

# Введение в рекомендательные системы

Владислав Гончаренко

Руководитель группы рекомендаций видео

Дзен

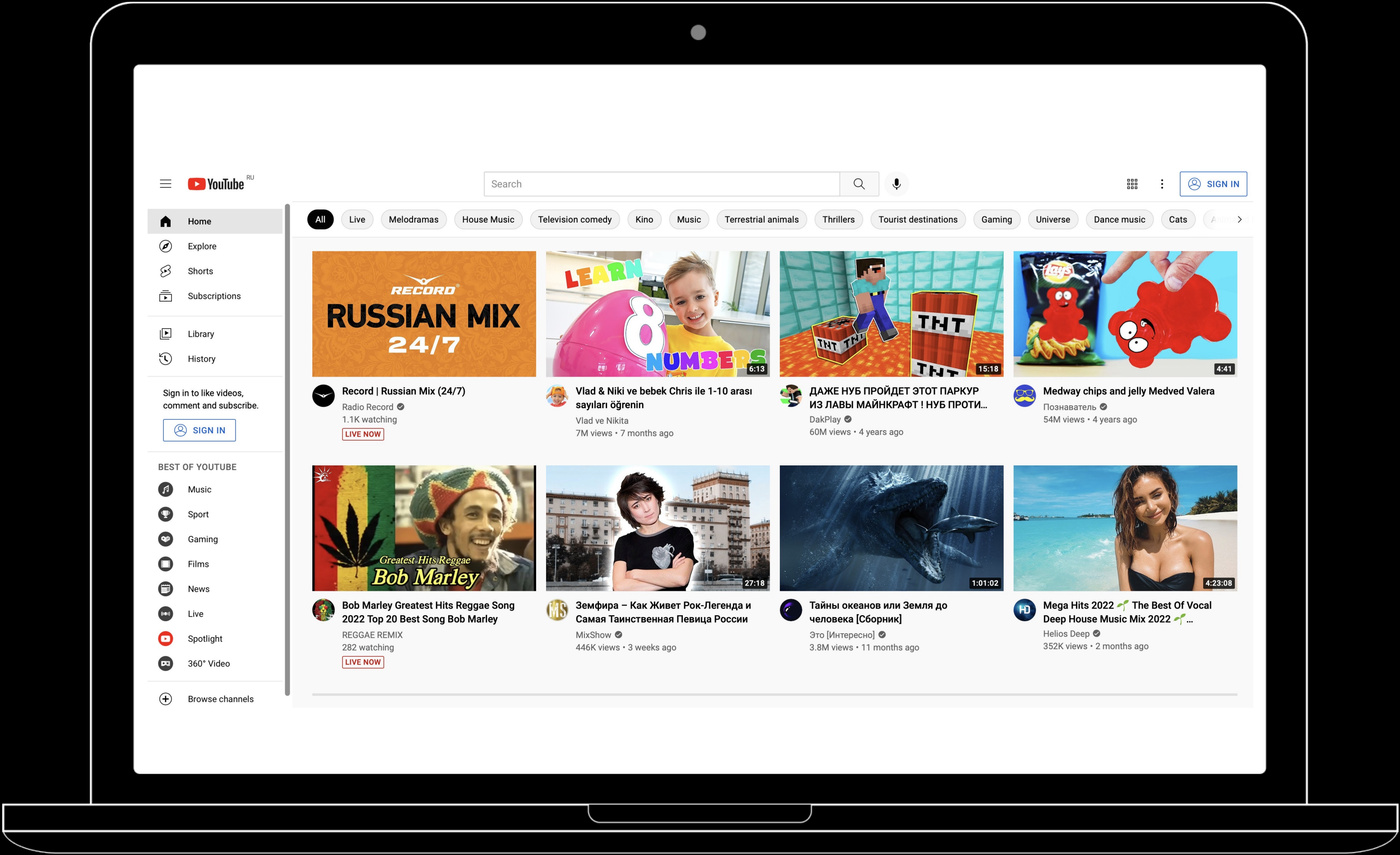
 v.goncharenko@vk.team

 @white\_pepper

ШАД, весна 2024

# Зачем нужны рекомендательные системы

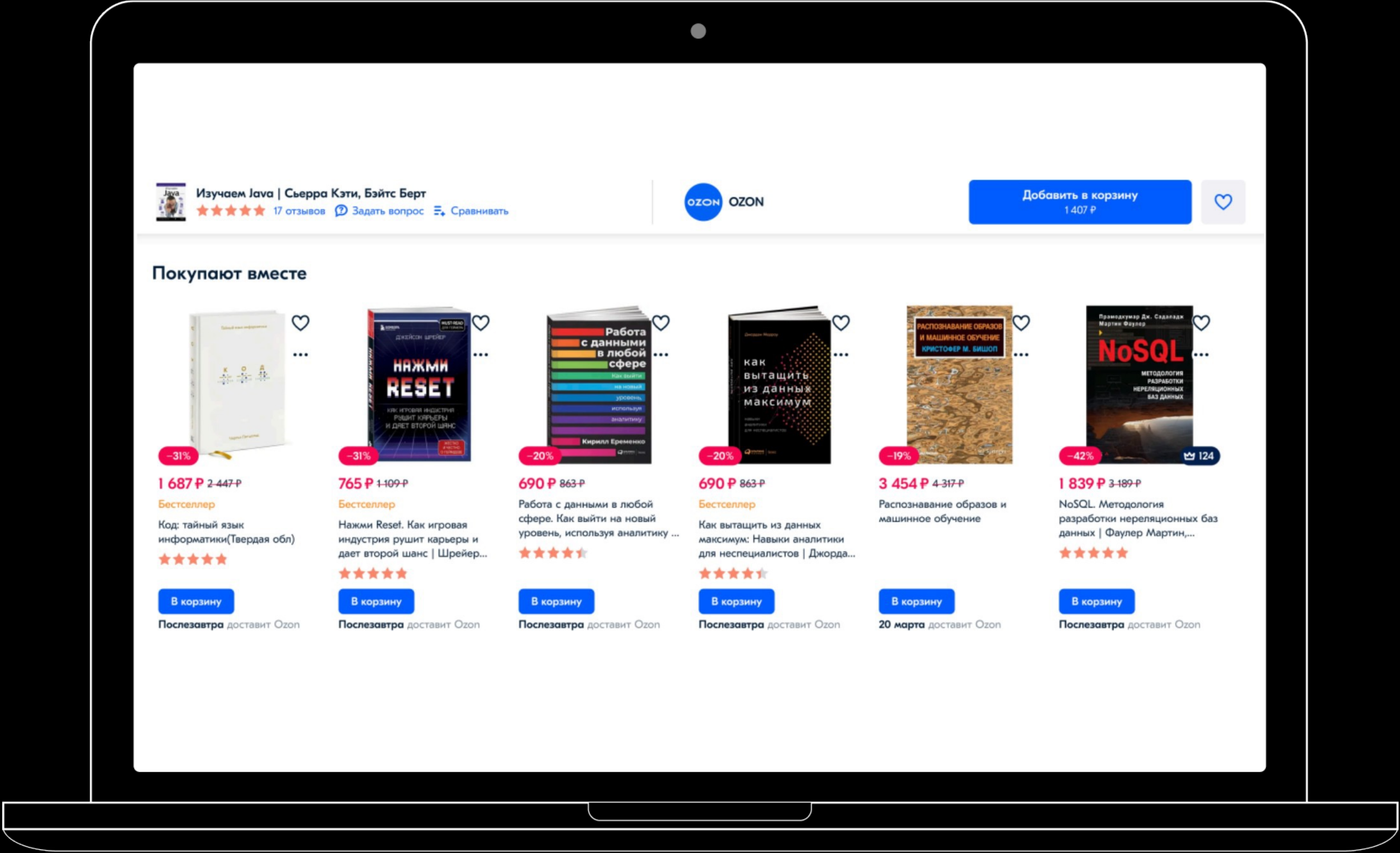
## Рекомендации видео





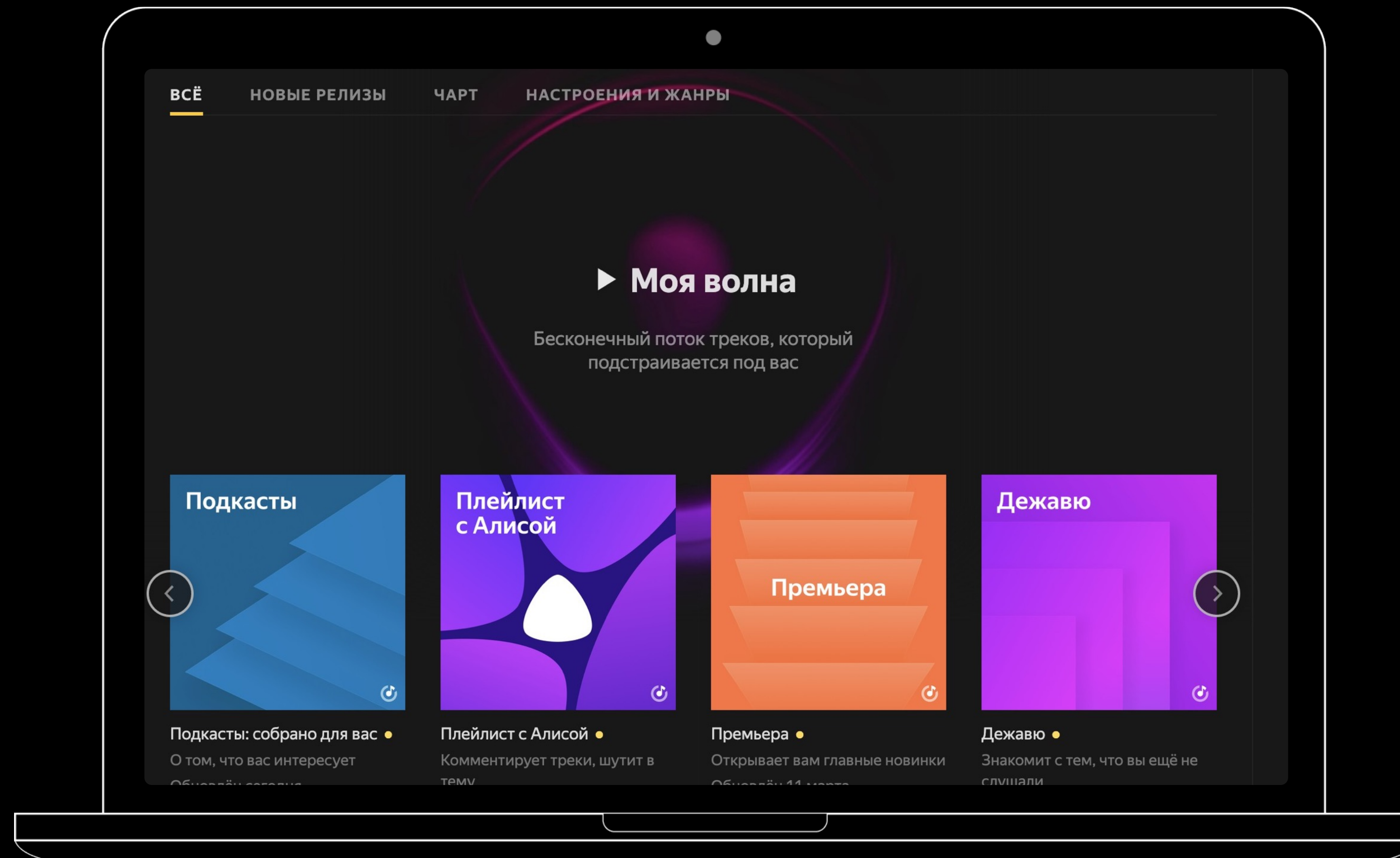
# Зачем нужны рекомендательные системы

Рекомендации товаров



# Зачем нужны рекомендательные системы

## Рекомендация музыки



# Зачем нужны рекомендательные системы

Как убедить человека остаться на видеосервисе?

✦ Предложить посмотреть что-нибудь еще

# Зачем нужны рекомендательные системы

Как убедить человека остаться на видеосервисе?

- ✦ Предложить посмотреть что-нибудь еще
- ✦ Предложенные видео должны быть интересны пользователю

# Зачем нужны рекомендательные системы

Как убедить человека остаться на видеосервисе?

- ✦ Предложить посмотреть что-нибудь еще
- ✦ Предложенные видео должны быть интересны пользователю
- ✦ В предложениях должно быть разнообразие, чтобы они не были навязчивыми
- ✦ Etc

# Зачем нужны рекомендательные системы

Как убедить человека остаться на видеосервисе?

