Рекомендательные системы 2

Андрей Зимовнов

Яндекс, ВШЭ

Контентные рекомендации

Зачем content-based?

- **Похожести товаров**
  - Поступил новый товар → пока не знаем кому он нравится, но знаем его описание → можем считать похожести описания → решаем проблему холодного старта для товара

- **Похожести юзер-товар**
  - Хотим учитывать предпочтения юзера к режиссеру, словам описания, картинке, …

- **Похожести юзеров**
  - Например: любители вестернов и Иствуда ☺

Зачем content-based?

- **Похожести товаров**
  - Можно использовать в CF
- **Похожести юзер-товар**
  - Готовый признак
- **Похожести юзеров**
  - Можно использовать в CF

Какой бывает контент

Название	
Режиссер	
Год	
Актер	
Жанр	
Описание	

- Структурированный (характеристики)
- Текстовый (описание)
- Медиа (аудио, видео, картинки)

Структурированный

Самый простой тип контента!

Товар-товар: совпадение или близость характеристик

- одинаковый бренд
- похожая диагональ
- …

Структурированный

Самый простой тип контента!

Юзер-товар: предпочтение характеристики юзером

- Частота жанра в истории юзера
- Средняя оценка фильмов этого жанра в истории юзера
- …

Структурированный

Самый простой тип контента!

Юзер-юзер: совпадение профилей юзеров

- Смотрят в одинаковых пропорциях жанры
- Или любят тех же актеров
- …

Пример

	Комедия	Боевик	Вестерн	Триллер
Юзер 1	0.3	0.1	0.2	0.4
Юзер 2	0.1	0.2	0.3	0.4
Фильм 1	1	0	0	0
Фильм 2	0	0	0	1

Как измерить похожесть юзер-товар?

Косинус

$$a \cdot b = ||a|| \cdot ||b|| \cos\theta$$

$$\cos\theta = \frac{a \cdot b}{||a|| \cdot ||b||}$$

Пример

	Комедия	Боевик	Вестерн	Триллер
Юзер 1	0.3	0.1	0.2	0.4
Юзер 2	0.1	0.2	0.3	0.4
Фильм 1	1	0	0	0
Фильм 2	0	0	0	1

Как измерить похожесть юзер-товар?

Косинус! Как и для других похожестей!

Пример

	Комедия	Боевик	Вестерн	Триллер
Юзер 1	0.3	0.1	0.2	0.4
Юзер 2	0.1	0.2	0.3	0.4
Фильм 1	1	0	0	0
Фильм 2	0	0	0	1

Как измерить похожесть юзер-товар?

Косинус! Как и для других похожестей!

Текстовый контент

Мешок слов

COUNT	a	Accelerating	and	applications	art	behavior	Building	Consumer	CRM	customer	data	for	Handbook	Introduction	Knowledge	Management	Marketing	Mastering	mining	of	relationship	Research	science	technology	the	to	using	website	your
Building data mining applications for CRM				1			1		1		1	1							1										
Accelerating customer relationships: using CRM and relationship technologies		1	1						1	1											2			1			1		
Mastering Data Mining: the art and science of Customer Relationship Management			1		1					1	1					1		1	1	1	1		1		1				
Data Mining your website											1								1									1	1
Introduction to Marketing														1			1									1			
Consumer behavior						1		1																					
Marketing Research: a Handbook	1												1				1					1							
Customer Knowledge Management										1					1	1													

Чуть умнее: TF-IDF

Частые слова менее важны: они есть везде, по ним все будут похожи

Редкие слова более важны: какая-то узкая тематика

Но не слишком редкие: опечатки ☺

Term Frequency

частота термина t
в документе d

$$TF_{t,d} = \frac{f_{td}}{\max_{z} f_{zd}}$$

максимальная частота
по всем терминам z
в документе d

Inverse Document Frequency

$$IDF_t = \log \frac{N}{df_t}$$

число документов в корпусе

число документов с термином t

TF-IDF

$$TF_{t,d} * IDF_t = \frac{f_{td}}{\max_z f_{zd}} \log \frac{N}{df_t}$$

TF-IDF

TFIDF Normed Vectors	a	Accelerating	and	applications	art	behavior	Building	Consumer	CRM	customer	data	for	Handbook	Introduction	Knowledge	Management	Marketing	Mastering	mining	of	relationship	Research	science	technology	the	to	using	website	your
Building data mining applications for CRM				0.502			0.502		0.344		0.251	0.502							0.251										
Accelerating customer relationships: using CRM and relationship technologies		0.432	0.296						0.296	0.216											0.468			0.432			0.432		
Mastering Data Mining: the art and science of Customer Relationship Management			0.256		0.374					0.187	0.187					0.256		0.374	0.187	0.374	0.256		0.374		0.374				
Data Mining your website											0.316								0.316									0.632	0.632
Introduction to Marketing														0.636			0.436									0.636			
Consumer behavior						0.707		0.707																					
Marketing Research: a Handbook	0.537												0.537				0.368					0.537							
Customer Knowledge Management										0.381					0.736	0.522													

Что с этим делать дальше?

Текстовая похожесть

- Косинусная похожесть неплохо работает на мешках слов!
- Можем находить похожие по описанию товары!

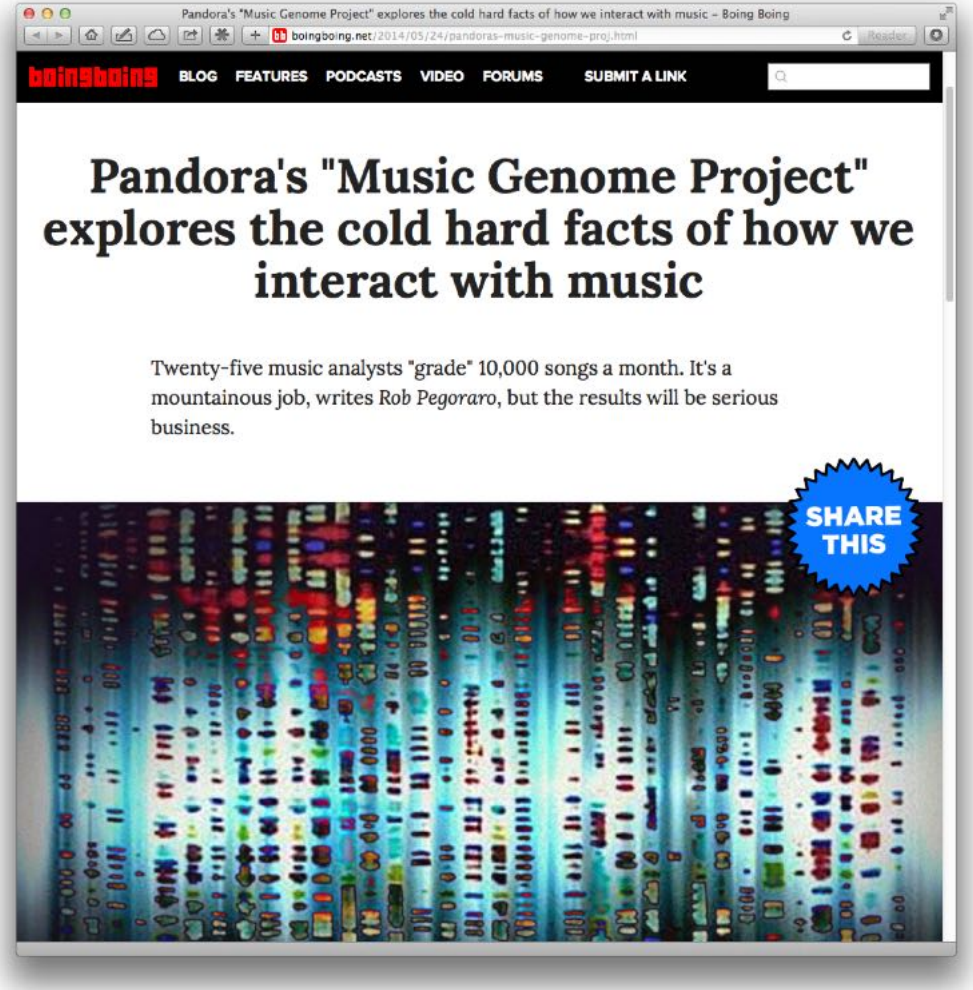
Текстовая похожесть

Плюсы:

- Решает проблему холодного старта для нового товара!
- Может рекомендовать даже непопулярные товары
- Интерпретируемые похожести (мешок слов)

Пример: Pandora Radio

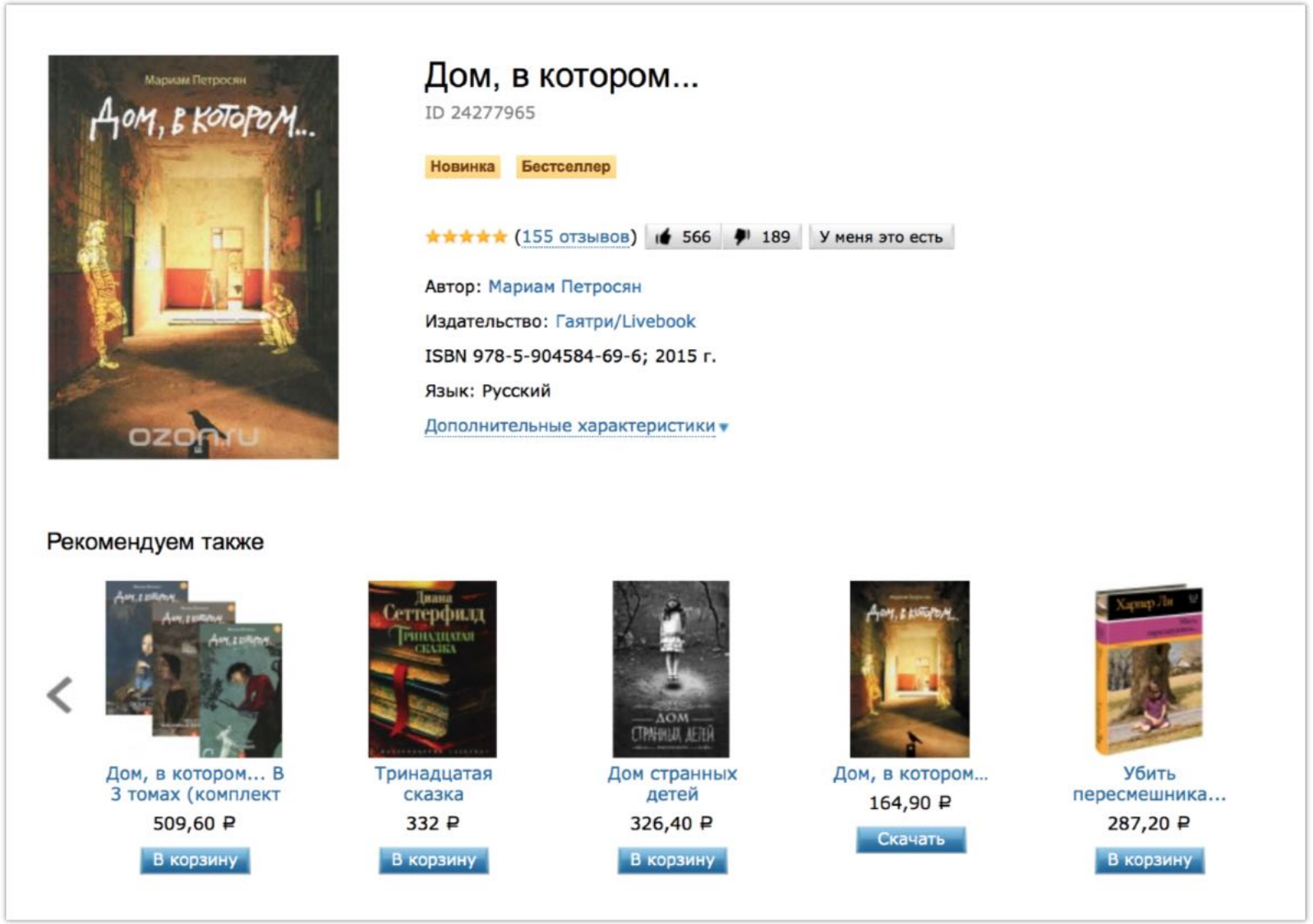
- Поставили много на контентные признаки
- 480 характеристик у каждой песни!
- Размечают эксперты (25 экспертов размечают 10000 песен в месяц)!



https://boingboing.net/2014/05/24/pandoras-music-genome-proj.html

Пример: ozon.ru

Контентные признаки в книгах особенно полезны!



