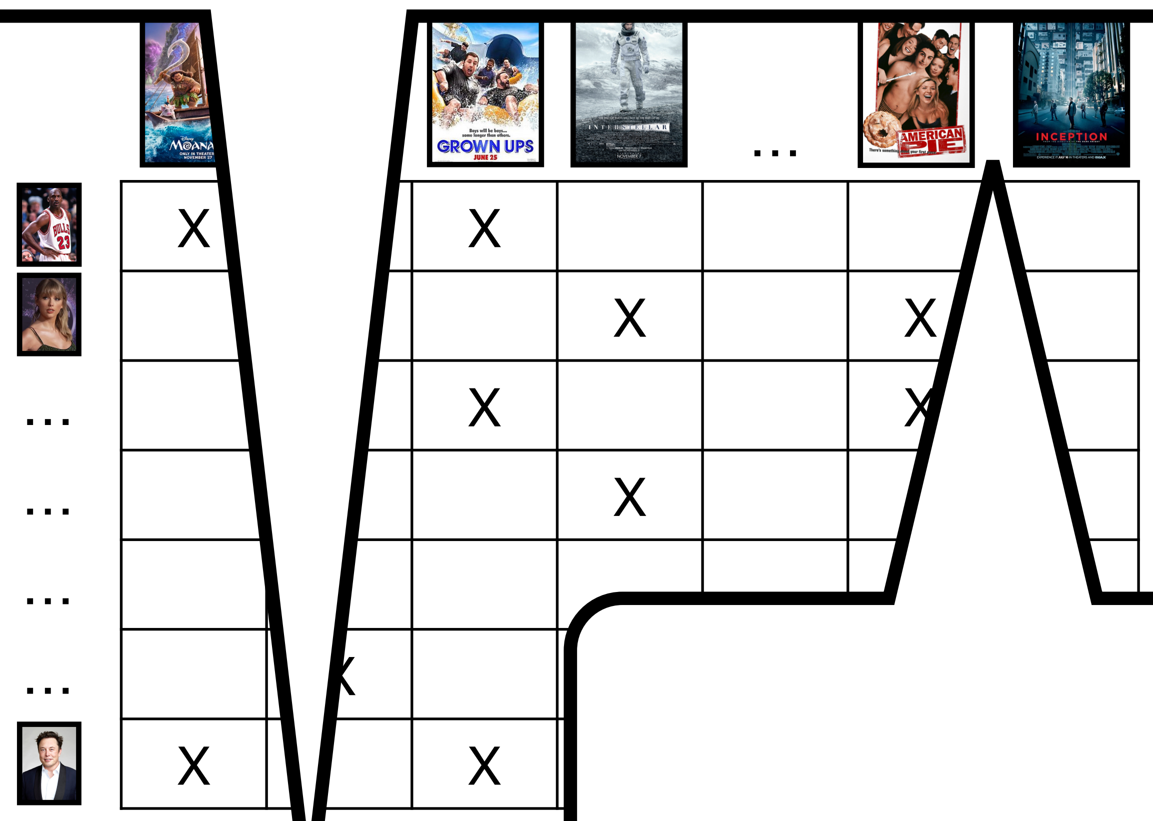
			1				1
	3	5					
…			2				
…					5		
				4			
				3			5

RATING (1-5)

