IA Frameworks Introduction to Recommender systems

Objectives

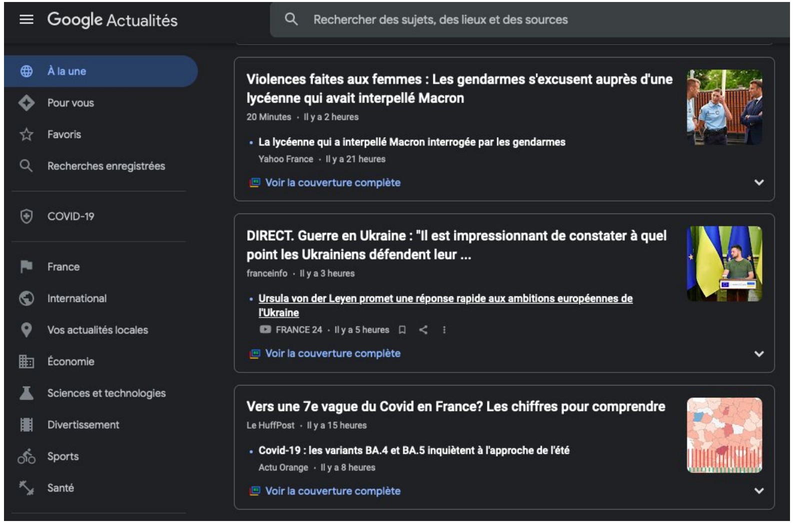
- Help users to match with the best items
- Ease information overload

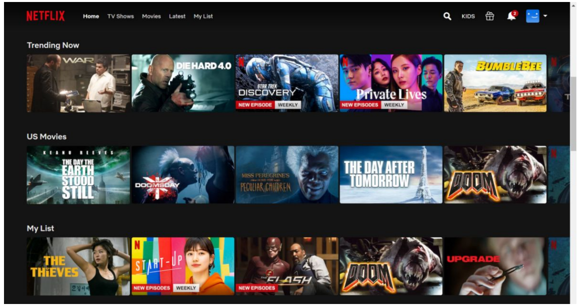
Recommender systems:

- Netflix => 2/3 of the movies watched are recommended

- Google =>news recommendations improved click-through rate (CTR) by 38%

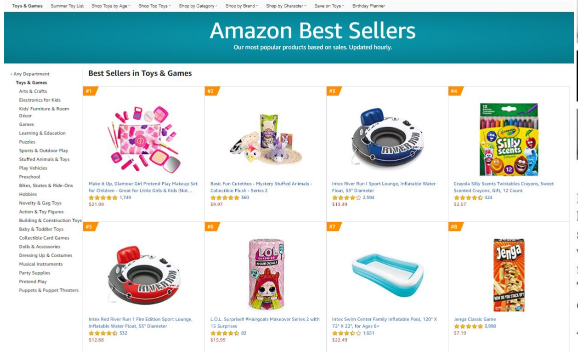
- Amazon => 35% of sales come from recommendations





Taxonomy

- Popularity
- Content based
- User based
- Item based
- Knowledge model

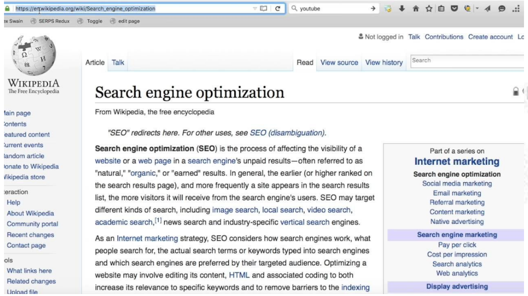


Content based

Goal: Find most similar items based on their characteristics

- General features
  (e.g. Movie: actors, director, movie type..., Product: price, category, color...)
- Text
- Image
- Sound

	id	imdb_id	original_title	director	production	genre	cast	budget	revenue	runtime	release_year	vote_count
0	135397	tt0369610	Jurassic World	Colin Trevorrow	Universal Studios	Action	Chris Pratt	150000000	1513528810	124	2015	5562
1	76341	tt1392190	Mad Max Fury Road	George Miller	Village Roadshow Pictures	Action	Tom Hardy	150000000	378436354	120	2015	6185
2	262500	tt2908446	Insurgent	Robert Schwentke	Summit Entertainment	Adventure	Shailene Woodley	110000000	295238201	119	2015	2480
3	140607	tt2488496	Star Wars The Force Awakens	JJ Abrams	Lucasfilm	Action	Harrison Ford	200000000	2068178225	136	2015	5292
4	168259	tt2820852	Furious	James Wan	Universal Pictures	Action	Vin Diesel	190000000	1506249360	137	2015	2947
5	281957	tt1663202	The Revenant	Alejandro Gonzlez Irritu	Regency Enterprises	Western	Leonardo DiCaprio	135000000	532950503	156	2015	3929
6	87101	tt1340138	Terminator Genisys	Alan Taylor	Paramount Pictures	Science Fiction	Arnold Schwarzenegger	155000000	440603537	125	2015	2598
7	286217	tt3659388	The Martian	Ridley Scott	Twentieth Century Fox Film Corporation	Drama	Matt Damon	108000000	595380321	141	2015	4572
8	211672	tt2293640	Minions	Kyle BaldaPierre Coffin	Universal Pictures	Family	Sandra Bullock	74000000	1156730962	91	2015	2893
9	150540	tt2096673	Inside Out	Pete Docter	Walt Disney Pictures	Comedy	Amy Poehler	175000000	853708609	94	2015	3935