Матричные факторизации: SVD

Уменьшение размерности

- Матрица рейтингов **разреженная** (много пропусков)
- Оценки товаров **коррелируют** (юзерам часто нравятся одни и те же пары товаров)
- **Как сжать матрицу оценок?**

Метод главных компонент

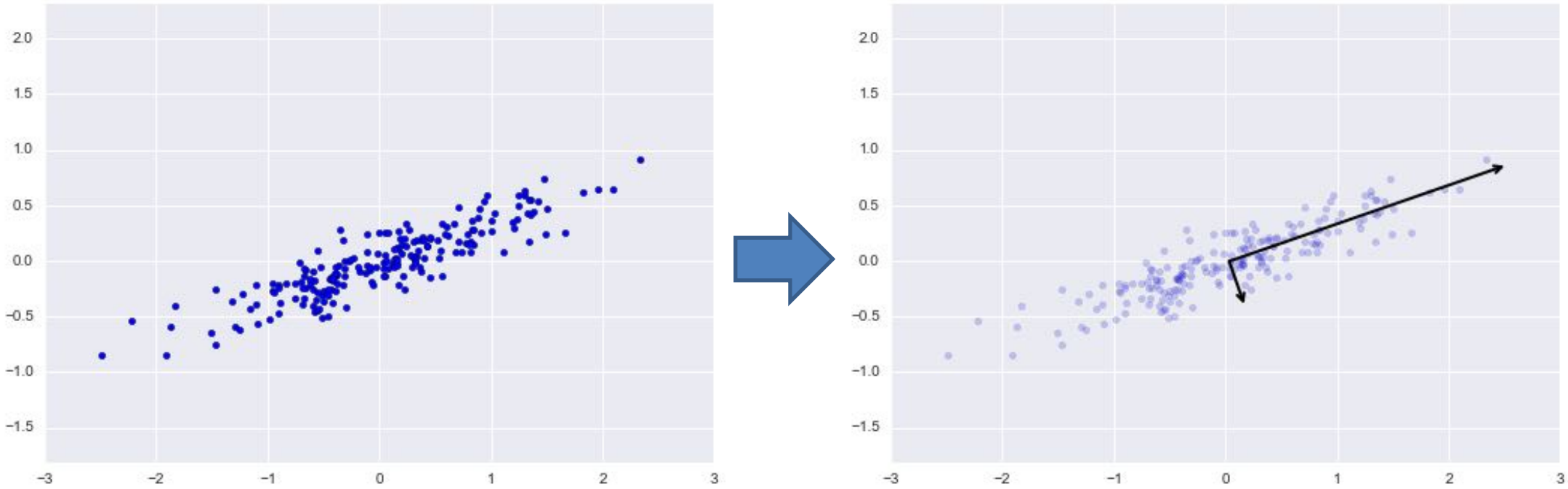
Также известен как PCA (Principal Component Analysis)

Идея: найдем для наших данных новые оси координат, которые лучше объясняют эти данные!

https://jakevdp.github.io/PythonDataScienceHandbook/05.09-principal-component-analysis.html

Метод главных компонент

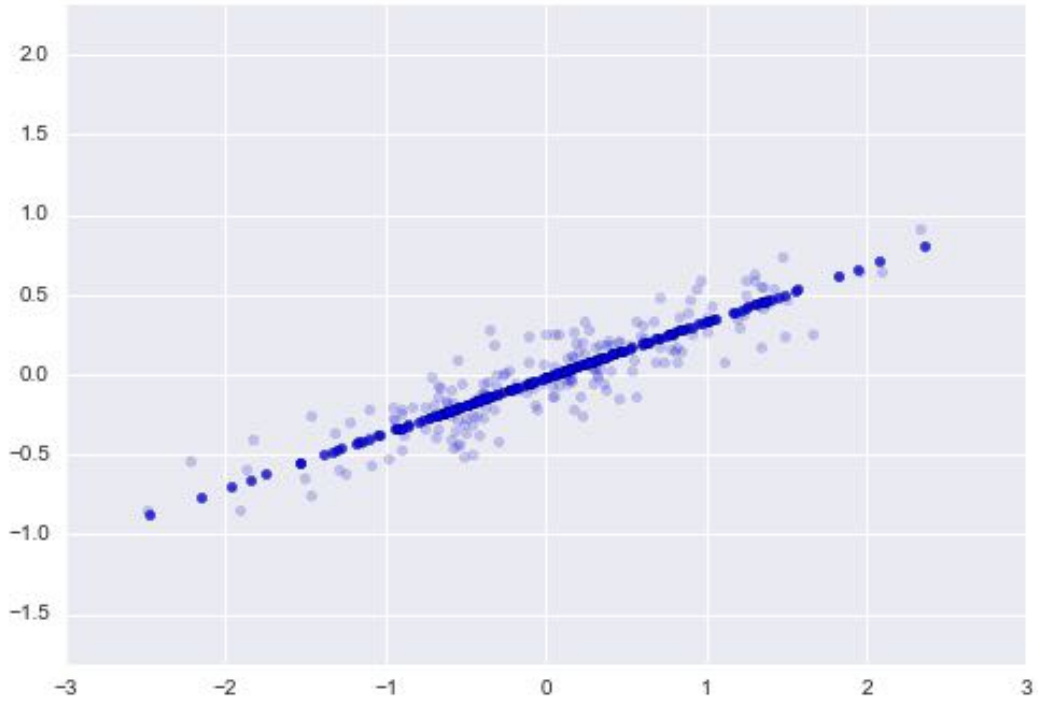
Новые оси координат ортогональны, они же все-таки оси



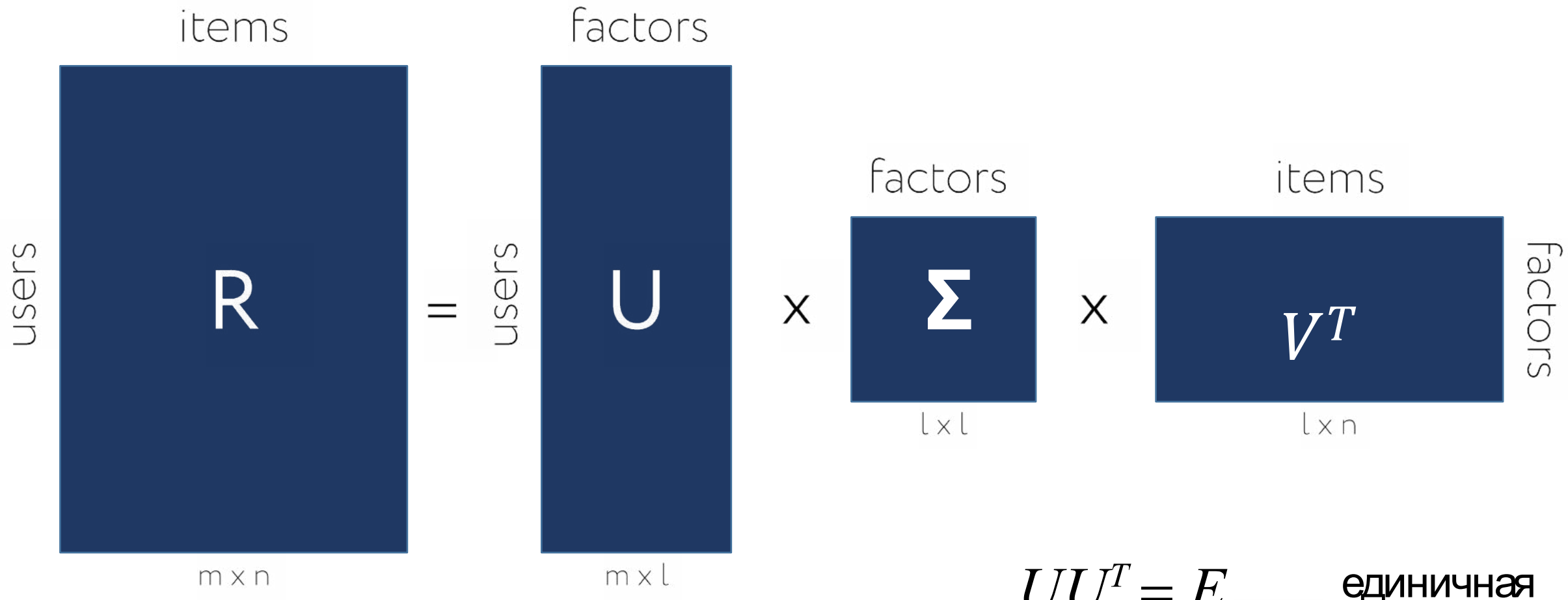
Метод главных компонент

Возьмем проекцию только на первую главную ось (они отсортированы по убыванию объясняемой дисперсии).

Сжали данные, потеряв немного информации!