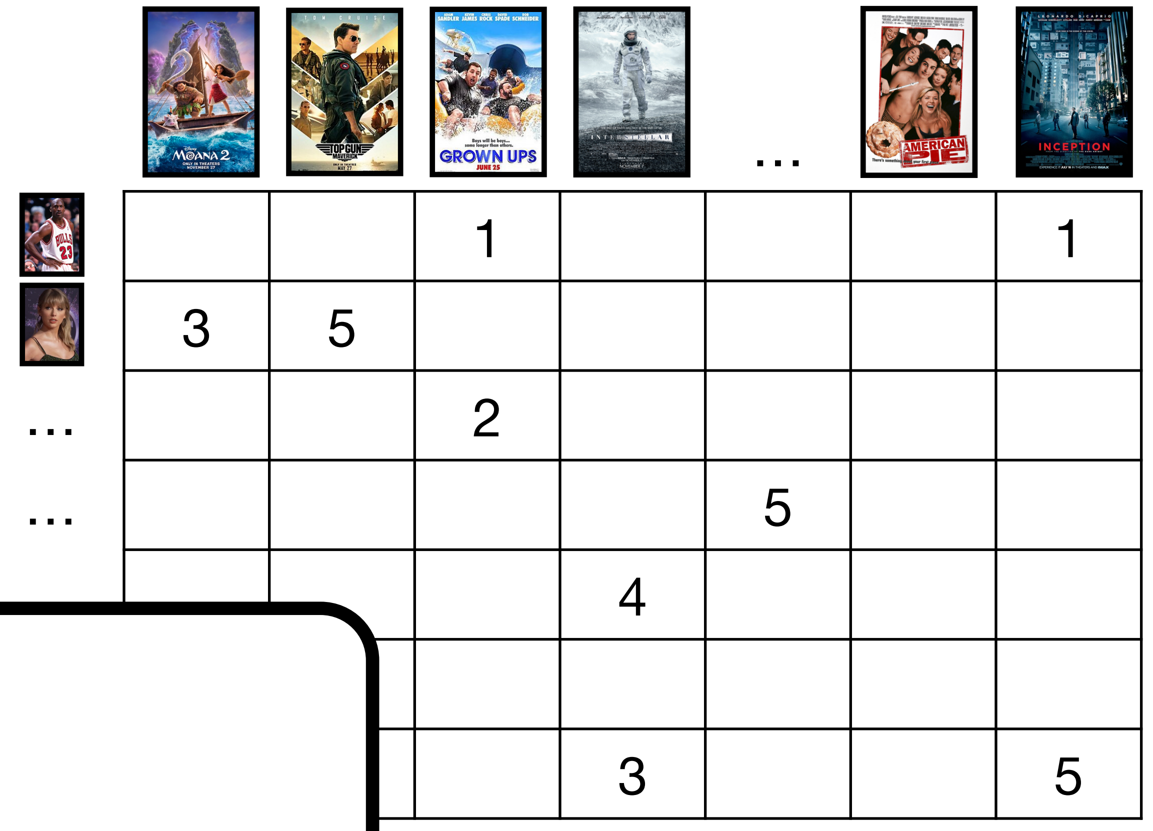
Multi event data matrix

There are techniques to “blend” or “merge” - for example, add 1 for “view” 2 for “share”, -1 for “dislike”…

data?



VI



RATING (1-5)

KISS
Keep It Simple, Stupid!
https://en.wikipedia.org/wiki/KISS_principle

—> use a single event, at least the first time…

Coldstart

Ok - implicit data is better.
So what to do when a "new" user enters the system?