- ✦ Предложить посмотреть что-нибудь еще
- ✦ Предложенные видео должны быть интересны пользователю
- ✦ В предложениях должно быть разнообразие, чтобы они не были навязчивыми
- ✦ Etc

Этим и занимаются рекомендательные системы



# Netflix Prize

2 октября 2006 г. началось соревнование по рекомендации фильмов от Netflix, на тот момент популярной компании по аренде DVD

NETFLIX

Netflix Prize

- ✦ Нужно было предсказывать оценки фильмам от 1 до 5
- ✦ Необходимо было уменьшить ошибку RMSE
- ✦ 17700 фильмов,  $\approx 480$  тыс клиентов и  $\approx 100$  млн оценок

# Netflix Prize

2 октября 2006 г. началось соревнование по рекомендации фильмов от Netflix, на тот момент популярной компании по аренде DVD

NETFLIX

**Netflix Prize**

- ✦ Нужно было предсказывать оценки фильмам от 1 до 5
- ✦ Необходимо было уменьшить ошибку RMSE
- ✦ 17700 фильмов,  $\approx 480$  тыс клиентов и  $\approx 100$  млн оценок
- ✦ Главный приз \$1 000 000
- ✦ Соревнование длилось почти 3 года
- ✦ 5100 команд-участниц

# Netflix Prize

Итоговые результаты:



Результаты на трейне

Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	<a href="#">The Ensemble</a>	0.8553	10.10	2009-07-26 18:38:22
2	<a href="#">BellKor's Pragmatic Chaos</a>	0.8554	10.09	2009-07-26 18:18:28

# Netflix Prize

Итоговые результаты:



Результаты на трейне

Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	<a href="#">The Ensemble</a>	0.8553	10.10	2009-07-26 18:38:22
2	<a href="#">BellKor's Pragmatic Chaos</a>	0.8554	10.09	2009-07-26 18:18:28



Результаты на тесте

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
1	<a href="#">BellKor's Pragmatic Chaos</a>	0.8567	10.06	2009-07-26 18:18:28
2	<a href="#">The Ensemble</a>	0.8567	10.06	2009-07-26 18:38:22

# Netflix Prize

Итоговые результаты:



Результаты на трейне

Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	<a href="#">The Ensemble</a>	0.8553	10.10	2009-07-26 18:38:22
2	<a href="#">BellKor's Pragmatic Chaos</a>	0.8554	10.09	2009-07-26 18:18:28



Результаты на тесте

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
1	<a href="#">BellKor's Pragmatic Chaos</a>	0.8567	10.06	2009-07-26 18:18:28
2	<a href="#">The Ensemble</a>	0.8567	10.06	2009-07-26 18:38:22

Соревнование дало толчок к развитию области  
рекомендательных систем и во многом определило вектор развития



# Структура курса

Всего в курсе 6 лекций

1. Введение, общая постановка задачи, основные концепции
2. Методы на основе матричных факторизаций, выдача ДЗ 1
3. Нейросетевые и content-based рекомендации, выдача ДЗ 2
4. Оценка качества рекомендаций
5. Приемы для применения в production, выдача ДЗ 3
6. Case study

# Структура курса

Всего в курсе 6 лекций

1. Введение, общая постановка задачи, основные концепции
2. Методы на основе матричных факторизаций, выдача ДЗ 1
3. Нейросетевые и content-based рекомендации, выдача ДЗ 2
4. Оценка качества рекомендаций
5. Приемы для применения в production, выдача ДЗ 3
6. Case study

В курсе будет 3 домашки, на каждую дается 2 недели

# План лекции

1. Постановка задачи рекомендательных систем
  - Основные понятия
  - Виды фидбека
2. Коллаборативная фильтрация
3. Другие типы рекомендаций
  - Контентные модели
  - Гибридные модели
4. Рекомендательные системы на практике
  - Ранжирующая модель
  - Отбор кандидатов
  - Реранкинг
5. Особенности рекомендательных систем

# План лекции

1. Постановка задачи рекомендательных систем
  - Основные понятия
  - Виды фидбека
2. Коллаборативная фильтрация
3. Другие типы рекомендаций
  - Контентные модели
  - Гибридные модели
4. Рекомендательные системы на практике
  - Ранжирующая модель
  - Отбор кандидатов
  - Реранкинг
5. Особенности рекомендательных систем

# Постановка задачи рекомендательных систем

- ✦ Заданы  $U$  — множество пользователей,  
 $I$  — множество объектов (айтемов),  
которые мы рекомендуем
- ✦ Для каждого пользователя  $u \in U$  задана его  
история взаимодействий: он взаимодействовал  
с айтемами  $I_u \subset I$ , которым поставил рейтинги  
$$R_u = (r_{ui})_{i \in I_u}$$
- ✦ Такие рейтинги пользователя называют **фидбеком**



# Постановка задачи рекомендательных систем

- ✦ Заданы  $U$  — множество пользователей,  
 $I$  — множество объектов (айтемов),  
которые мы рекомендуем
- ✦ Для каждого пользователя  $u \in U$  задана его  
история взаимодействий: он взаимодействовал  
с айтемами  $I_u \subset I$ , которым поставил рейтинги  
$$R_u = (r_{ui})_{i \in I_u}$$
- ✦ Такие рейтинги пользователя называют **фидбеком**

Цель — предложить пользователю новые для него айтемы, которые были бы ему интересны

Как понять, что айтем  
интересен пользователю?

# Постановка задачи рекомендательных систем

Как понять, что айтем интересен пользователю?

Мы можем это оценить по качеству взаимодействия. Например:

# Постановка задачи рекомендательных систем

Как понять, что айтем интересен пользователю?

Мы можем это оценить по качеству взаимодействия. Например:

✦ Для товара, что его положили в корзину

# Постановка задачи рекомендательных систем

Как понять, что айтем интересен пользователю?

Мы можем это оценить по качеству взаимодействия. Например:

- ✦ Для товара, что его положили в корзину
- ✦ Для музыки, что ее дослушали до конца



# Постановка задачи рекомендательных систем

Как понять, что айтем интересен пользователю?

Мы можем это оценить по качеству взаимодействия. Например:

- ✦ Для товара, что его положили в корзину
- ✦ Для музыки, что ее дослушали до конца
- ✦ Для статьи, что ее лайкнули

# Постановка задачи рекомендательных систем

Как понять, что айтем интересен пользователю?

Мы можем это оценить по качеству взаимодействия. Например:

- ✦ Для товара, что его положили в корзину
- ✦ Для музыки, что ее дослушали до конца
- ✦ Для статьи, что ее лайкнули
- ✦ Для видео, что посмотрели хотя бы половину
- ✦ Etc

# Explicit feedback

Explicit feedback — это такие действия, которые выражают явное отношение пользователя к товару.

# Explicit feedback

Explicit feedback — это такие действия, которые выражают явное отношение пользователя к товару.

Например:



рейтинг фильма  
от 1 до 5



лайк/дизлайк  
треку



рецензия на товар

# Implicit feedback

Implicit feedback – это любая другая информация, действия пользователя на сайте. Он не отражают его явного отношения к товару, но может выступать в качестве прокси к explicit feedback-у.