https://en.wikipedia.org/wiki/Search_engine_optimization

WIKIPEDIA The Free Encyclopedia

Article Talk Read View source View history

Search engine optimization

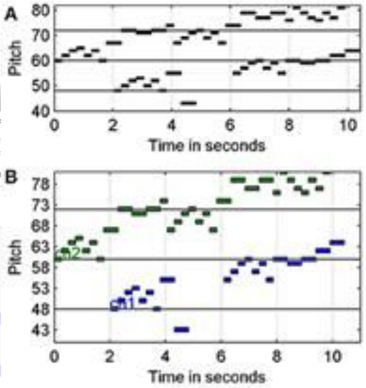
From Wikipedia, the free encyclopedia

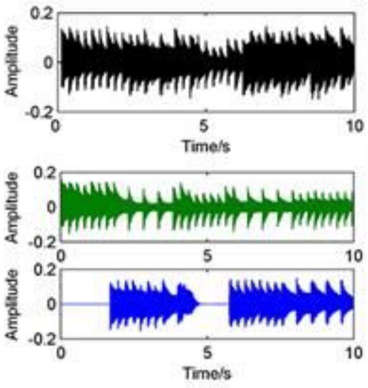
"SEO" redirects here. For other uses, see SEO (disambiguation).

Search engine optimization (**SEO**) is the process of affecting the visibility of a website or a web page in a search engine's unpaid results—often referred to as "natural," "organic," or "earned" results. In general, the earlier (or higher ranked on the search results page), and more frequently a site appears in the search results list, the more visitors it will receive from the search engine's users. SEO may target different kinds of search, including image search, local search, video search, academic search,[1] news search and industry-specific vertical search engines.

As an Internet marketing strategy, SEO considers how search engines work, what people search for, the actual search terms or keywords typed into search engines and which search engines are preferred by their targeted audience. Optimizing a website may involve editing its content, HTML and associated coding to both increase its relevance to specific keywords and to remove barriers to the indexing

Part of a series on **Internet marketing**: Search engine optimization, Social media marketing, Email marketing, Referral marketing, Content marketing, Native advertising; **Search engine marketing**: Pay per click, Cost per impression, Search analytics, Web analytics; **Display advertising**





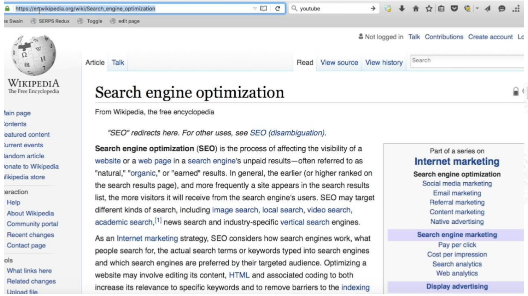
Content based

	id	imdb_id	original_title	director	production	genre	cast	budget	revenue	runtime	release_year	vote_count
0	135397	tt0369610	Jurassic World	Colin Trevorrow	Universal Studios	Action	Chris Pratt	150000000	1513528810	124	2015	5562
1	76341	tt1392190	Mad Max Fury Road	George Miller	Village Roadshow Pictures	Action	Tom Hardy	150000000	378436354	120	2015	6185
2	262500	tt2908446	Insurgent	Robert Schwentke	Summit Entertainment	Adventure	Shailene Woodley	110000000	295238201	119	2015	2480
3	140607	tt2488496	Star Wars The Force Awakens	JJ Abrams	Lucasfilm	Action	Harrison Ford	200000000	2068178225	136	2015	5292
4	168259	tt2820852	Furious	James Wan	Universal Pictures	Action	Vin Diesel	190000000	1506249360	137	2015	2947
5	281957	tt1663202	The Revenant	Alejandro Gonzlez Irritu	Regency Enterprises	Western	Leonardo DiCaprio	135000000	532950503	156	2015	3929
6	87101	tt1340138	Terminator Genisys	Alan Taylor	Paramount Pictures	Science Fiction	Arnold Schwarzenegger	155000000	440603537	125	2015	2598
7	286217	tt3659388	The Martian	Ridley Scott	Twentieth Century Fox Film Corporation	Drama	Matt Damon	108000000	595380321	141	2015	4572
8	211672	tt2293640	Minions	Kyle BaldaPierre Coffin	Universal Pictures	Family	Sandra Bullock	74000000	1156730962	91	2015	2893
9	150540	tt2096673	Inside Out	Pete Docter	Walt Disney Pictures	Comedy	Amy Poehler	175000000	853708609	94	2015	3935