Coldstart

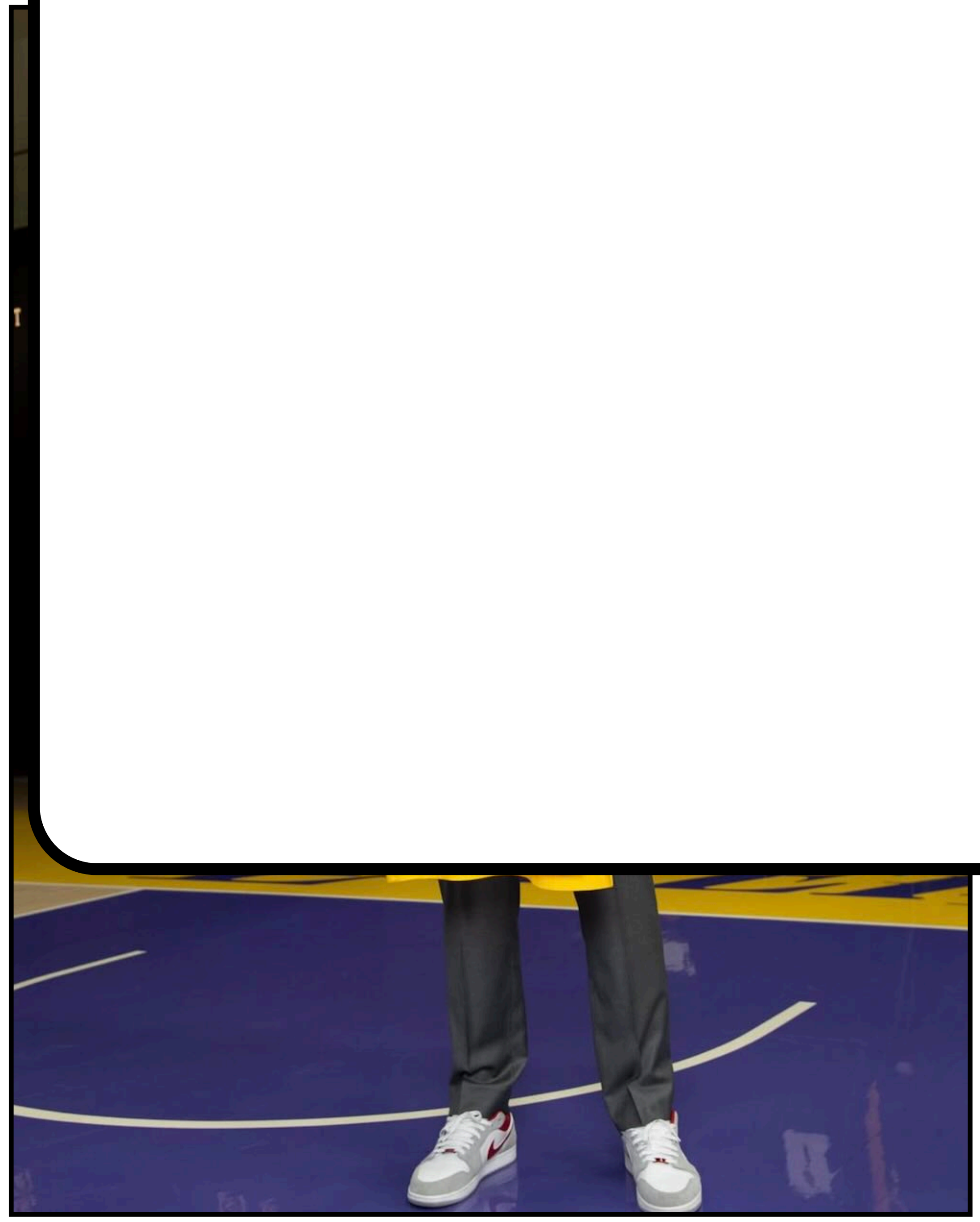
Ok - implicit data is better.
So what to do when a “new” user enters the system?

For CF, not too much by definition…

- Show average
- Use semantics if available (location, gender, age..)
- Ask for explicit data
- Build a “real time” system that after the first event kicks in

Coldstart

Can you have a “Coldstart” problem for items?