# Implicit feedback

Implicit feedback – это любая другая информация, действия пользователя на сайте. Он не отражают его явного отношения к товару, но может выступать в качестве прокси к explicit feedback-у.

Например:



покупка товара



просмотр статьи



время просмотра  
видео

# Особенности explicit feedback

- ✦ Как правило, таких данных довольно мало и их сложнее получать

# Особенности explicit feedback

- ✦ Как правило, таких данных довольно мало и их сложнее получать
- ✦ Дизлайк статьи не означает, что подобные статьи не надо рекомендовать пользователю. Возможно, ему интересна тематика, но не понравилась конкретно эта статья

# Особенности implicit feedback

✦ Таких данных больше

# Особенности implicit feedback

- ✦ Таких данных больше
- ✦ Как правило, неявным данным можно доверять меньше, поскольку они выражают лишь неявные сигналы, а не явные оценки пользователя. Например, пользователь мог купить товар в подарок, но ему самому товар не нравится

# Особенности implicit feedback

- ✦ Таких данных больше
- ✦ Как правило, неявным данным можно доверять меньше, поскольку они выражают лишь неявные сигналы, а не явные оценки пользователя. Например, пользователь мог купить товар в подарок, но ему самому товар не нравится
- ✦ Так как неявный сигнал имеет иную природу, для его оптимизации в рекомендательной системе обычно используют техники отличные от explicit feedback-а

# План лекции

1. Постановка задачи рекомендательных систем
  - Основные понятия
  - Виды фидбека
2. Коллаборативная фильтрация
3. Другие типы рекомендаций
  - Контентные модели
  - Гибридные модели
4. Рекомендательные системы на практике
  - Ранжирующая модель
  - Отбор кандидатов
  - Реранкинг
5. Особенности рекомендательных систем

# Пример

Рассмотрим пример. Зеленые клетки — лайки, красные — дизлайки

Item \ User	1	2	3	4	5	6	7	8	9	10
Петя	<div></div>		<div></div>	<div></div>	<div></div>			<div></div>		
Маша		<div></div>		<div></div>						
Вася	<div></div>		<div></div>		<div></div>		<div></div>			<div></div>
Катя	<div></div>			<div></div>	<div></div>			<div></div>		<div></div>

Что стоит рекомендовать Кате?



# Пример

Рассмотрим пример. Зеленые клетки — лайки, красные — дизлайки

Item \ User	1	2	3	4	5	6	7	8	9	10
Петя	<div></div>		<div></div>	<div></div>	<div></div>			<div></div>		
Маша		<div></div>		<div></div>						
Вася	<div></div>		<div></div>		<div></div>		<div></div>			<div></div>
Катя	<div></div>			<div></div>	<div></div>			<div></div>		<div></div>

Что стоит рекомендовать Кате?

- 1. Петя и Катя похожи по взаимодействиям, поэтому Кате стоит порекомендовать 3

# Пример

Рассмотрим пример. Зеленые клетки — лайки, красные — дизлайки

Item \ User	1	2	3	4	5	6	7	8	9	10
Петя	<div></div>		<div></div>	<div></div>	<div></div>			<div></div>		
Маша		<div></div>		<div></div>						
Вася	<div></div>		<div></div>		<div></div>		<div></div>			<div></div>
Катя	<div></div>			<div></div>	<div></div>			<div></div>		<div></div>

Что стоит рекомендовать Кате?

1. Петя и Катя похожи по взаимодействиям, поэтому Кате стоит порекомендовать 3

А что лучше не показывать Пете?

# Пример

Рассмотрим пример. Зеленые клетки — лайки, красные — дизлайки

Item \ User	1	2	3	4	5	6	7	8	9	10
Петя	<div></div>		<div></div>	<div></div>	<div></div>			<div></div>		
Маша		<div></div>		<div></div>						
Вася	<div></div>		<div></div>		<div></div>		<div></div>			<div></div>
Катя	<div></div>			<div></div>	<div></div>			<div></div>		<div></div>

Что стоит рекомендовать Кате?

1. Петя и Катя похожи по взаимодействиям, поэтому Кате стоит порекомендовать 3

А что лучше не показывать Пете?

2. Пете не стоит рекомендовать 10, так как Кате он не понравился

# Коллаборативная фильтрация

Идея

Хотим рекомендовать пользователю товары,  
понравившиеся похожим на него пользователям

# Коллаборативная фильтрация

## Идея

Хотим рекомендовать пользователю товары, понравившиеся похожим на него пользователям

Можно действовать по-другому. А именно, рекомендовать айтемы, похожие на понравившиеся пользователю. Похожести между айтемами можно выявлять по пользовательским взаимодействиям

# Коллаборативная фильтрация

## Идея

Хотим рекомендовать пользователю товары, понравившиеся похожим на него пользователям

Можно действовать по-другому. А именно, рекомендовать айтемы, похожие на понравившиеся пользователю. Похожести между айтемами можно выявлять по пользовательским взаимодействиям

## Коллаборативная фильтрация

В общем смысле, коллаборативная фильтрация — это семейство методов рекомендаций, основанных на похожестях по истории взаимодействия между пользователями и товарами

# User2User рекомендации

- ✦ Рассмотрим некоторую меру похожести  $s(u, v)$  между пользователями  
Тогда можно считать соседями пользователя  $u$  пользователей

$$N(u) = \{v \in U \setminus \{u\} \mid s(u, v) > \alpha\}$$

# User2User рекомендации

- ✦ Рассмотрим некоторую меру похожести  $s(u, v)$  между пользователями  
Тогда можно считать соседями пользователя  $u$  пользователей

$$N(u) = \{v \in U \setminus \{u\} \mid s(u, v) > \alpha\}$$

- ✦ Пользователю  $u$  хотим предложить айтемы, которые понравились похожим на него пользователям. Оценим рейтинг пользователя по рейтингам соседей



# User2User рекомендации

- ✦ Рассмотрим некоторую меру похожести  $s(u, v)$  между пользователями  
Тогда можно считать соседями пользователя  $u$  пользователей

$$N(u) = \{v \in U \setminus \{u\} \mid s(u, v) > \alpha\}$$

- ✦ Пользователю  $u$  хотим предложить айтемы, которые понравились похожим на него пользователям. Оценим рейтинг пользователя по рейтингам соседей
- ✦ Оценив рейтинги непросмотренных айтемов, порекомендовать некоторое количество айтемов с максимальной оценкой