PCA реализуют через SVD



Факторизация матрицы

$$UU^T = E$$

$$VV^T = E$$

единичная матрица

Интерпретация для рейтингов

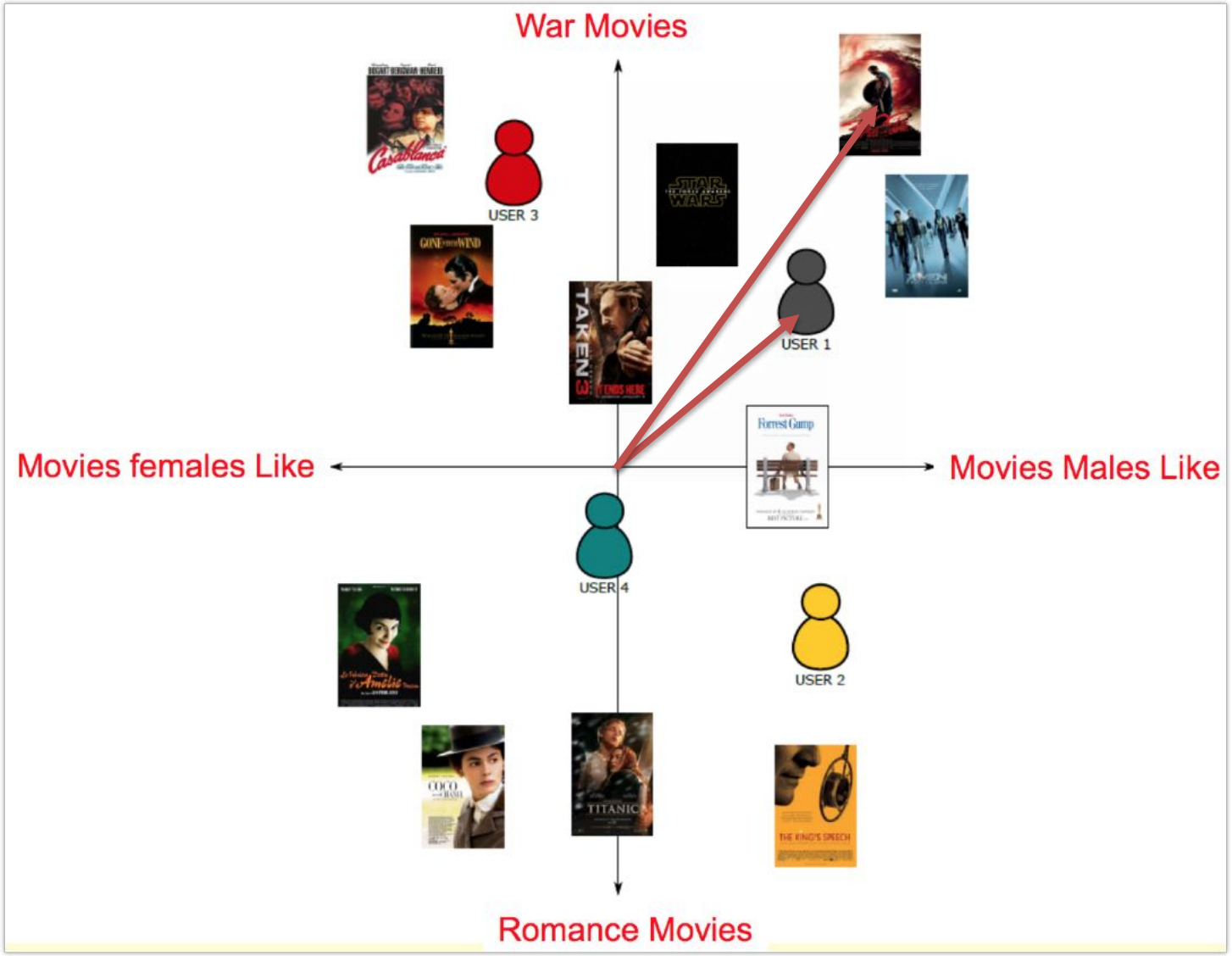
Юзеры и товары погружаются в новое пространство (малой размерности).

Координаты в этом пространстве – «характеристики» товаров и юзеров.

Похожесть в этом пространстве – это скалярное произведение!

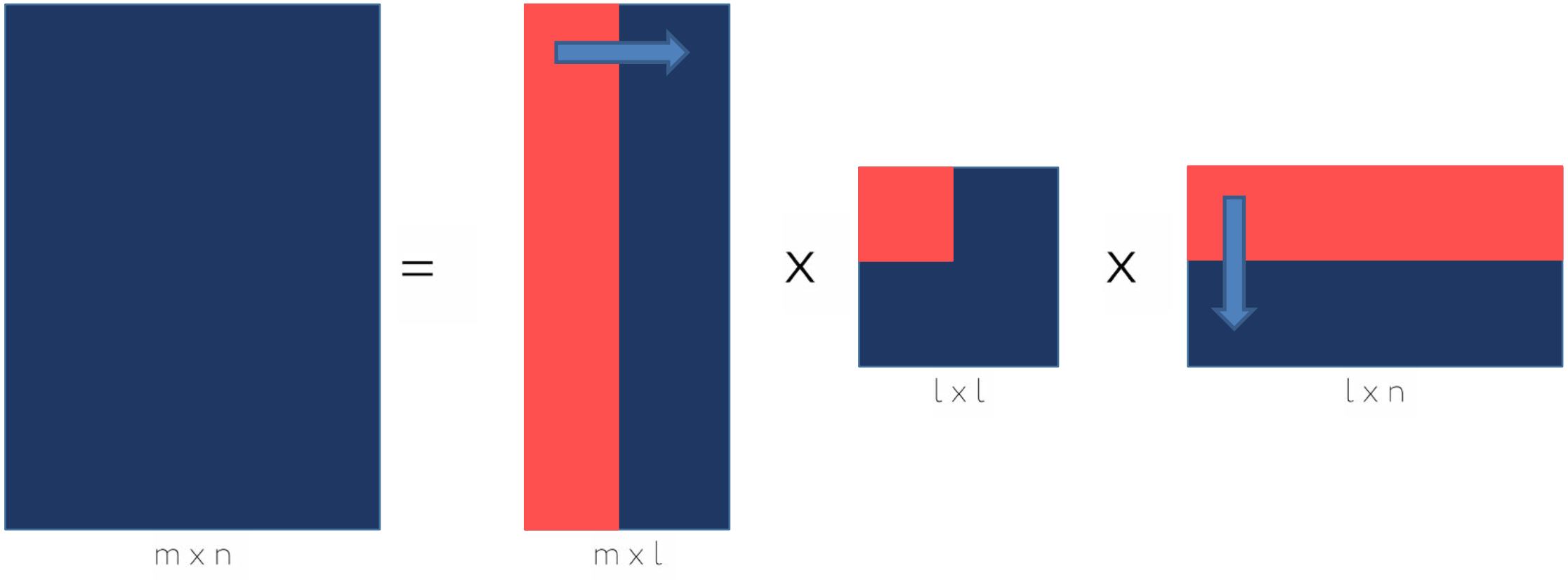
Интерпретация для рейтингов

Взяли 2 компоненты



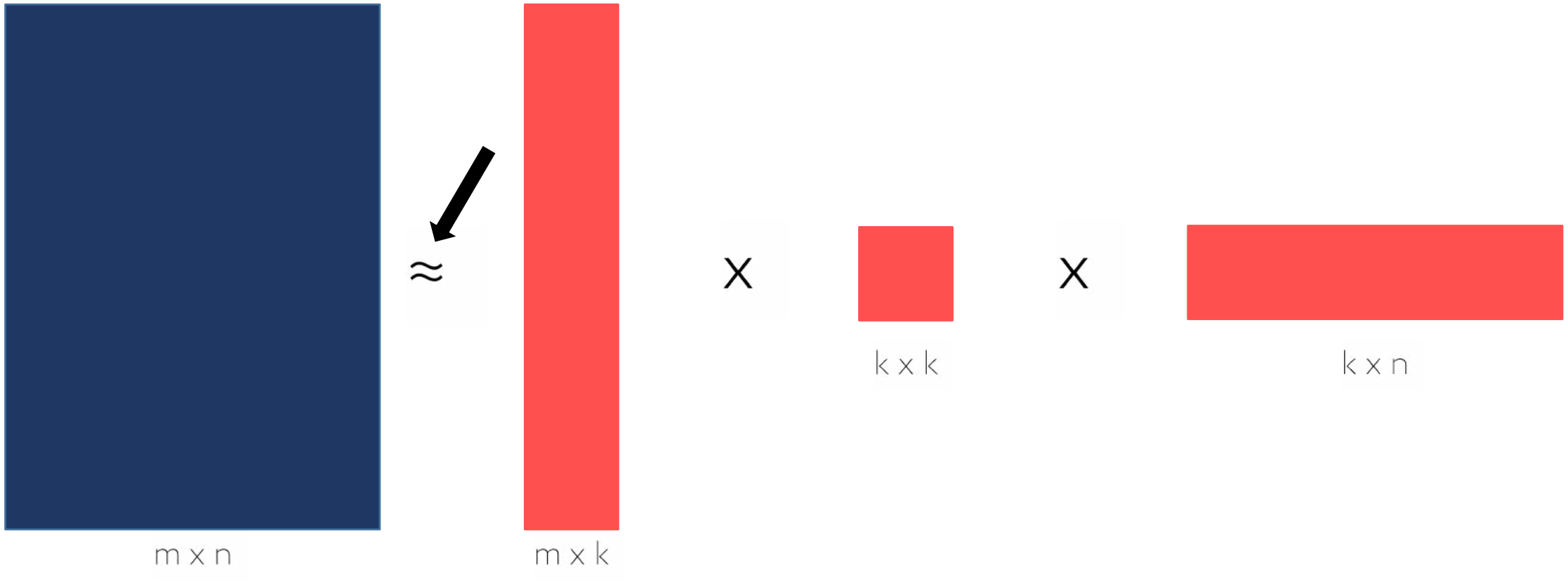
Эти оси модель может найти сама!

Свойство SVD



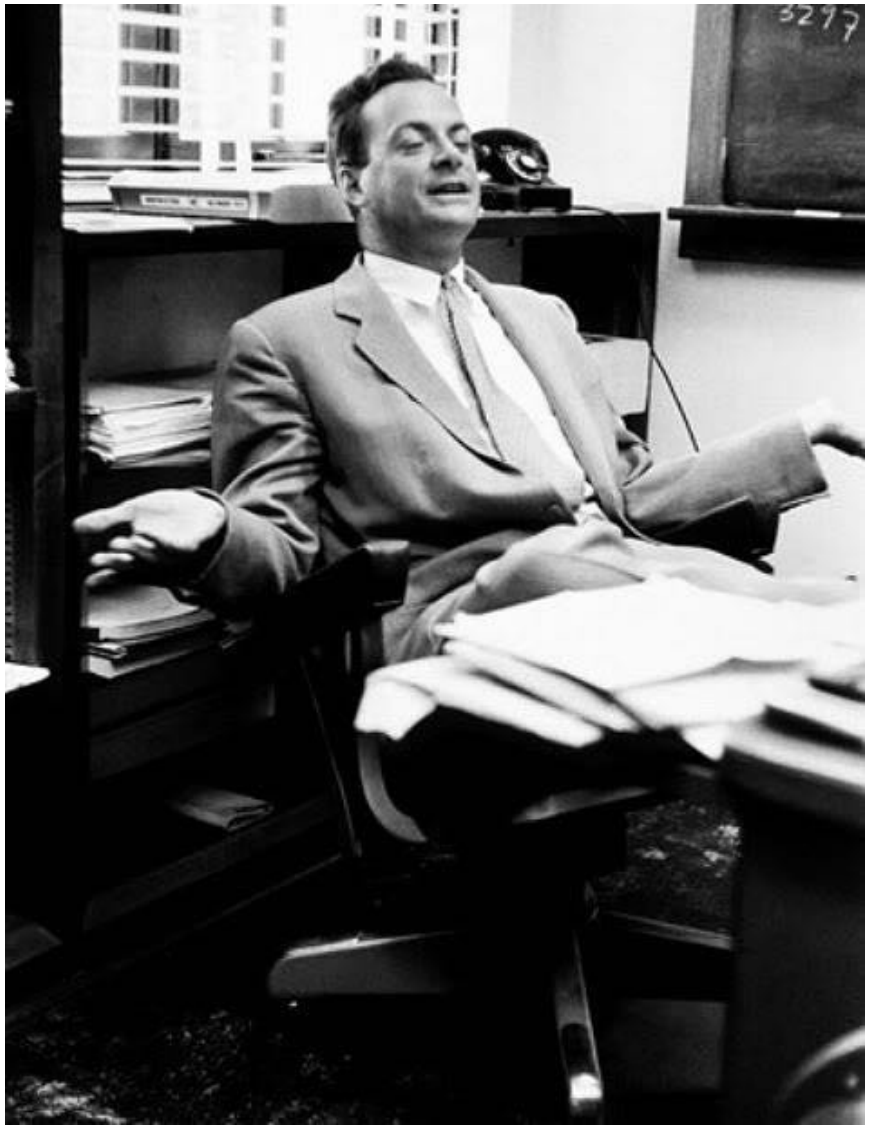
Векторы отсортированы по убыванию вклада в приближение матрицы!