- Build a “real tin system that after the first event kicks in

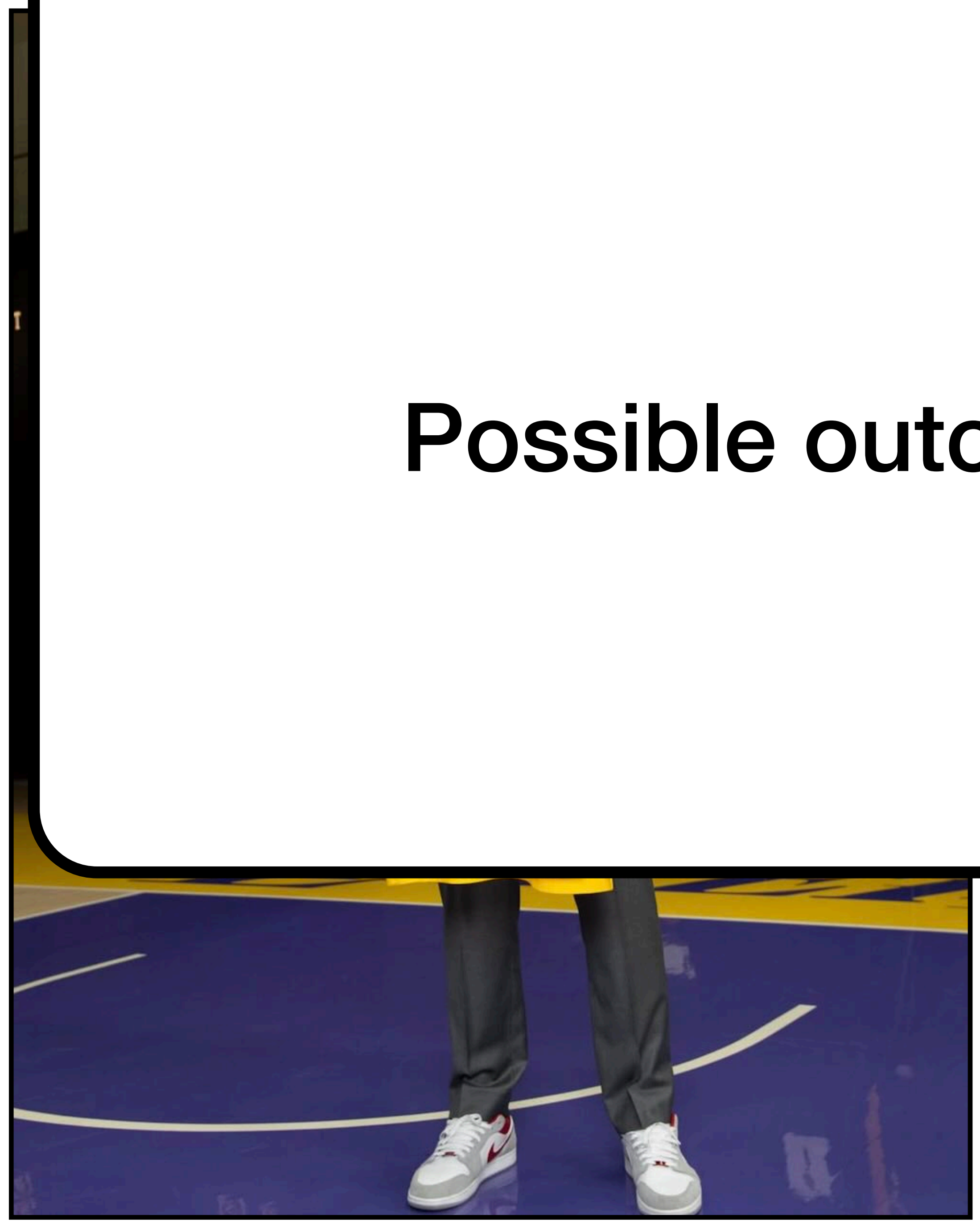
Coldstart

Can you have a “Coldstart” problem for items?

YES - IT IS EVEN A BIGGER PROBLEM!

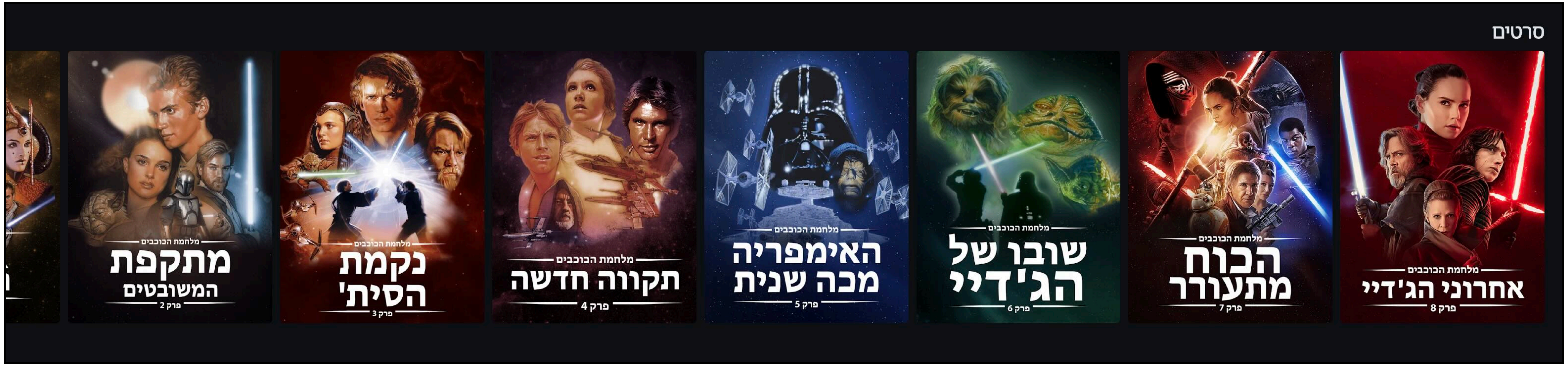
(Because you need to “retrain” your ML model)