Goal: Find most similar items based on their characteristics

How:

- Compute an embedding representation for all items
- Compute distance / similarity between embeddings



Search engine optimization

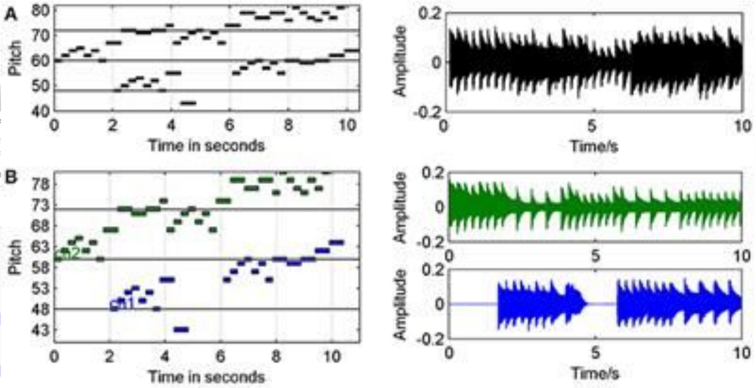
From Wikipedia, the free encyclopedia

"SEO" redirects here. For other uses, see SEO (disambiguation).

Search engine optimization (SEO) is the process of affecting the visibility of a website or a web page in a search engine's unpaid results—often referred to as "natural," "organic," or "earned" results. In general, the earlier (or higher ranked on the search results page), and more frequently a site appears in the search results list, the more visitors it will receive from the search engine's users. SEO may target different kinds of search, including image search, local search, video search, academic search,[1] news search and industry-specific vertical search engines.

As an Internet marketing strategy, SEO considers how search engines work, what people search for, the actual search terms or keywords typed into search engines and which search engines are preferred by their targeted audience. Optimizing a website may involve editing its content, HTML and associated coding to both increase its relevance to specific keywords and to remove barriers to the indexing

Part of a series on
Internet marketing
Search engine optimization
Social media marketing
Email marketing
Referral marketing
Content marketing
Native advertising
Search engine marketing
Pay per click
Cost per impression
Search analytics
Web analytics
Display advertising



Similarity

Similarity measure $s : E \times E \rightarrow \mathbb{R}$

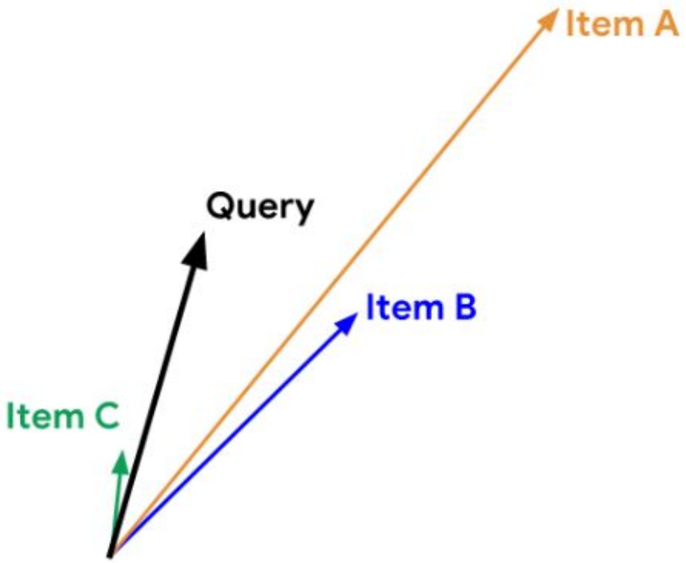
Given a query $q \in E$ the system looks for item embeddings $x \in E$ with high similarity $s(q, x)$

- Cosine similarity $s(q, x) = cos(q, x)$
- Dot Product $s(q, x) = \langle q, x \rangle = \sum_{i=1}^{d} q_i x_i$

$$= \|x\| \|q\| \cos(q, x)$$

- Dot Product $s(q, x) = \|q - x\| = \left[\sum_{i=1}^{d} (q_i - x_i)^2\right]^{\frac{1}{2}}$

Similarity



Dot-Product

Query : Item A > Item B > Item C

Cosine

Query : Item C > Item A > Item B

(-) Euclidean Distance

Query : Item B > Item C > Item A

Source: https://developers.google.com/machine-learning/recommendation

Content based

Advantages:

- No need of user interactions
- Easy to scale to large number of users

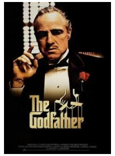
Drawbacks:

- Embeddings are often handcrafted and require expert knowledge

Collaborative filtering

- Users x items interactions
- Automatic learning of embeddings
- User based
- Item based
- Matrix factorization

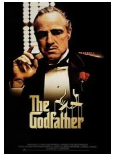
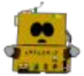
Collaborative filtering

					i 
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	4
	2	2	2	3	2
	5	5	1	1	1
	4	2	4	3	4
	3	1	4	3	3

User based

Principle:

- Find most similar users
- Estimate rating by the weighted average of similar users

					i 
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	4
	2	2	2	3	2
	5	5	1	1	1
	4	2	4	3	4
	3	1	4	3	3

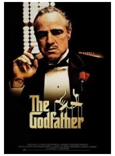
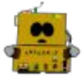
User based

Similar users:

- k-nearest neighbors
- Pearson correlation
- Cosine similarity

Estimate rating $r_{u,i}$ **:**

- Neighborhood U
- Similarity measure $sim(u, u')$

					i 
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	4
	2	2	2	3	2
	5	5	1	1	1
	4	2	4	3	4
	3	1	4	3	3

User based

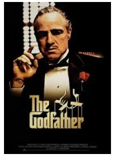
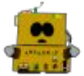
Similar users:

- k-nearest neighbors
- Pearson correlation
- Cosine similarity

Estimate rating $r_{u,i}$:

- Neighborhood U
- Similarity measure $sim(u, u')$

$$r_{u,i} = \frac{1}{N} \sum_{u' \in U} r_{u',i}$$

					i 
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	4
	2	2	2	3	2
	5	5	1	1	1
	4	2	4	3	4
	3	1	4	3	3

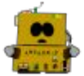
User based

Similar users:

- k-nearest neighbors
- Pearson correlation
- Cosine similarity

Estimate rating $r_{u,i}$:

- Neighborhood U
- Similarity measure $sim(u, u')$

					i
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	4
	2	2	2	3	2
	5	5	1	1	1
	4	2	4	3	4
	3	1	4	3	3

$$r_{u,i} = k \sum_{u' \in U} \operatorname{sim}(u, u')\, r_{u',i} \quad \text{with} \quad k = 1 / \sum_{u' \in U} |\operatorname{sim}(u, u')|$$

User based

Similar users:

- k-nearest neighbors
- Pearson correlation
- Cosine similarity

Estimate rating $r_{u,i}$:

- Neighborhood U
- Similarity measure $sim(u, u')$

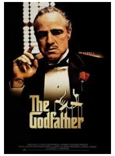
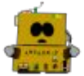
					i
u	4	1	4	3	$r_{u,i}$
	1	5	5	4	4
	2	2	2	3	2
	5	5	1	1	1
	4	2	4	3	4
	3	1	4	3	3

$r_{u,i} = \overline{r_u} + k \sum_{u' \in U} \mathrm{sim}\,(u, u')\left(r_{u',i} - \bar{r_{u'}}\right)$ with $\overline{r_u}$ the average rating of user u

User based

Drawbacks:

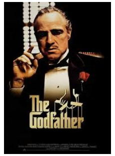
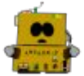
- How to handle new users?
- Does not scale to large real-world scenarios
- |Users| >> |Items|

					i
					
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	4
	2	2	2	3	2
	5	5	1	1	1
	4	2	4	3	4
	3	1	4	3	3

Item based

Principle:

- Use similarity between items
- Cosine similarity on users who have rated both items
- Adjusted cosine similarity: subtract average user ratings before computing cosine similarity

					i 
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	2
	2	2	2	3	2
	5	5	1	1	4
	4	2	4	3	4
	3	1	4	3	5

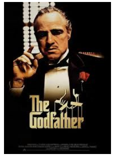
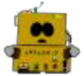
Item based

Similar items:

- k-nearest neighbors

Estimate rating :

- Neighborhood I
- Similarity measure $sim(i, i')$

					i 
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	2
	2	2	2	3	2
	5	5	1	1	4
	4	2	4	3	4
	3	1	4	3	5

$r_{u,i} = k \sum_{i' \in I} \text{sim}\left(i, i'\right) r_{u,i}$ with $k = 1 / \sum_{i' \in I} \left|\text{sim}\left(i, i'\right)\right|$

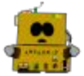
Item based

Advantages:

- Supposed to be more stable
- Pre-compute pairwise similarities
- Easier to scale