Свойство SVD



Возьмем только часть векторов!

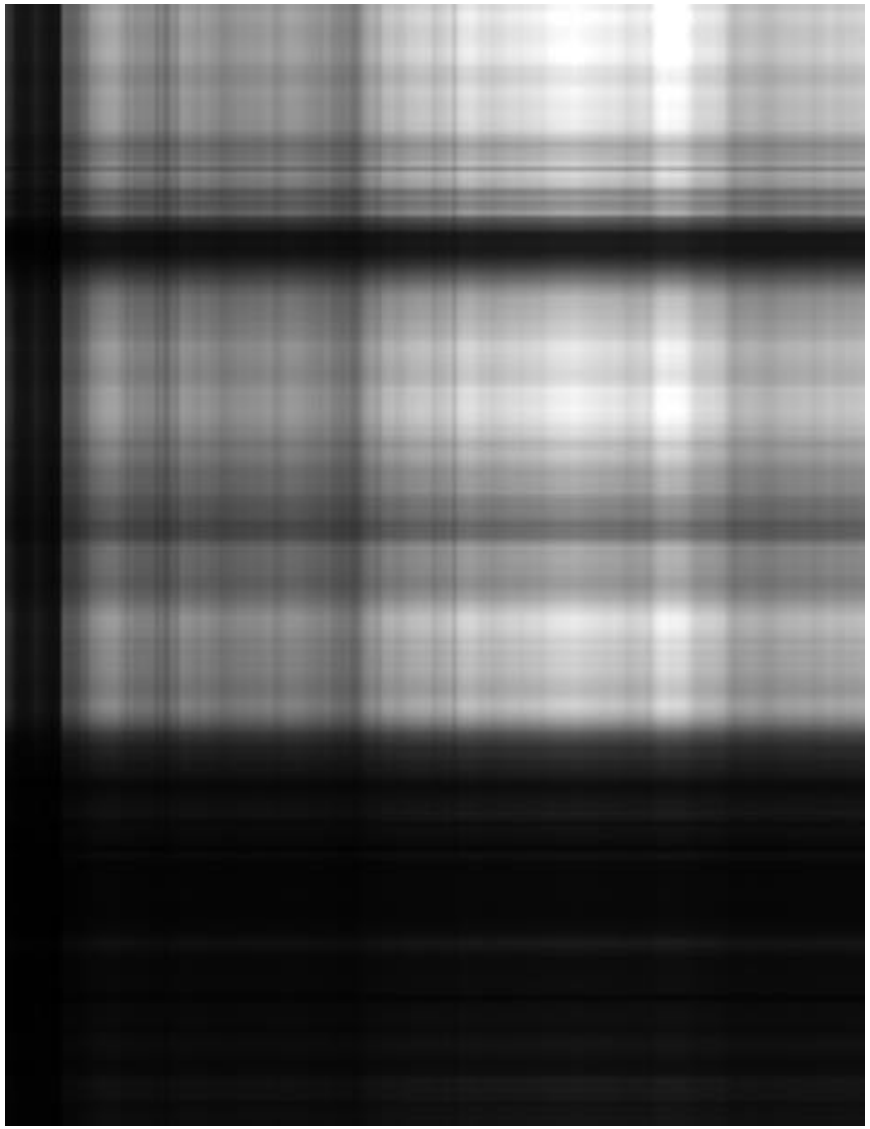
SVD для картинки



Рассмотрим качество приближения для особенной матрицы – картинки!

Картинка – это набор чисел от 0 до 255 (яркость).

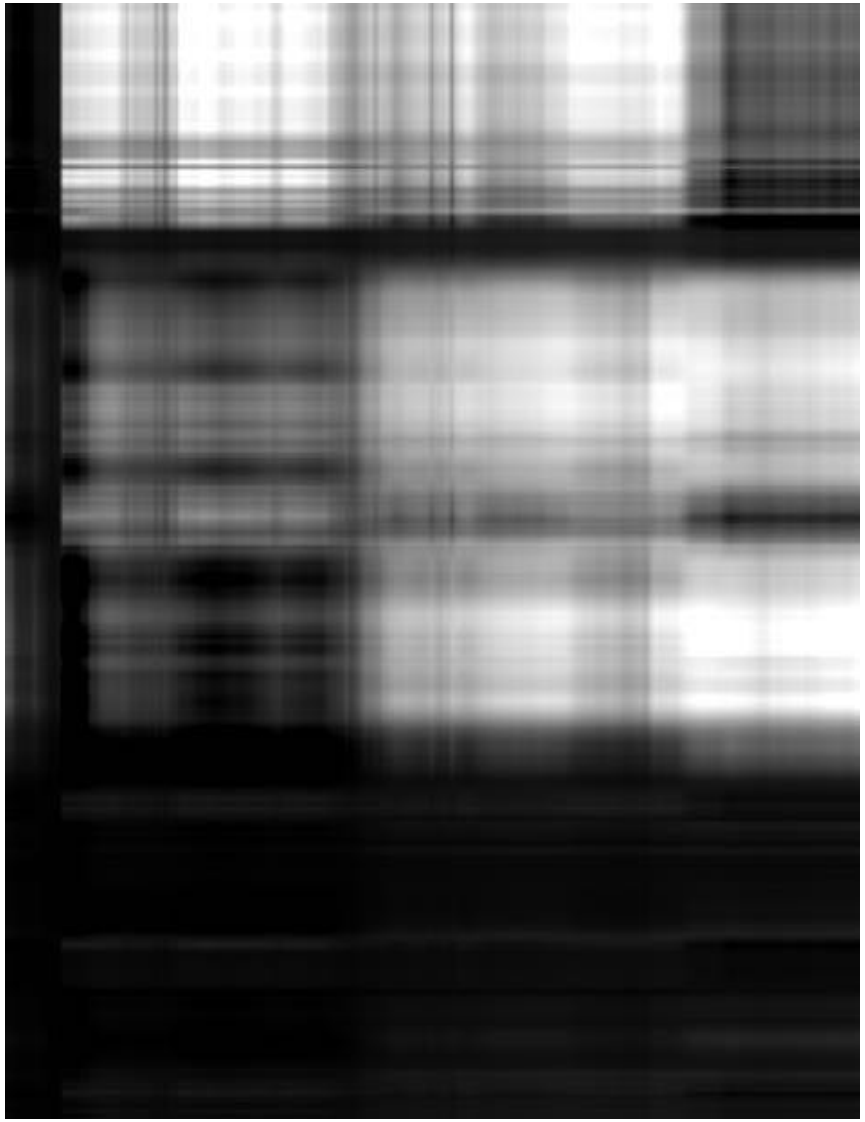
SVD для картинки



Оставили 1 компоненту.

Восстановили наиболее сильные изменения яркости (сверху светло, внизу темно).

SVD для картинки



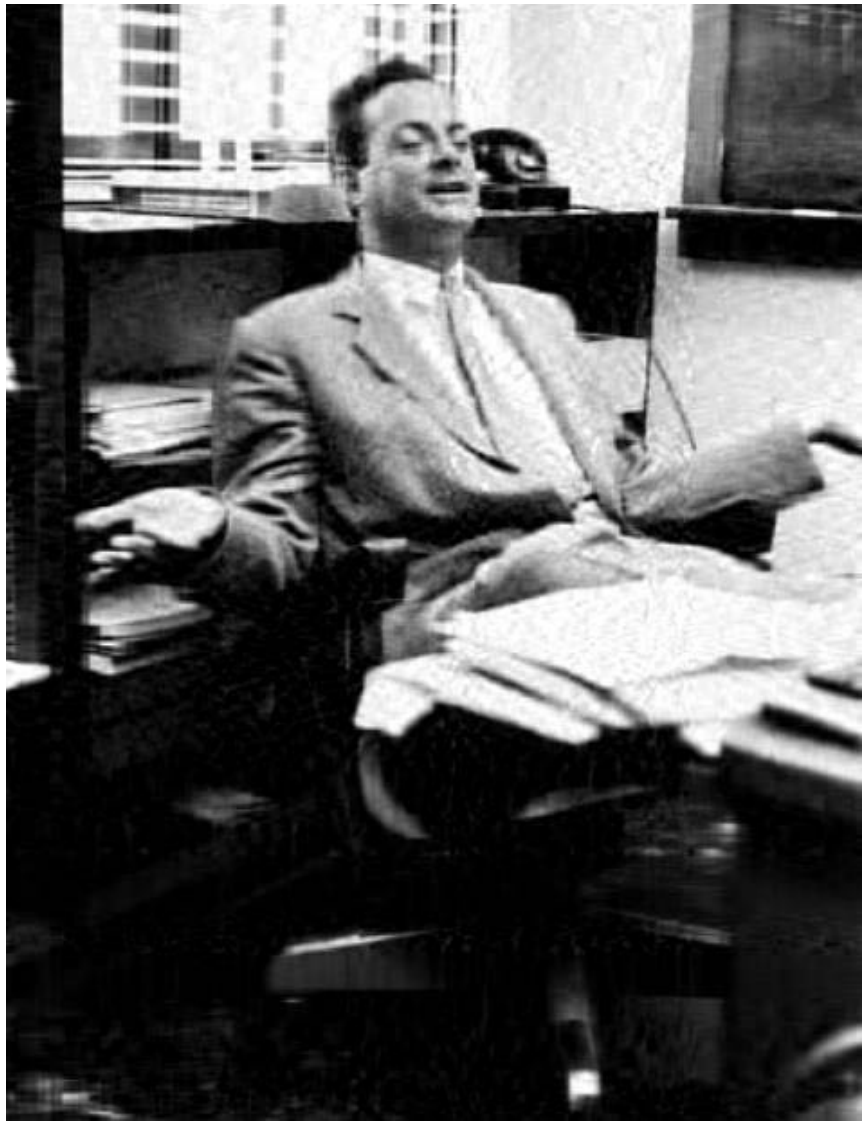
2 компоненты.

SVD для картинки

10 компонент.

Уже практически виден человек.