# User2User рекомендации

- ✦ Рассмотрим некоторую меру похожести  $s(u, v)$  между пользователями  
Тогда можно считать соседями пользователя  $u$  пользователей

$$N(u) = \{v \in U \setminus \{u\} \mid s(u, v) > \alpha\}$$

- ✦ Пользователю  $u$  хотим предложить айтемы, которые понравились похожим на него пользователям. Оценим рейтинг пользователя по рейтингам соседей
- ✦ Оценив рейтинги непросмотренных айтемов, порекомендовать некоторое количество айтемов с максимальной оценкой
- ✦ Обозначим  
 $I_u$  — множество айтемов, которые оценил  $u$   
 $\bar{r}_u$  — средний рейтинг пользователя  $u$

# User2User рекомендации

- ✦ Оценка рейтинга, как среднее по соседям с весами

$$\hat{r}_{ui} = \frac{\sum_{v \in N(u)} s(u, v) r_{vi}}{\sum_{v \in N(u)} |s(u, v)|}$$

# User2User рекомендации

- ✦ Оценка рейтинга, как среднее по соседям с весами

$$\hat{r}_{ui} = \frac{\sum_{v \in N(u)} s(u, v) r_{vi}}{\sum_{v \in N(u)} |s(u, v)|}$$

- ✦ Проблема: разные пользователи ставят рейтинги в разной шкале, поэтому сделаем поправку на средний рейтинг пользователя

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v \in N(u)} s(u, v) (r_{vi} - \bar{r}_v)}{\sum_{v \in N(u)} |s(u, v)|}$$

# User2User рекомендации

- ✦ Оценка рейтинга, как среднее по соседям с весами

$$\hat{r}_{ui} = \frac{\sum_{v \in N(u)} s(u, v) r_{vi}}{\sum_{v \in N(u)} |s(u, v)|}$$

- ✦ Проблема: разные пользователи ставят рейтинги в разной шкале, поэтому сделаем поправку на средний рейтинг пользователя

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v \in N(u)} s(u, v) (r_{vi} - \bar{r}_v)}{\sum_{v \in N(u)} |s(u, v)|}$$

- ✦ Кроме того, оценки от разных пользователей могут быть в разном масштабе, поэтому можем их отнормировать

$$\hat{r}_{ui} = \bar{r}_u + \sigma_u \frac{\sum_{v \in N(u)} s(u, v) (r_{vi} - \bar{r}_v) / \sigma_v}{\sum_{v \in N(u)} |s(u, v)|}$$

$$\sigma_u = \sqrt{\frac{1}{|I_u|} \sum_{i \in I_u} (r_{ui} - \bar{r}_u)^2}$$

# User2User рекомендации

Как можно выбрать  $s(u, v)$ ?

# User2User рекомендации

Как можно выбрать  $s(u, v)$ ?

- ✦ Нормированное число  
общих товаров, мера Жаккара

$$s(u, v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|}$$

# User2User рекомендации

Как можно выбрать  $s(u, v)$ ?

- ✦ Нормированное число общих товаров, мера Жаккара

$$s(u, v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|}$$

- ✦ Скалярное произведение векторов общих рейтингов

$$s(u, v) = \sum_{i \in I_u \cap I_v} r_{ui} r_{vi}$$



# User2User рекомендации

Как можно выбрать  $s(u, v)$ ?

- ✦ Нормированное число общих товаров, мера Жаккара

$$s(u, v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|}$$

- ✦ Скалярное произведение векторов общих рейтингов

$$s(u, v) = \sum_{i \in I_u \cap I_v} r_{ui} r_{vi}$$

- ✦ Корреляция Пирсона между векторами общих рейтингов

$$s(u, v) = \frac{\sum_{i \in I_u \cap I_v} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_u \cap I_v} (r_{vi} - \bar{r}_v)^2}}$$

# User2User рекомендации

Как можно выбрать  $s(u, v)$ ?

- ✦ Нормированное число общих товаров, мера Жаккара

$$s(u, v) = \frac{|I_u \cap I_v|}{|I_u \cup I_v|}$$

- ✦ Скалярное произведение векторов общих рейтингов

$$s(u, v) = \sum_{i \in I_u \cap I_v} r_{ui} r_{vi}$$

- ✦ Корреляция Пирсона между векторами общих рейтингов

$$s(u, v) = \frac{\sum_{i \in I_u \cap I_v} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_u \cap I_v} (r_{vi} - \bar{r}_v)^2}}$$

- ✦ Корреляция Пирсона между векторами общих рейтингов с поправкой в случае малого числа общих рейтингов

$$s(u, v) = \min\left(\frac{|I_u \cap I_v|}{50}, 1\right) \frac{\sum_{i \in I_u \cap I_v} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_u \cap I_v} (r_{vi} - \bar{r}_v)^2}}$$

# Item2Item рекомендации

- ✦ Идея: к оцененным пользователем товарам найдем наиболее похожие на них и порекомендуем

# Item2Item рекомендации

- ✦ Идея: к оцененным пользователем товарам найдем наиболее похожие на них и порекомендуем
- ✦ Похожесть между айтемами определяется, как косинусная мера с поправкой на средний рейтинг пользователей, где  $U_i$  — пользователи, поставившие рейтинг айтему  $i$

$$s(i, j) = \frac{\sum_{u \in U_i \cap U_j} (r_{ui} - \bar{r}_u)(r_{uj} - \bar{r}_u)}{\sqrt{\sum_{u \in U_i \cap U_j} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{u \in U_i \cap U_j} (r_{uj} - \bar{r}_u)^2}}$$

# Item2Item рекомендации

- ✦ Идея: к оцененным пользователем товарам найдем наиболее похожие на них и порекомендуем
- ✦ Похожесть между айтемами определяется, как косинусная мера с поправкой на средний рейтинг пользователей, где  $U_i$  — пользователи, поставившие рейтинг айтему  $i$

$$s(i, j) = \frac{\sum_{u \in U_i \cap U_j} (r_{ui} - \bar{r}_u)(r_{uj} - \bar{r}_u)}{\sqrt{\sum_{u \in U_i \cap U_j} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{u \in U_i \cap U_j} (r_{uj} - \bar{r}_u)^2}}$$

- ✦ Далее, используя похожести и историю взаимодействия пользователя, аналогично оцениваем рейтинги неоцененных айтемов

# Особенности коллаборативной фильтрации

- ✦ Не опирается на дополнительную информацию про пользователей и товары, методы коллаборативной фильтрации самодостаточны

# Особенности коллаборативной фильтрации



Не опирается на дополнительную информацию про пользователей и товары, методы коллаборативной фильтрации самодостаточны



Неприменимы для холодных пользователей и товаров, то есть для имеющих маленькую историю взаимодействий

# Особенности коллаборативной фильтрации

✦ Не опирается на дополнительную информацию про пользователей и товары, методы коллаборативной фильтрации самодостаточны

✦ Неприменимы для холодных пользователей и товаров, то есть для имеющих маленькую историю взаимодействий