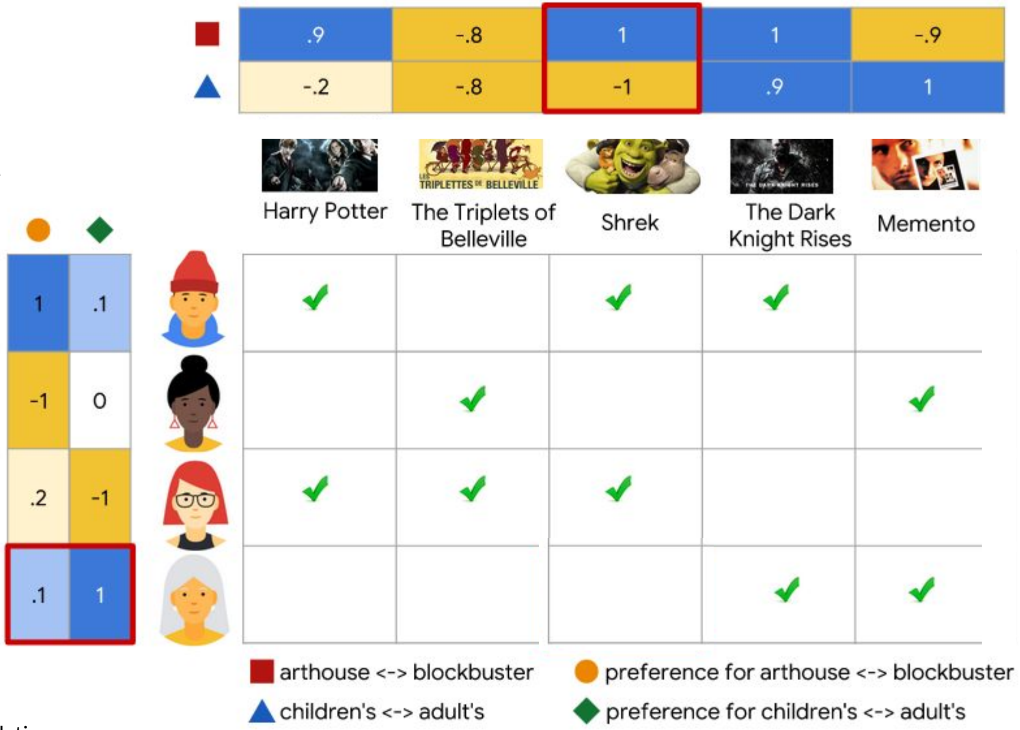
Drawbacks:

- New items
- Items with few interactions

					i
 u	4	1	4	3	$r_{u,i}$
	1	5	5	4	2
	2	2	2	3	2
	5	5	1	1	4
	4	2	4	3	4
	3	1	4	3	5

Matrix factorization

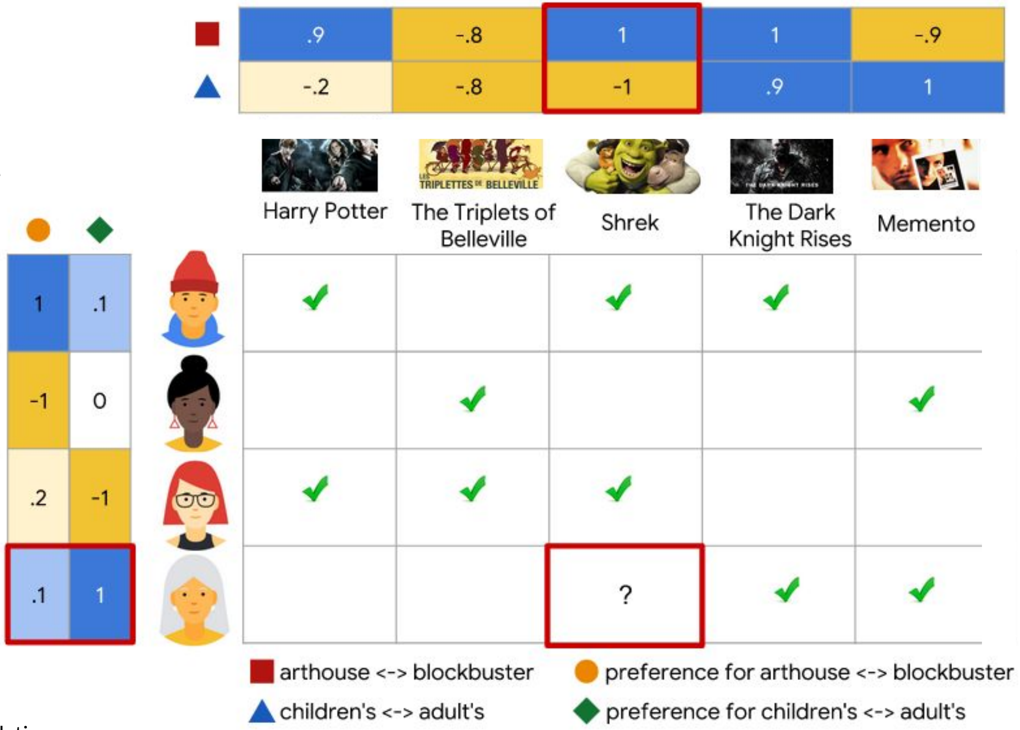
- Factorize the interaction matrix A
- A user embedding matrix U
- An item embedding matrix V