SVD для картинки

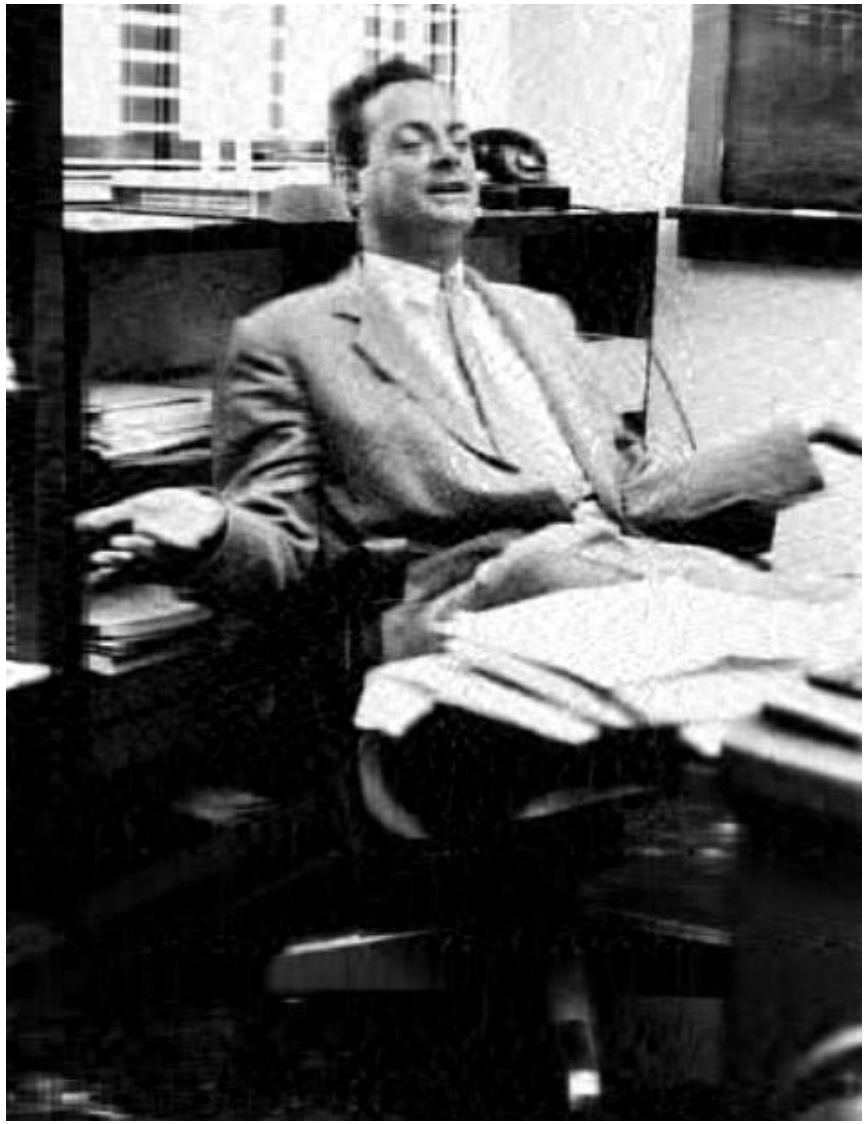


50 компонент.

Исходная картинка размера 500x600.

Насколько сильно сжали?

SVD для картинки



50 компонент.

Исходная картинка размера 500x600.

Насколько сильно сжали?

500*50 + 50*50 + 50*600 = 57500.
500*600 = 300000.
В пять раз!

Пример для рейтингов

	Yesterday, Beatles	Summertime Sadness, Lana Del Rey	November Rain, Guns 'n Roses	Diamonds, Rihanna	Highway to Hell, AC/DC	What a Wonderful World, Louis Armstrong	Hit the Road Jack!, Ray Charles
Артем	5	2	5	2	5	4	4
Вася	4	3	3	3	3	5	5
Маша	3	5	2	5	2	4	3
Саша	5	2	4	2	5	4	4
Клара	5	5	3	4	3	5	5

Результат SVD

Профили юзеров:

U

	f1	f2	f3
Артем	0.3262	0.5236	-0.455
Вася	-0.719	-0.052	-0.44
Маша	0.5477	-0.644	-0.392
Саша	0.1567	0.4523	-0.44
Клара	-0.23	-0.322	-0.503

Важность факторов:

$\sum$

	f1	f2	f3
f1	22.73	0	0
f2	0	5.3211	0
f3	0	0	1.7328

Профили товаров (по столбцам):

V^T

Интерпретация компонент?

	Yesterday, Beatles	Summertime Sadness, Lana Del Rey	November Rain, Guns 'n Roses	Diamonds, Rihanna	Highway to Hell, AC/DC	What a Wonderful World, Louis Armstrong	Hit the Road Jack!, Ray Charles
f1	0.0205983	0.23140511	0.29370711	0.36389253	0.38414103	-0.35664957	-0.67274252
f2	0.2122569	-0.57029559	0.37908873	-0.50977121	0.46409138	-0.10192274	0.01912943
f3	-0.43649738	-0.33353455	-0.33632132	-0.31142383	-0.35567176	-0.43374969	-0.41651737

Результат SVD

Профили юзеров:

U

	f1	f2	f3
Артем	0.3262	0.5236	-0.455
Вася	-0.719	-0.052	-0.44
Маша	0.5477	-0.644	-0.392
Саша	0.1567	0.4523	-0.44
Клара	-0.23	-0.322	-0.503

Важность факторов:

$\sum$

	f1	f2	f3
f1	22.73	0	0
f2	0	5.3211	0
f3	0	0	1.7328

Профили товаров (по столбцам):

V^T

f1 – энергичность

	Yesterday, Beatles	Summertime Sadness, Lana Del Rey	November Rain, Guns 'n Roses	Diamonds, Rihanna	Highway to Hell, AC/DC	What a Wonderful World, Louis Armstrong	Hit the Road Jack!, Ray Charles
f1	0.0205983	0.23140511	0.29370711	0.36389253	0.38414103	-0.35664957	-0.67274252
f2	0.2122569	-0.57029559	0.37908873	-0.50977121	0.46409138	-0.10192274	0.01912943
f3	-0.43649738	-0.33353455	-0.33632132	-0.31142383	-0.35567176	-0.43374969	-0.41651737

Результат SVD

Профили юзеров:

U

	f1	f2	f3
Артем	0.3262	0.5236	-0.455
Вася	-0.719	-0.052	-0.44
Маша	0.5477	-0.644	-0.392
Саша	0.1567	0.4523	-0.44
Клара	-0.23	-0.322	-0.503

Важность факторов:

$\sum$

	f1	f2	f3
f1	22.73	0	0
f2	0	5.3211	0
f3	0	0	1.7328

Профили товаров (по столбцам):

V^T

f2 – рок

	Yesterday, Beatles	Summertime Sadness, Lana Del Rey	November Rain, Guns 'n Roses	Diamonds, Rihanna	Highway to Hell, AC/DC	What a Wonderful World, Louis Armstrong	Hit the Road Jack!, Ray Charles
f1	0.0205983	0.23140511	0.29370711	0.36389253	0.38414103	-0.35664957	-0.67274252
f2	0.2122569	-0.57029559	0.37908873	-0.50977121	0.46409138	-0.10192274	0.01912943
f3	-0.43649738	-0.33353455	-0.33632132	-0.31142383	-0.35567176	-0.43374969	-0.41651737

Результат SVD

Профили юзеров:

U

	f1	f2	f3
Артем	0.3262	0.5236	-0.455
Вася	-0.719	-0.052	-0.44
Маша	0.5477	-0.644	-0.392
Саша	0.1567	0.4523	-0.44
Клара	-0.23	-0.322	-0.503

Важность факторов:

$\sum$

	f1	f2	f3
f1	22.73	0	0
f2	0	5.3211	0
f3	0	0	1.7328

Профили товаров (по столбцам):

V^T

f3 – кажется рандом ☺

	Yesterday, Beatles	Summertime Sadness, Lana Del Rey	November Rain, Guns 'n Roses	Diamonds, Rihanna	Highway to Hell, AC/DC	What a Wonderful World, Louis Armstrong	Hit the Road Jack!, Ray Charles
f1	0.0205983	0.23140511	0.29370711	0.36389253	0.38414103	-0.35664957	-0.67274252
f2	0.2122569	-0.57029559	0.37908873	-0.50977121	0.46409138	-0.10192274	0.01912943
f3	-0.43649738	-0.33353455	-0.33632132	-0.31142383	-0.35567176	-0.43374969	-0.41651737

Хватит 2 компоненты

	Yesterday, Beatles	Summertime Sadness, Lana Del Rey	November Rain, Guns 'n Roses	Diamonds, Rihanna	Highway to Hell, AC/DC	What a Wonderful World, Louis Armstrong	Hit the Road Jack!, Ray Charles
Артем	5	2	5	2	5	4	4
Вася	4	3	3	3	3	5	5
Маша	3	5	2	5	2	4	3
Саша	5	2	4	2	5	4	4
Клара	5	5	3	4	3	5	5

	Yesterday, Beatles	Summertime Sadness, Lana Del Rey	November Rain, Guns 'n Roses	Diamonds, Rihanna	Highway to Hell, AC/DC	What a Wonderful World, Louis Armstrong	Hit the Road Jack!, Ray Charles
Артем	5	2	5	2	5	4	4
Вася	4	3	3	3	3	4	4
Маша	3	5	2	5	2	4	4
Саша	5	2	4	2	5	4	4
Клара	5	5	3	4	3	5	5

Ошибок уже мало!

Заметили подвох?

	Yesterday, Beatles	Summertime Sadness, Lana Del Rey	November Rain, Guns 'n Roses	Diamonds, Rihanna	Highway to Hell, AC/DC	What a Wonderful World, Louis Armstrong	Hit the Road Jack!, Ray Charles
Артем	5	2	5	2	5	4	4
Вася	4	3	3	3	3	5	5
Маша	3	5	2	5	2	4	3
Саша	5	2	4	2	5	4	4
Клара	5	5	3	4	3	5	5

В реальных задачах большинство ячеек пропущены!

Как быть?

Наивный подход

Заполнить пропуски нулями и применить SVD.

Лучше так не делать:

- Вносим шум в данные, ведь на самом деле там не нули!
- Матрица становится плотной (в противовес разреженной) → занимает терабайты, честный SVD считается годами ☺

Распишем рейтинг

вес фичи

$$\hat{r}_{ai} = \sum_{f=1}^{k} u_{af} \sigma_f v_{if}$$

интерес пользователя к данной фиче

релевантность данной фичи товару

Избавимся от матрицы весов

$$\hat{r}_{ai} = \sum_{f=1}^{k} u_{af}\sigma_f v_{if} = \sum_{f=1}^{k} \left(u_{af}\sqrt{\sigma_f}\right)\left(\sqrt{\sigma_f}\, v_{if}\right) = q_i^T p_a$$

Для простоты

Добавим сдвиги

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u$$

- μ – глобальное среднее по таблице рейтингов
- b_u – индивидуальный эффект пользователя (оптимизм) u
- b_i – индивидуальный эффект товара (качество) i
- q_i – профиль товара (вектор в пространстве фичей)
- p_u – профиль пользователя (вектор в пространстве фичей)
- q_i^T – транспонированный вектор

Задача оптимизации

$$\min_{q_*, p_*, b_*} \sum_{u,i} (r_{ui} - \mu - b_u - b_i - q_i^T p_u)^2$$

Сумма только по имеющимся рейтингам!