✦ Рекомендации основываются на истории взаимодействий в прошлом, которая зависит от прошлых рекомендаций пользователю



# Особенности коллаборативной фильтрации

✦ Не опирается на дополнительную информацию про пользователей и товары, методы коллаборативной фильтрации самодостаточны

✦ Рекомендации основываются на истории взаимодействий в прошлом, которая зависит от прошлых рекомендаций пользователю

✦ Неприменимы для холодных пользователей и товаров, то есть для имеющих маленькую историю взаимодействий

✦ Более продвинутые методы будут рассмотрены на лекции 2

# План лекции

1. Постановка задачи рекомендательных систем
  - Основные понятия
  - Виды фидбека
2. Коллаборативная фильтрация
3. Другие типы рекомендаций
  - Контентные модели
  - Гибридные модели
4. Рекомендательные системы на практике
  - Ранжирующая модель
  - Отбор кандидатов
  - Реранкинг
5. Особенности рекомендательных систем

# Content-based рекомендации

Вернемся к примеру и посмотрим на сами айтемы

User \ Item										
Петя										
Маша										
Вася										
Катя										

Что стоит рекомендовать Пете?

# Content-based рекомендации

Вернемся к примеру и посмотрим на сами айтемы

User \ Item										
Петя										
Маша										
Вася										
Катя										

Что стоит рекомендовать Пете?

- 1. У Пети ярко выраженная любовь к кошкам, можно показать еще непоказанную кошку



# Content-based рекомендации

Вернемся к примеру и посмотрим на сами айтемы

User \ Item										
Петя										
Маша										
Вася										
Катя										

Что стоит рекомендовать Пете?

1. У Пети ярко выраженная любовь к кошкам, можно показать еще непоказанную кошку

А что делать с Катей?

# Content-based рекомендации

Вернемся к примеру и посмотрим на сами айтемы

User \ Item										
Петя										
Маша										
Вася										
Катя										

Что стоит рекомендовать Пете?

1. У Пети ярко выраженная любовь к кошкам, можно показать еще непоказанную кошку

А что делать с Катей?

2. У Кати всего одно взаимодействие с машинами, этого мало, поэтому показываем только кошек, с которыми было 2 лайка

# Контентный подход

закljučаются в рекомендации пользователям  
таких айтемов, которые по содержанию похожи  
на понравившиеся им

# Пример content-based рекомендаций

Пусть каждому айтому  $i$  соответствует эмбединг  $e_i$ , полученный по его содержанию.



# Пример content-based рекомендаций

Пусть каждому айтому  $i$  соответствует эмбединг  $e_i$ , полученный по его содержанию.

Тогда пользователю  $u$  можно рекомендовать айтемы по мере близости  $\rho$  между векторами

$$\hat{r}_{ui} = \max_{j \in I_u, r_{uj} > \alpha} \rho(e_i, e_j) r_{uj}$$

# Пример content-based рекомендаций

Пусть каждому айтому  $i$  соответствует эмбединг  $e_i$ , полученный по его содержанию.

Тогда пользователю  $u$  можно рекомендовать айтемы по мере близости  $\rho$  между векторами

$$\hat{r}_{ui} = \max_{j \in I_u, r_{uj} > \alpha} \rho(e_i, e_j) r_{uj}$$

В качестве  $\rho$  может выступать

✦ Скалярное произведение

$$\rho(e_i, e_j) = \sum_{k=1}^d e_{ik} e_{jk} = \langle e_i, e_j \rangle$$

# Пример content-based рекомендаций

Пусть каждому айтому  $i$  соответствует эмбединг  $e_i$ , полученный по его содержанию.

Тогда пользователю  $u$  можно рекомендовать айтемы по мере близости  $\rho$  между векторами

$$\hat{r}_{ui} = \max_{j \in I_u, r_{uj} > \alpha} \rho(e_i, e_j) r_{uj}$$

В качестве  $\rho$  может выступать

✦ Скалярное произведение

$$\rho(e_i, e_j) = \sum_{k=1}^d e_{ik} e_{jk} = \langle e_i, e_j \rangle$$

✦ Косинусное расстояние

$$\rho(e_i, e_j) = \frac{\langle e_i, e_j \rangle}{\|e_i\| \cdot \|e_j\|}$$

✦ Etc

# Пример content-based рекомендаций

Пусть каждому айтому  $i$  соответствует эмбединг  $e_i$ , полученный по его содержанию.

Тогда пользователю  $u$  можно рекомендовать айтемы по мере близости  $\rho$  между векторами

$$\hat{r}_{ui} = \max_{j \in I_u, r_{uj} > \alpha} \rho(e_i, e_j) r_{uj}$$

В качестве  $\rho$  может выступать

✦ Скалярное произведение

$$\rho(e_i, e_j) = \sum_{k=1}^d e_{ik} e_{jk} = \langle e_i, e_j \rangle$$

✦ Косинусное расстояние

$$\rho(e_i, e_j) = \frac{\langle e_i, e_j \rangle}{\|e_i\| \cdot \|e_j\|}$$

✦ Etc

Различные методы будут рассмотрены на лекции 3

# Гибридные модели

Под гибридными моделями понимают модели рекомендательных систем, которые используют одновременно и коллаборативную и контентную информацию для построения пользовательских рекомендаций

# План лекции

1. Постановка задачи рекомендательных систем
  - Основные понятия
  - Виды фидбека
2. Коллаборативная фильтрация
3. Другие типы рекомендаций
  - Контентные модели
  - Гибридные модели
4. Рекомендательные системы на практике
  - Ранжирующая модель
  - Отбор кандидатов
  - Реранкинг
5. Особенности рекомендательных систем

# Ранжирующая модель

# Ранжирующая модель

- 1 Коллаборативные  
и контентные модели  
имеют свои достоинства  
и недостатки



# Ранжирующая модель

1

Коллаборативные  
и контентные модели  
имеют свои достоинства  
и недостатки

2

При рекомендациях  
хочется использовать  
контекстную информацию.

Например, предпочтения  
пользователя в разные дни  
недели могут различаться

# Ранжирующая модель

1

Коллаборативные и контентные модели имеют свои достоинства и недостатки

2

При рекомендациях хочется использовать контекстную информацию.

Например, предпочтения пользователя в разные дни недели могут различаться

3

Полезной была бы информация про товар: бренд, цена и т. д.

# Ранжирующая модель

1

Коллаборативные и контентные модели имеют свои достоинства и недостатки

2

При рекомендациях хочется использовать контекстную информацию.

Например, предпочтения пользователя в разные дни недели могут различаться

3

Полезной была бы информация про товар: бренд, цена и т. д.