Source: https://developers.google.com/machine-learning/recommendation

Matrix factorization

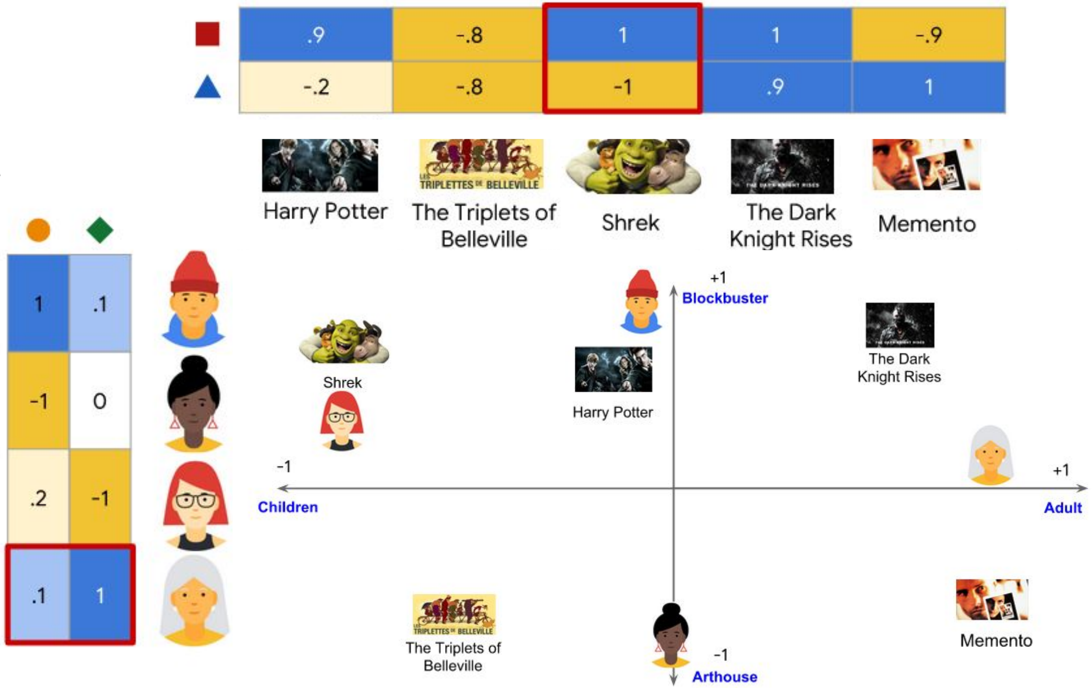
- Factorize the interaction matrix A
- A user embedding matrix U
- An item embedding matrix V



Source: https://developers.google.com/machine-learning/recommendation

Matrix factorization

- Factorize the interaction matrix A
- A user embedding matrix U
- An item embedding matrix V

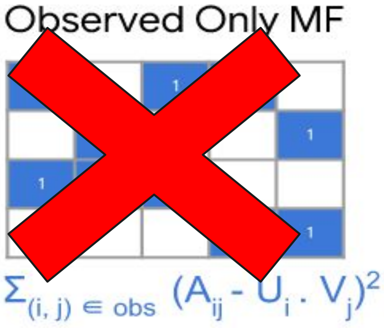


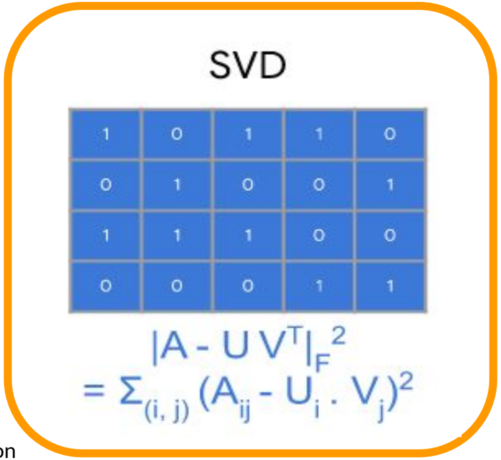
Source: https://developers.google.com/machine-learning/recommendation

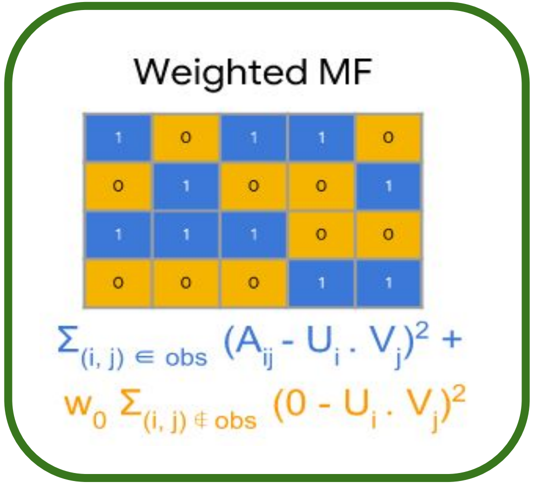
Matrix factorization

Optimization:

- Stochastic gradient descent
- Alternating Least Squares:
    - Fix U and solve for V
    - Fix V and solve for U







Source: https://developers.google.com/machine-learning/recommendation

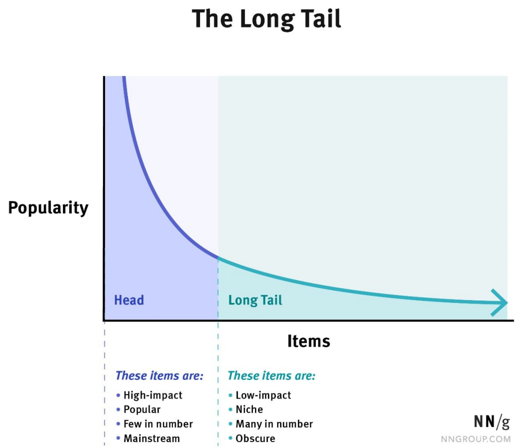
Matrix factorization

Advantages:

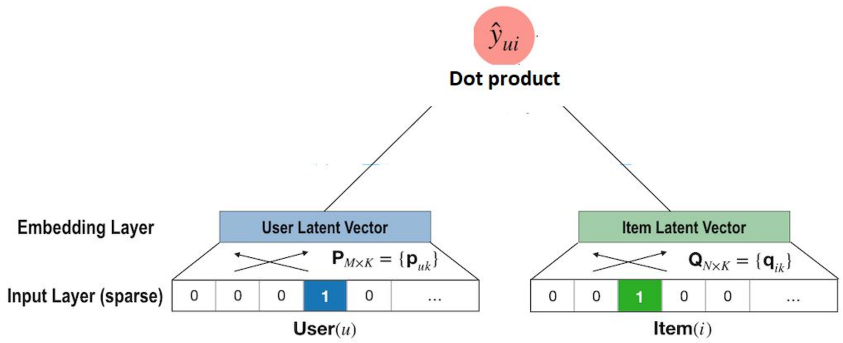
- Can be parallelized (ALS)
- Can be computed offline
- Embeddings can be used for item-item recommendations
- Good for serendipity

Drawbacks:

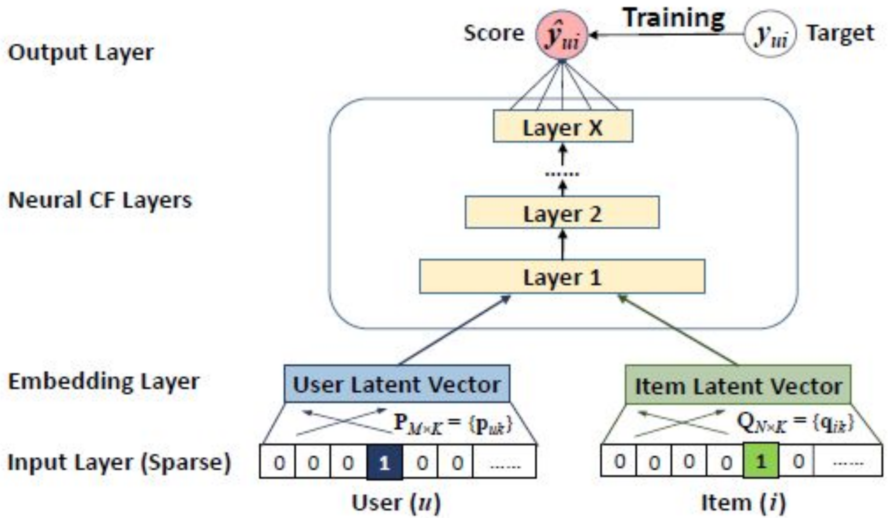
- Can't handle new items
- Does not include other possible meaningful features