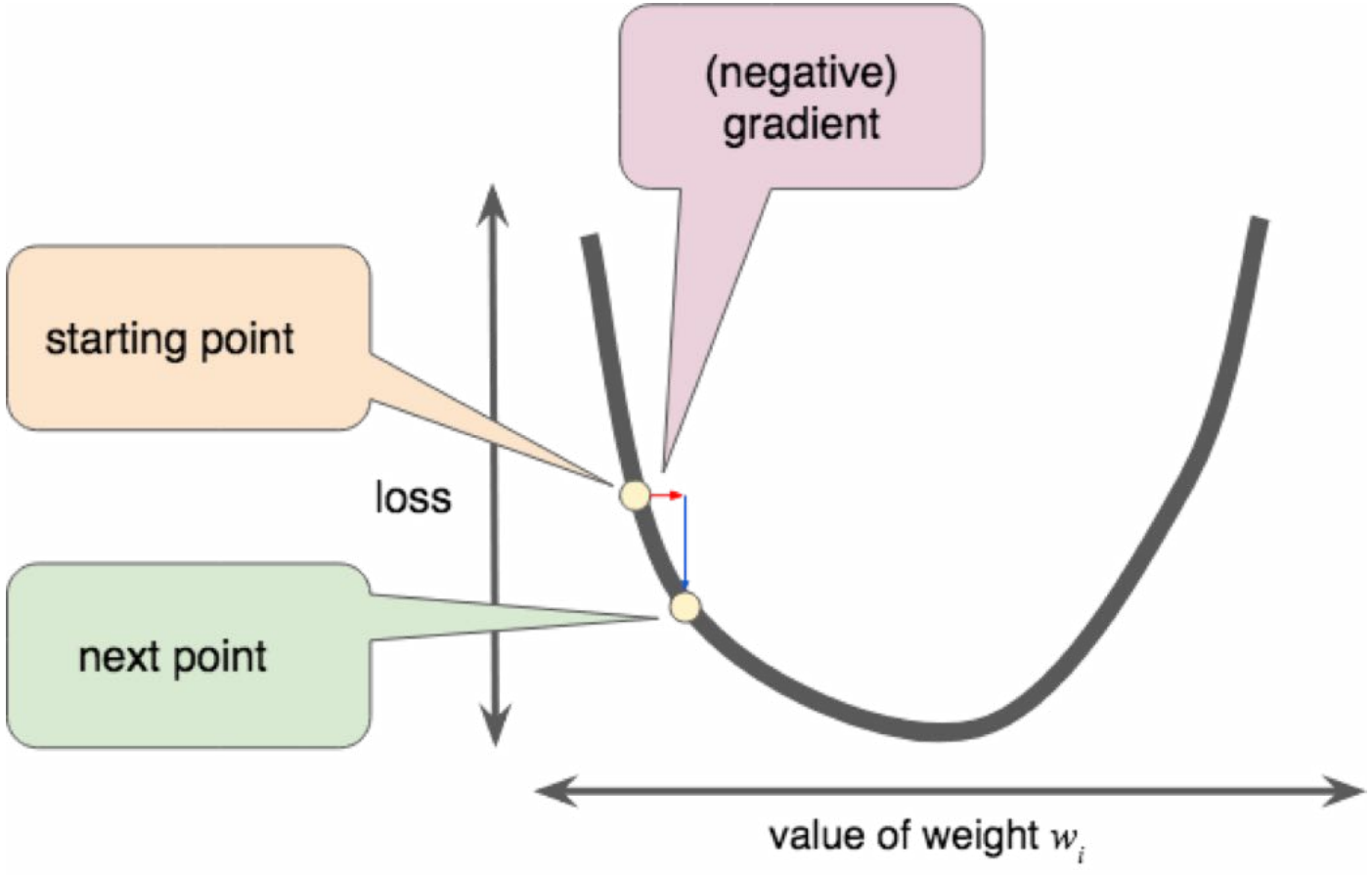
С регуляризацией

$$\min_{q_*, p_*, b_*} \sum_{u,i} (r_{ui} - \mu - b_u - b_i - q_i^T p_u)^2 + \lambda(b_i^2 + b_u^2 + \|q_i\|^2 + \|p_u\|^2)$$

Сумма только по имеющимся рейтингам!

Как решать?

Градиентный спуск (GD)



Stochastic GD (Funk SVD)

- Начинаем со случайных профилей (~$N(0, 0.01)$)
- **В цикле:** берем случайный рейтинг, двигаем под него профили товара и юзера

$$q_{if} \leftarrow q_{if} + \gamma(e_{ui} p_{uf} - \lambda q_{if})$$

$$p_{uf} \leftarrow p_{uf} + \gamma(e_{ui} q_{if} - \lambda p_{uf})$$

$$b_i \leftarrow b_i + \gamma(e_{ui} - \lambda b_i)$$

$$b_u \leftarrow b_u + \gamma(e_{ui} - \lambda b_u)$$

Частные производные

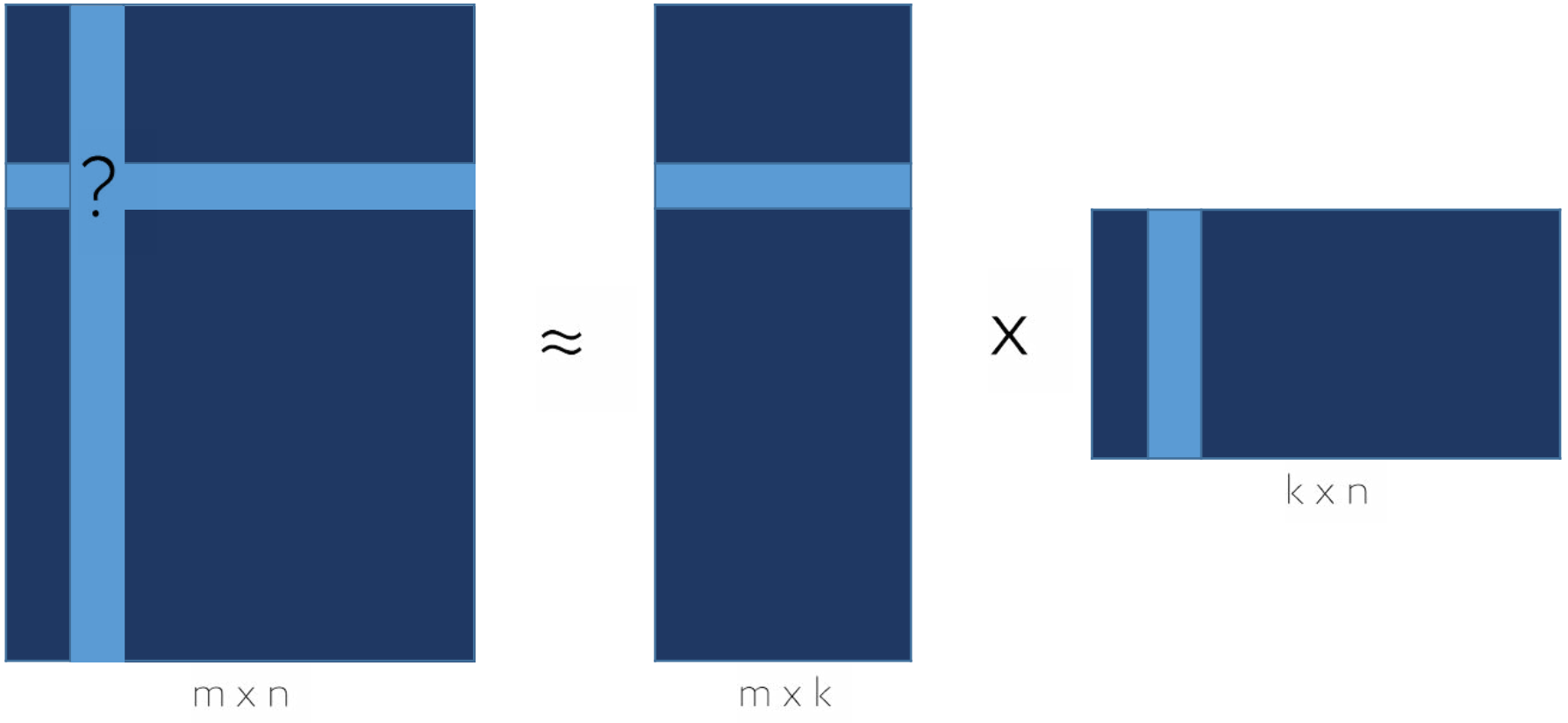
$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T p_u$$

$$e_{ui} = r_{ui} - \hat{r}_{ui}$$

- e_{ui} – ошибка
- λ – коэффициент регуляризации
- γ - шаг градиентного спуска

Netflix → $\lambda = 0.005, \gamma = 0.02, \gamma \leftarrow 0.9\gamma$

Как делать предсказания



Резюме: Funk SVD

Работает лучше, чем классическая CF!

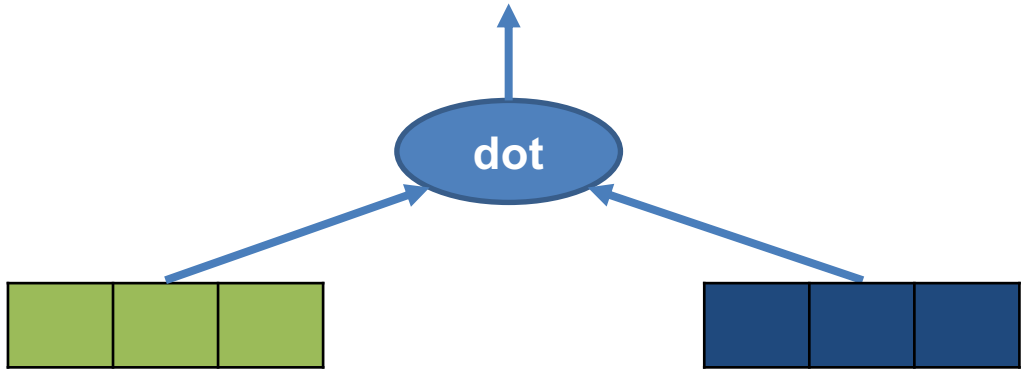
Потому что не считаются напрямую попарные похожести (которые могут быть неуверенными)

А ищется такое пространство, в котором все эти неуверенные похожести одновременно объясняются.