На практике, как правило, используют ранжирующую модель, которая комбинирует различные признаки

# Ранжирующая модель: признаки

Признаками такой модели могут быть

# Ранжирующая модель: признаки

Признаками такой модели могут быть

- 1 Предсказания базовых  
коллаборативных,  
контентных моделей

# Ранжирующая модель: признаки

Признаками такой модели могут быть

- 1 Предсказания базовых коллаборативных, контентных моделей
- 2 Признаки пользователя: пол, возраст, характеристики истории взаимодействий и т. д.

# Ранжирующая модель: признаки

Признаками такой модели могут быть

1

Предсказания базовых  
коллаборативных,  
контентных моделей

2

Признаки пользователя:  
пол, возраст, характеристики  
истории взаимодействий и т. д.

3

Признаки объекта:  
цена, вес, жанр, характеристики  
истории взаимодействий и т. д.

# Ранжирующая модель: признаки

Признаками такой модели могут быть

1

Предсказания базовых  
коллаборативных,  
контентных моделей

2

Признаки пользователя:  
пол, возраст, характеристики  
истории взаимодействий и т. д.

3

Признаки объекта:  
цена, вес, жанр, характеристики  
истории взаимодействий и т. д.

4

Контекстная информация:  
день недели, погода,  
местоположение и т. д.



# Ранжирующая модель: устройство модели

Задачи, на которую такую модель можно обучать

# Ранжирующая модель: устройство модели

Задачи, на которую такую модель можно обучать

1

Бинарная классификация.  
Модель предсказывает  
наличие или отсутствие  
целевого положительного  
взаимодействия

# Ранжирующая модель: устройство модели

Задачи, на которую такую модель можно обучать

- 1 Бинарная классификация. Модель предсказывает наличие или отсутствие целевого положительного взаимодействия
- 2 Регрессия. Например, модель предсказывает длительность просмотра видео, по которому определяется положительность взаимодействия

# Ранжирующая модель: устройство модели

Задачи, на которую такую модель можно обучать

1

Бинарная классификация. Модель предсказывает наличие или отсутствие целевого положительного взаимодействия

2

Регрессия. Например, модель предсказывает длительность просмотра видео, по которому определяется положительность взаимодействия

3

**Ранжирование.** Существуют pairwise и listwise подходы, при которых модель учится правильно упорядочивать для пользователя набор объектов

# Ранжирующая модель: устройство модели

Задачи, на которую такую модель можно обучать

- 1 Бинарная классификация. Модель предсказывает наличие или отсутствие целевого положительного взаимодействия
- 2 Регрессия. Например, модель предсказывает длительность просмотра видео, по которому определяется положительность взаимодействия

- 3 Ранжирование. Существуют pairwise и listwise подходы, при которых модель учится правильно упорядочивать для пользователя набор объектов

Как правило, в качестве модели используют вариации градиентного бустинга, поскольку на текущий момент они лучше справляются с задачами на табличных данных.

# Отбор кандидатов

Проблемы:

# Отбор кандидатов

Проблемы:

- ✦ В реальной рекомендательной системе может быть огромное количество айтеров, порядка сотен миллионов или миллиардов

# Отбор кандидатов

## Проблемы:

- ✦ В реальной рекомендательной системе может быть огромное количество айтемов, порядка сотен миллионов или миллиардов
- ✦ При каждом запросе пользователя на новые рекомендации необходимо провести все айтемы через всю систему



# Отбор кандидатов

## Проблемы:

- ✦ В реальной рекомендательной системе может быть огромное количество айтеров, порядка сотен миллионов или миллиардов
- ✦ При каждом запросе пользователя на новые рекомендации необходимо провести все айтеры через всю систему
- ✦ Есть сложные модели, которые невозможно применять к миллионам айтеров

# Отбор кандидатов

## Проблемы:

- ✦ В реальной рекомендательной системе может быть огромное количество айтемов, порядка сотен миллионов или миллиардов
- ✦ Есть сложные модели, которые невозможно применять к миллионам айтемов

- ✦ При каждом запросе пользователя на новые рекомендации необходимо провести все айтемы через всю систему

## ✦ Решение:

на начальном этапе отобрать относительно небольшое число кандидатов и последующие шаги применять только к ним.  
Отбор кандидатов — первый шаг в пайплайне рекомендаций

# Отбор кандидатов

При отборе кандидатов важно не испортить полноту подходящих товаров, чтобы вообще оставалось, из чего выбирать.

Возможные подходы:

# Отбор кандидатов

При отборе кандидатов важно не испортить полноту подходящих товаров, чтобы вообще оставалось, из чего выбирать.

Возможные подходы:

- ✦ Эвристические: самые популярные товары, самые популярные среди жителей того же города и т.д.

# Отбор кандидатов

При отборе кандидатов важно не испортить полноту подходящих товаров, чтобы вообще оставалось, из чего выбирать.

Возможные подходы:

- ✦ Эвристические: самые популярные товары, самые популярные среди жителей того же города и т.д.
- ✦ Коллаборативные: считаем item2item или user2user похожести

# Отбор кандидатов

При отборе кандидатов важно не испортить полноту подходящих товаров, чтобы вообще оставалось, из чего выбирать.

Возможные подходы:

- ✦ Эвристические: самые популярные товары, самые популярные среди жителей того же города и т.д.
- ✦ Коллаборативные: считаем item2item или user2user похожести
- ✦ Контентные похожести: ищем близких по эмбедингу; можно быстро выделять с помощью структур данных HNSW, FAISS и др.

# Отбор кандидатов

При отборе кандидатов важно не испортить полноту подходящих товаров, чтобы вообще оставалось, из чего выбирать.

Возможные подходы:

- ✦ Эвристические: самые популярные товары, самые популярные среди жителей того же города и т.д.
- ✦ Коллаборативные: считаем `item2item` или `user2user` похожести
- ✦ Контентные похожести: ищем близких по эмбедингу; можно быстро выделять с помощью структур данных HNSW, FAISS и др.
- ✦ Подходы, учитывающие бизнес-логики: свежее, новое

# Реранкинг

Кроме того, зачастую на практике необходимо учитывать различную бизнес-логику в рекомендуемых объектах



# Реранкинг

Кроме того, зачастую на практике необходимо учитывать различную бизнес-логику в рекомендуемых объектах

Например

- ✦ Ограничить количество старых или слишком длинных видео

# Реранкинг

Кроме того, зачастую на практике необходимо учитывать различную бизнес-логику в рекомендуемых объектах

Например

- ✦ Ограничить количество старых или слишком длинных видео
- ✦ Обеспечить разнообразие

# Реранкинг

Кроме того, зачастую на практике необходимо учитывать различную бизнес-логику в рекомендуемых объектах

Например

- ✦ Ограничить количество старых или слишком длинных видео
- ✦ Обеспечить разнообразие

Для этого после применения ранжирующей модели применяют механизм **реранкинг**.