Neural Collaborative Filtering

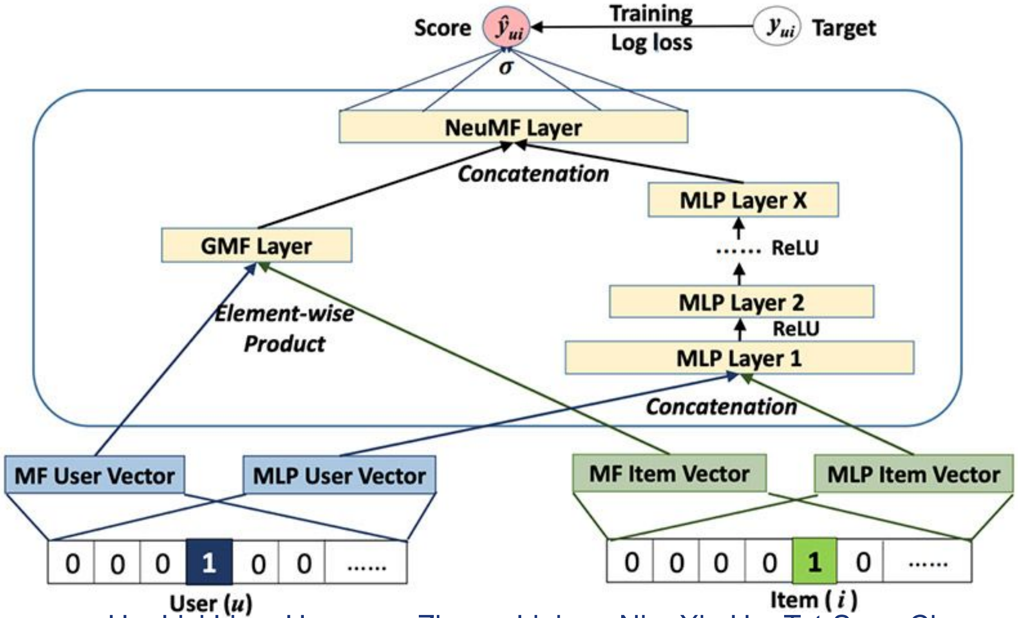


Neural Collaborative Filtering



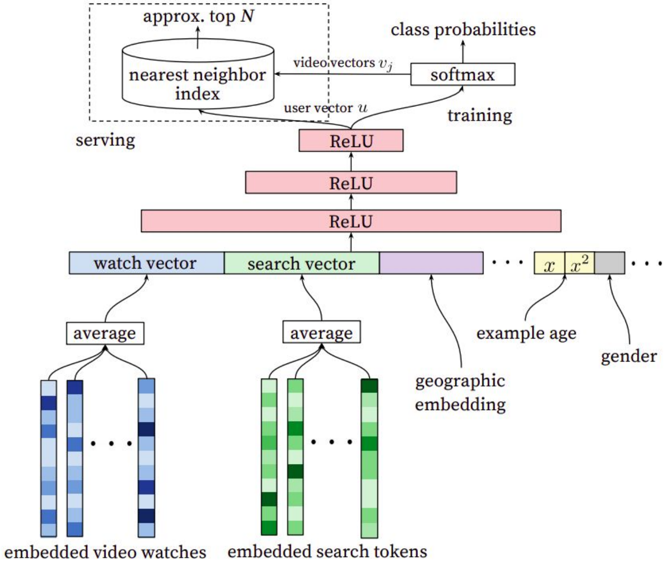
Neural collaborative filtering: Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, Tat-Seng Chua

Neural Collaborative Filtering



Neural collaborative filtering: Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, Tat-Seng Chua

Neural Collaborative Filtering



Deep neural networks for youtube recommendations: P Covington, J Adams, E Sargin

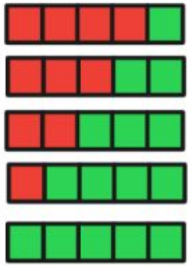
Evaluation

- Explicit feedbacks: MAE, MSE, RMSE
- Implicit feedbacks:
  - Precision@k:

Rank	Product	Is recom.	Result
1	product B	1	TP
2	product A	1	FP
3	product E	1	FP
4	product C	1	TP
5	product D	1	TP

P(k=5) = 3/5

  - Average precision (AP): $$\text{AP@N} = \frac{1}{m}\sum_{k=1}^{N}(P(k) \text{ if } k^{th} \text{ item was relevant})$$



Recommendations	Precision@k's	AP@5
	[0, 0, 0, 0, 1/5]	(1/5)(1/5) = 0.04
	[0, 0, 0, 1/4, 2/5]	(1/5)(1/4 + 2/5) = 0.13
	[0, 0, 1/3, 2/4, 3/5]	(1/5)(1/3 + 2/4 + 3/5) = 0.29
	[0, 1/2, 2/3, 3/4, 4/5]	(1/5)(1/2 + 2/3 + 3/4 + 4/5) = 0.54
	[1/1, 2/2, 3/3, 4/4, 5/5]	(1/5)(1/1 + 2/2 + 3/3 + 4/4 + 5/5) = 1

- Compare with best salers

source: https://neptune.ai/blog/recommender-systems-metrics

Recommender systems in real life

Real evaluation

- Click-through rates
- Adoption and conversion (percentage of song listened, percentage of products bought, …
- Global revenue
- User behaviour and engagement (are the user coming more often? Do they stay longer?)
- A/B testing

Tricks

- Efficient recommendation depend on context
- Efficiency depends on position
- Explaining why an item was recommended improves conversion rate
- Best sellers are an efficient option

Challenges

- Seasonality and behavioral change
- Change in catalog
- New users
- Fake ratings and bots
- User tend to give negative ratings only or over optimistic ones
- Recommendation constraints
- Scalability