Нейросети в рекомендациях

Funk SVD как нейросеть

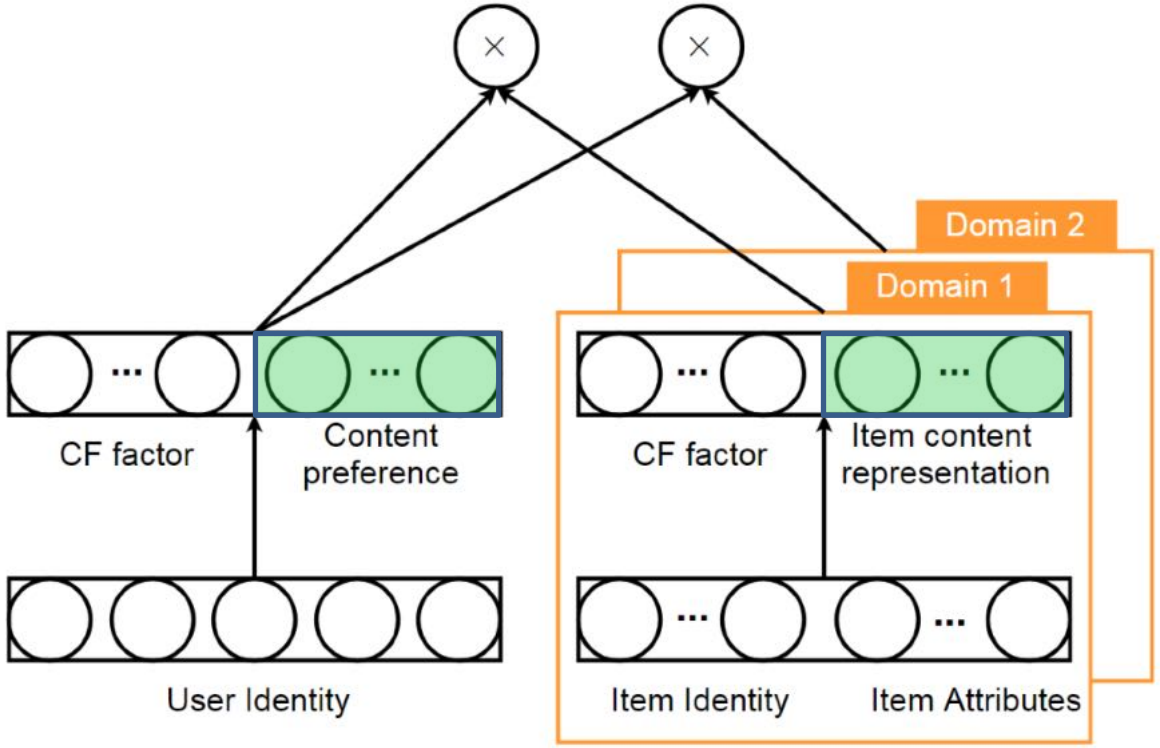
Обучаем при помощи SGD



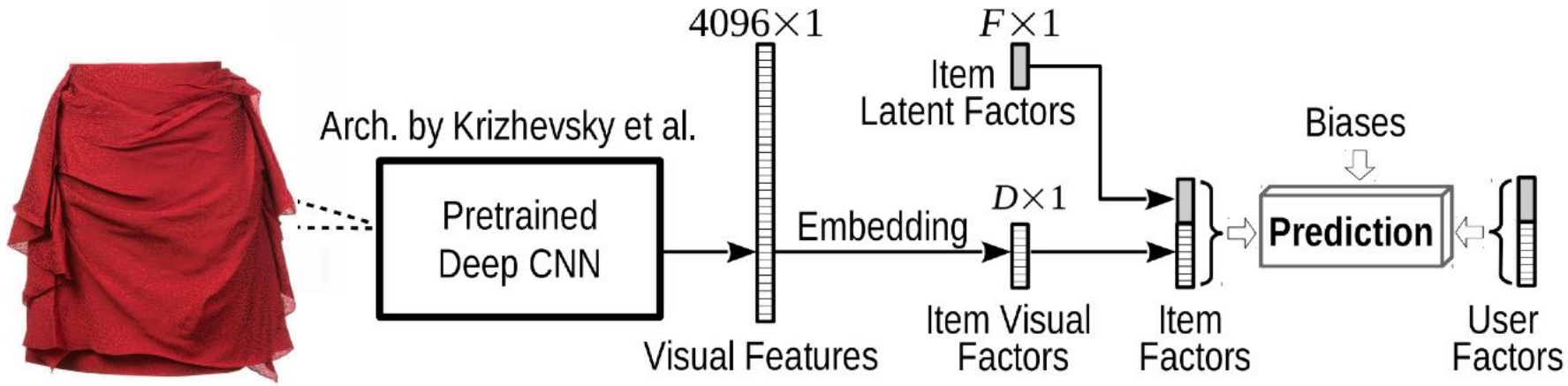
https://arxiv.org/pdf/1707.07435.pdf

Content-boosted CF

Добавим в профиль товара имбединг его контента!



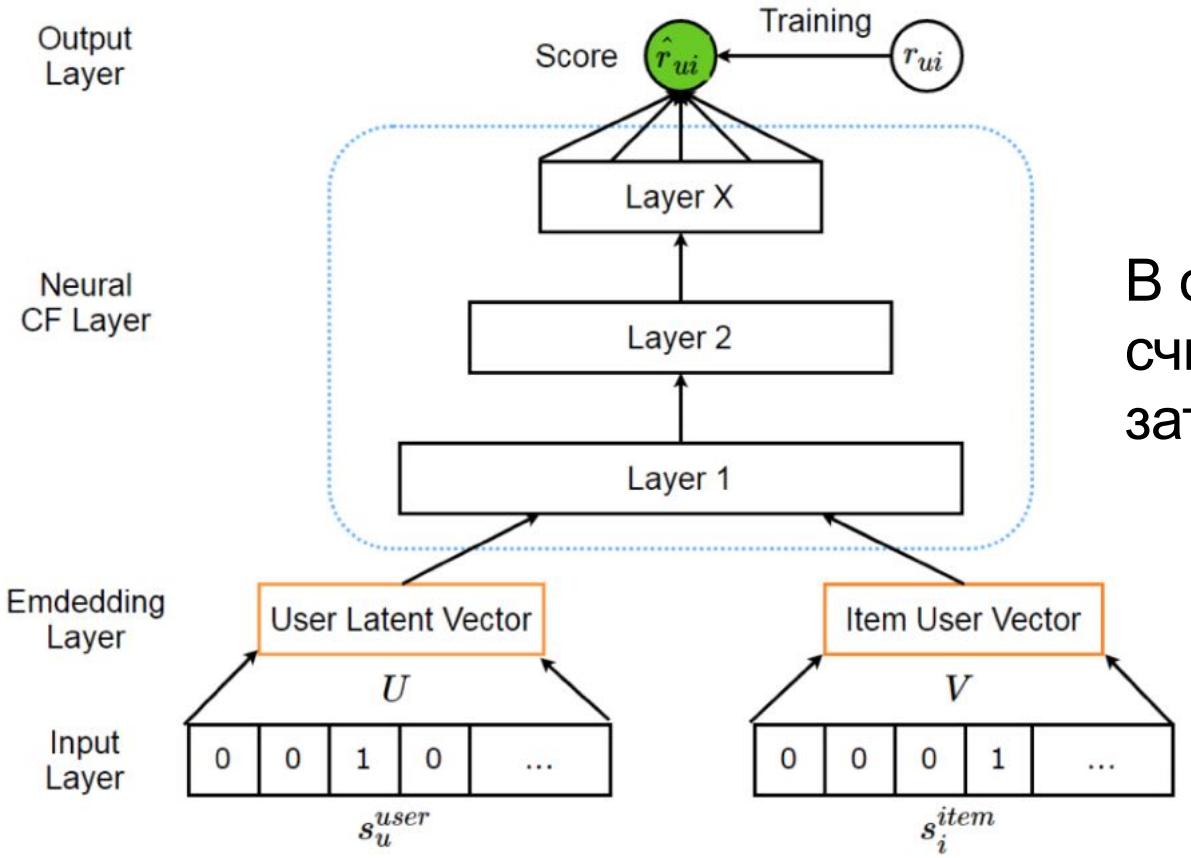
Content-boosted CF



Контент бывает разный, например, картинки

https://arxiv.org/pdf/1510.01784.pdf

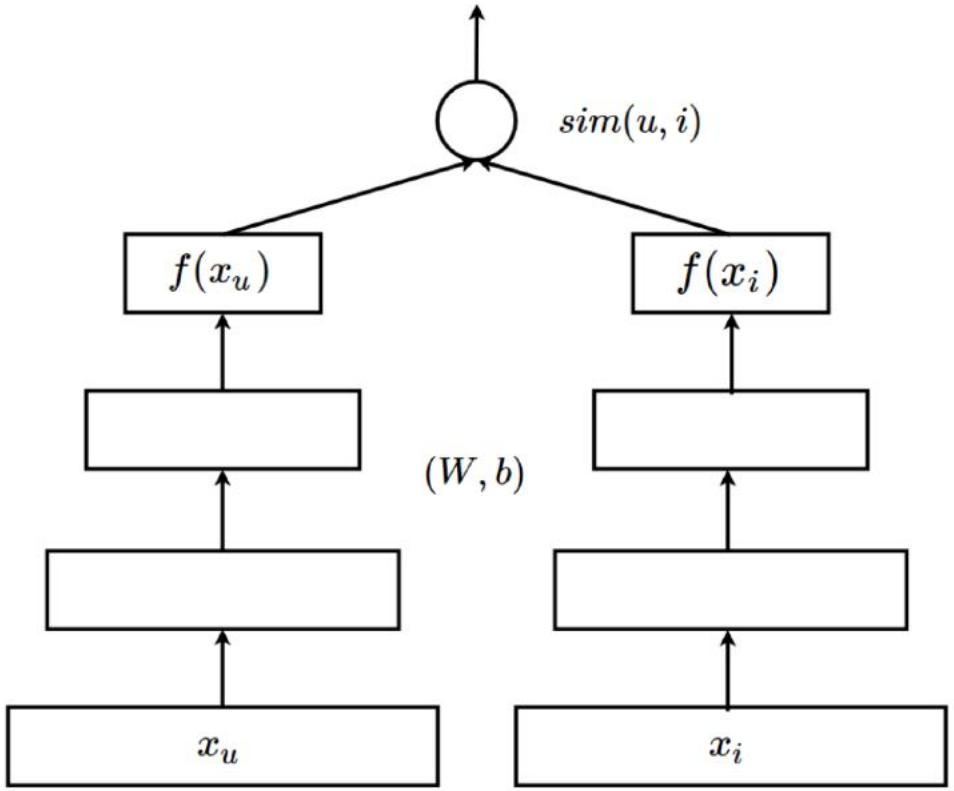
Neural Collaborative Filtering



В онлайне считать затратно

DSSM

Работает в онлайне!