# Реранкинг

Кроме того, зачастую на практике необходимо учитывать различную бизнес-логику в рекомендуемых объектах

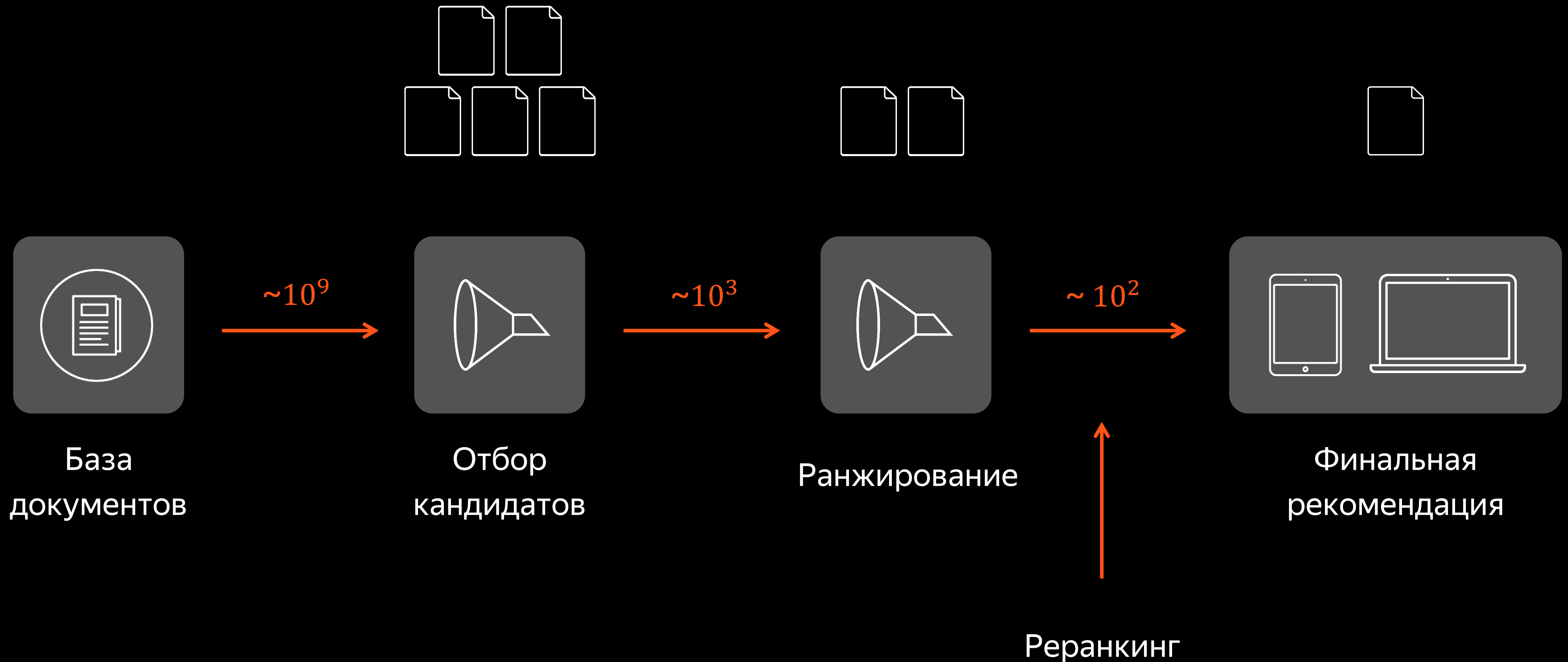
Например

- ✦ Ограничить количество старых или слишком длинных видео
- ✦ Обеспечить разнообразие

Для этого после применения ранжирующей модели применяют механизм **реранкинг**.

Отранжированные моделью объекты (топ) переранжируются с учетом требуемых условий

# Итоговая схема



# План лекции

1. Постановка задачи рекомендательных систем
  - Основные понятия
  - Виды фидбека
2. Коллаборативная фильтрация
3. Другие типы рекомендаций
  - Контентные модели
  - Гибридные модели
4. Рекомендательные системы на практике
  - Ранжирующая модель
  - Отбор кандидатов
  - Реранкинг
5. Особенности рекомендательных систем

# Проблема холодного старта

- 1 Проблема: как что-то рекомендовать новому пользователю? Как понять, кому показывать новый товар? Как вообще начинать что либо рекомендовать?

# Проблема холодного старта

1

Проблема: как что-то рекомендовать новому пользователю? Как понять, кому показывать новый товар? Как вообще начинать что либо рекомендовать?

2

Новый пользователь: попытаться узнать как можно больше информации о нем, использовать технические подходы или удлинить регистрацию, онбординг



# Проблема холодного старта

1

Проблема: как что-то рекомендовать новому пользователю? Как понять, кому показывать новый товар? Как вообще начинать что либо рекомендовать?

3

Новый товар: рекомендовать, используя только контентные модели

2

Новый пользователь: попытаться узнать как можно больше информации о нем, использовать технические подходы или удлинить регистрацию, онбординг

# Feedback loop

- ✦ Рекомендательная система учится на множестве примеров, которые сама же порекомендовала. Из-за этого мы можем "застрять" в локальном оптимуме, из которого сложно выбраться

# Feedback loop

- ✦ Рекомендательная система учится на множестве примеров, которые сама же порекомендовала. Из-за этого мы можем "застрять" в локальном оптимуме, из которого сложно выбраться
- ✦ Также есть риски, что рекомендательная система подстроится под множество пользователей или товаров, которые смотрят сейчас больше всего

# Feedback loop

- ✦ Рекомендательная система учится на множестве примеров, которые сама же порекомендовала. Из-за этого мы можем "застрять" в локальном оптимуме, из которого сложно выбраться
- ✦ Также есть риски, что рекомендательная система подстроится под множество пользователей или товаров, которые смотрят сейчас больше всего

## ✦ Простые решения:

1. С определенной вероятностью подмешивать случайные айтемы в выдачу
2. Стратифицировать обучающую выборку, например, по тегам, темам или популярности

# О чем сегодня поговорили

1. Постановка задачи рекомендательных систем
  - Основные понятия
  - Виды фидбека
2. Коллаборативная фильтрация
3. Другие типы рекомендаций
  - Контентные модели
  - Гибридные модели
4. Рекомендательные системы на практике
  - Ранжирующая модель
  - Отбор кандидатов
  - Реранкинг
5. Особенности рекомендательных систем