Possible outcome: the new item would not get recommender to any user

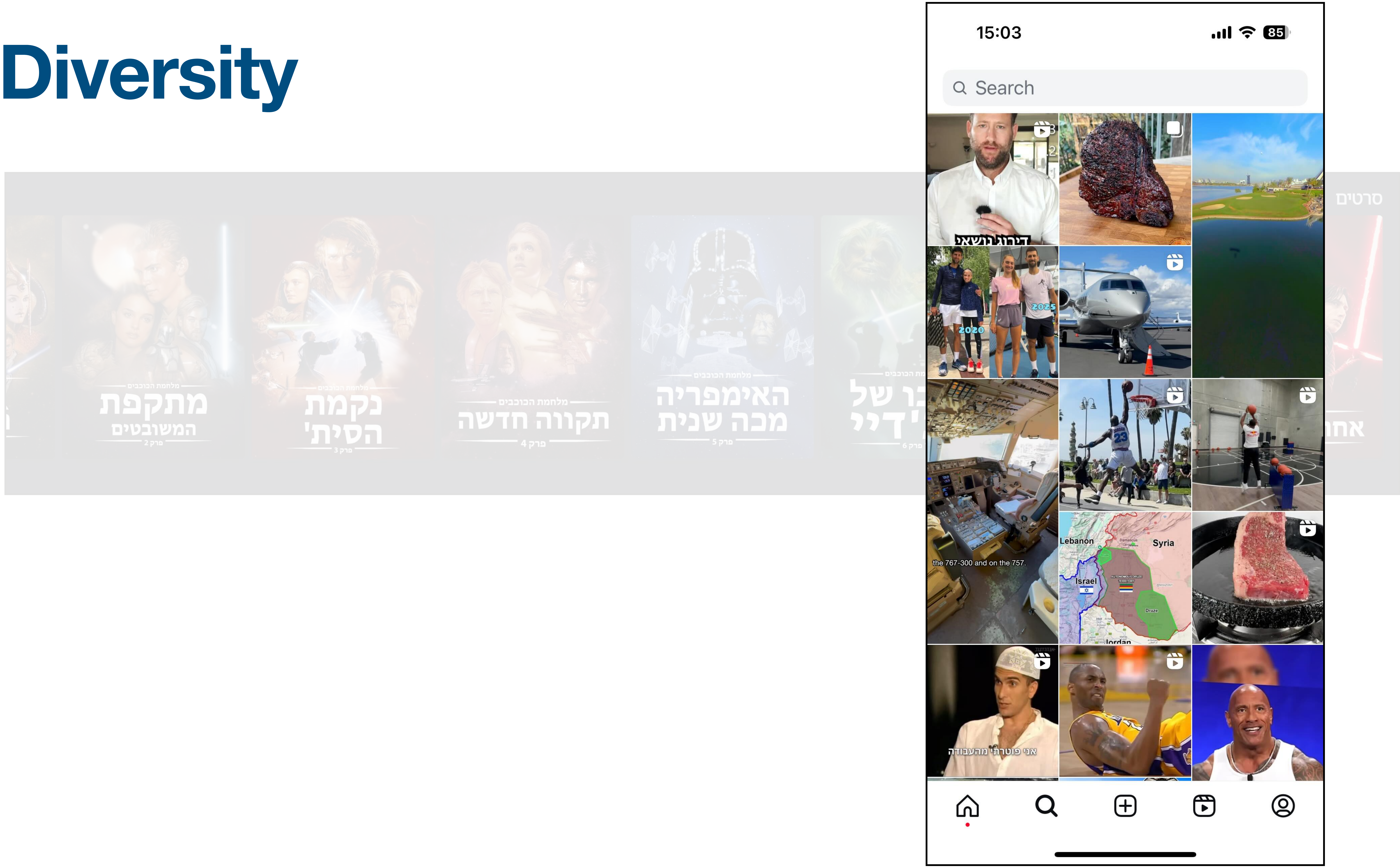


- Build a “real tin system that after the first event kicks in

Diversity

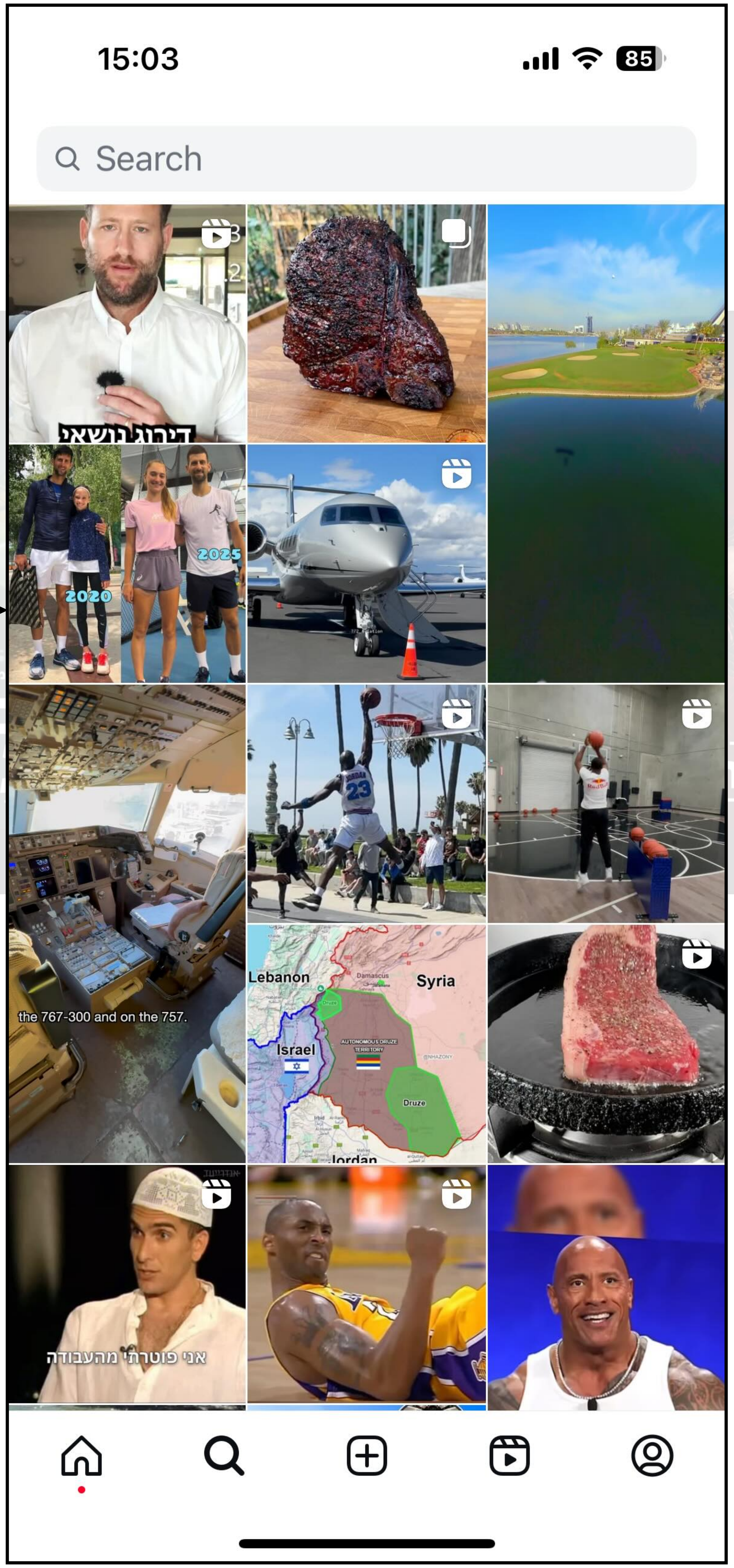

סרטים
מלחמת הכוכבים
אחרוני הג'דיי
פרק 8
מלחמת הכוכבים
הכוח מתעורר
פרק 7
מלחמת הכוכבים
שובו של הג'דיי
פרק 6
מלחמת הכוכבים
האימפריה מכה שנית
פרק 5
מלחמת הכוכבים
תקווה חדשה
פרק 4
מלחמת הכוכבים
נקמת הסית'
פרק 3
מלחמת הכוכבים
מתקפת המשובטים
פרק 2

Diversity


מתקפת המשובטים
נקמת הסית'
תקווה חדשה
האימפריה מכה שנית
סרטים
15:03
Search
the 767-300 and on the 757.
Lebanon
Syria
Israel

Diversity

How can you implement diversity?



Diversity

How can you implement diversity?

“Easier” with metadata / semantics / CB…
So if you have it, use it