Признаки пользователя

Признаки товара

Например: мешки слов

Удобно работать в DL фреймворке

Можно использовать крутые оптимайзеры (Adam, …)

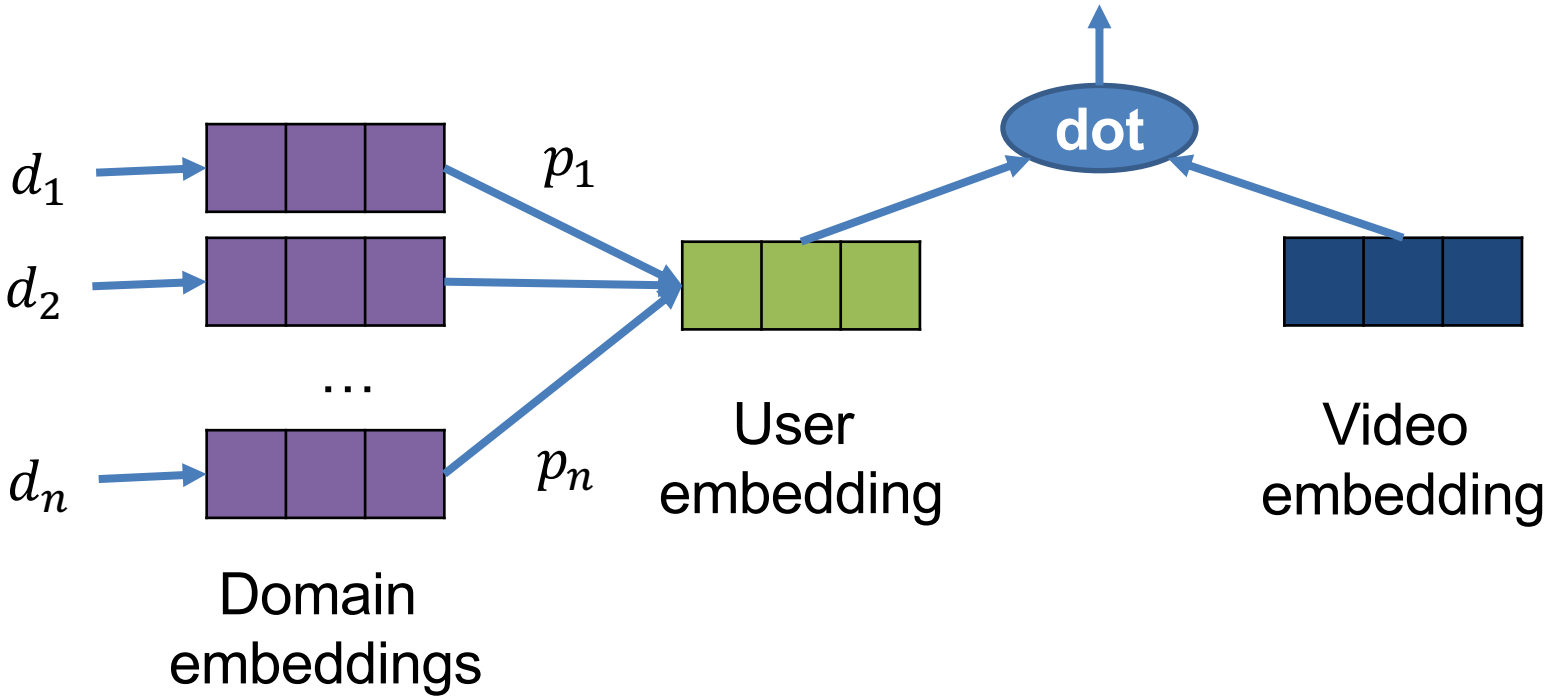
Можно имбеддить картинки, текст, …

Описываем какие взаимосвязи имеют смысл.

Детали выучит сеть.

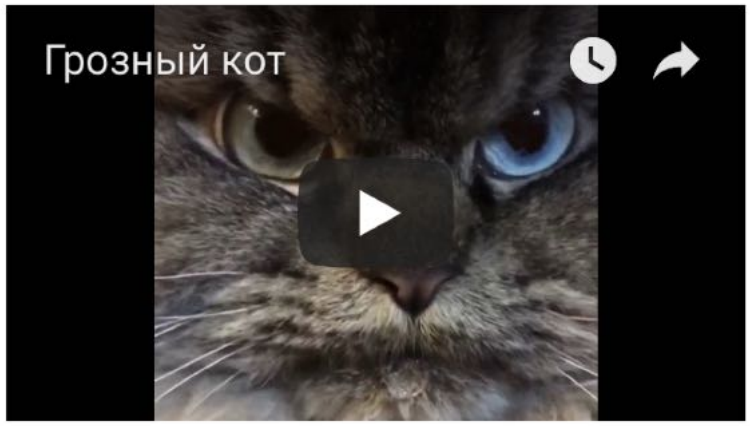
Реальная задача: домены → видео

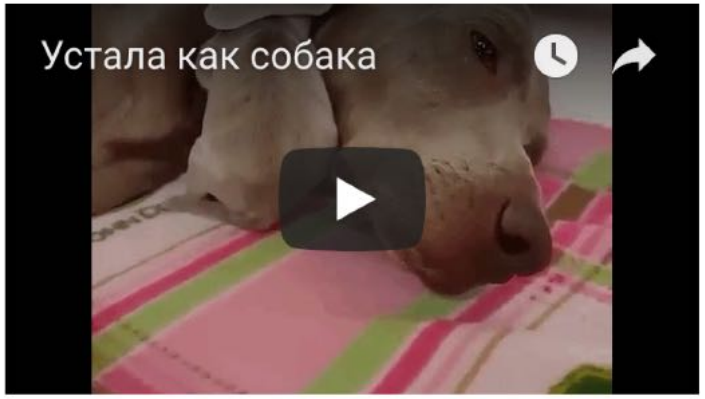
Хотим помочь холодному старту рекомендаций видео

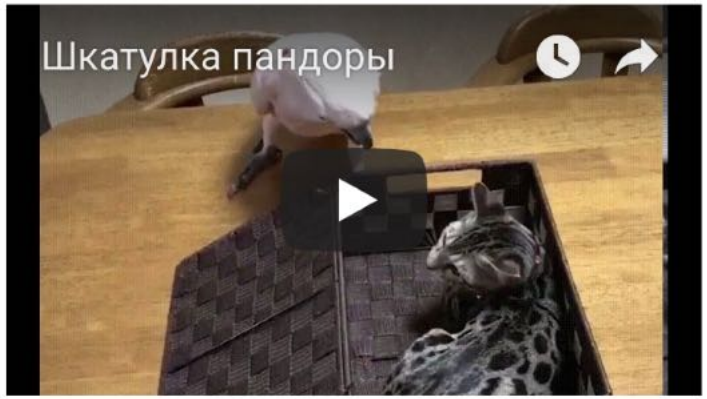


Знаем куда ходит пользователь

Похожие видео (по dot)

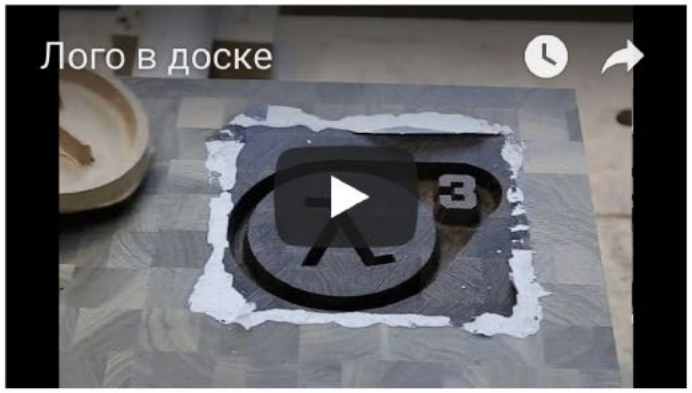






Самые похожие на **hi-tech.mail.ru**

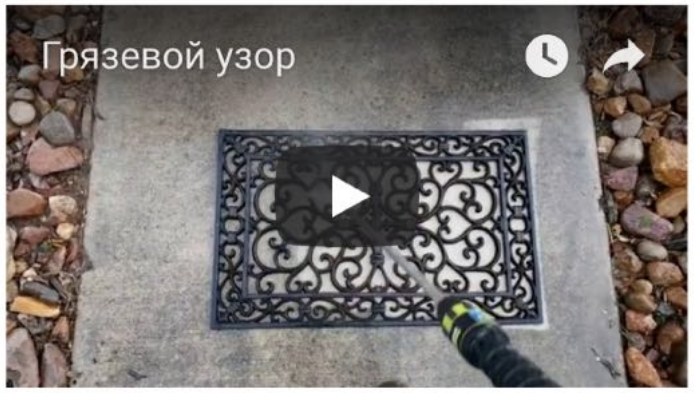
[0.94703776]



[0.92087275]

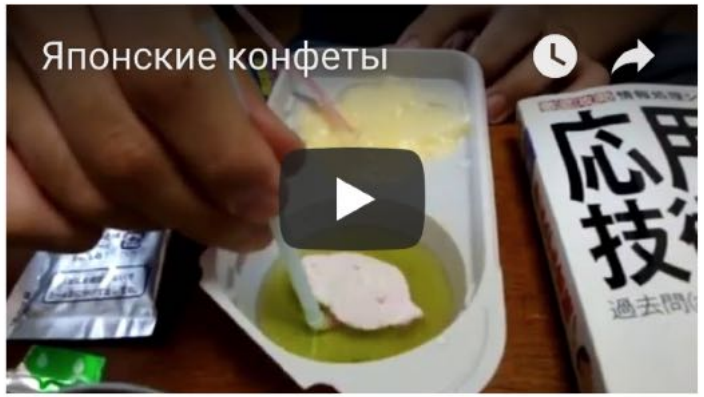
[0.9381]

[0.9200871]



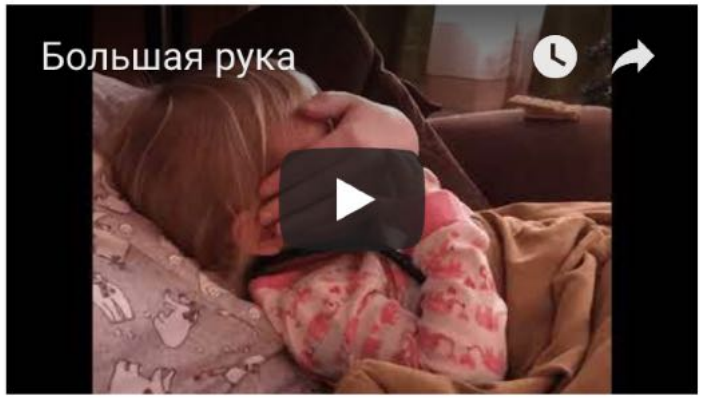
Самые похожие на **goodhouse.ru**

[0.89342886]

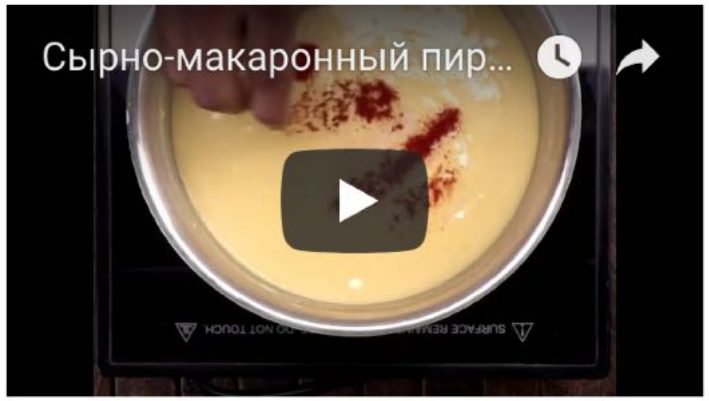


[0.8903276]

[0.89316946]



[0.88709587]



Спасибо за внимание!