

Рекомендательные системы

ДПО ВШЭ

Современный анализ данных, глубокое обучение и приложения

Зима 2019

Эмиль Каюмов

Примеры

Рекомендуем также



-10%

2 488 ₽ ~~2 789 ₽~~

SSD диск SSD диск WD
Green 2,5" 240GB
(WDS240G2G0A)



-11%

1 401 ₽ ~~1 579 ₽~~

Smartbuy Splash 3
120GB SSD-накопитель
(SB120GB-SPLH3-25SAT3)



-7%

1 615 ₽ ~~1 739 ₽~~

Silicon Power Slim S55
60GB
(SP060GBSS3S55S25)



2 648 ₽

Kingston UV400 120Gb
SSD-накопитель
(SUV400S37/120G)



- Интернет-магазины

Примеры

The screenshot shows a user profile page with a blue header bar. On the left is the Facebook logo, followed by a search bar with the placeholder 'Поиск' and a magnifying glass icon. To the right is a user profile picture of a man named 'Эмиль' and a 'Гл' button. Below the header, the main content area has a light gray background. It features a section titled 'Вы можете их знать' (You may know them) with three items, each consisting of a small profile picture (redacted here), a name, and some descriptive text. At the end of each row are two buttons: a blue 'Добавить в друзья' (Add friend) button with a person icon and a white 'Удалить' (Delete) button.

- Работает в Яндекс
4 общих друга

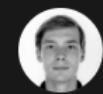
Добавить в друзья Удалить
- Работает в Яндекс
6 общих друзей

Добавить в друзья Удалить
- ваш общий друг.

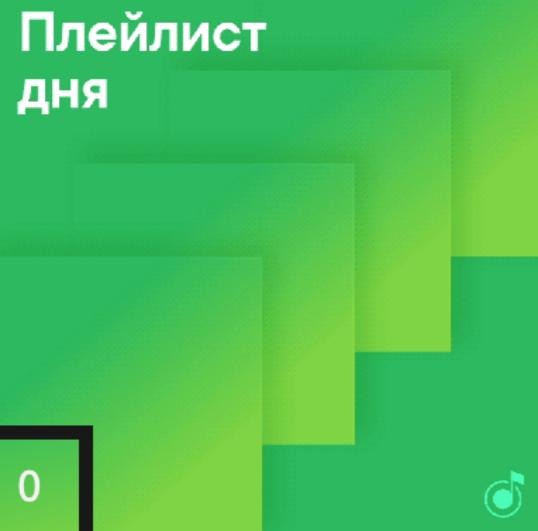
Добавить в друзья Удалить

- Социальные сети

Примеры



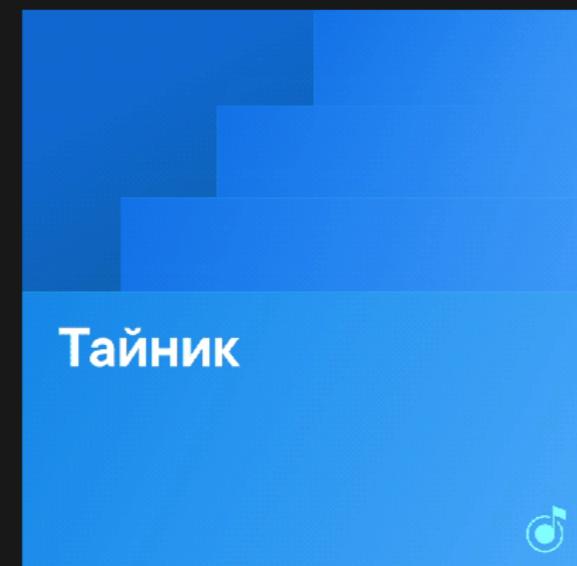
Умные плейлисты для вас, Эмиль



Плейлист дня

Каждый день — новый. Каждый день — ваш!

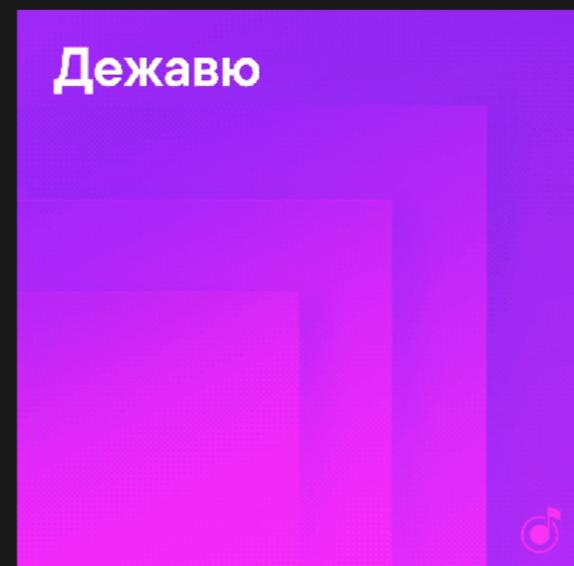
Обновлён сегодня



Тайник •

Треки из вашей фонотеки, которые вы ещё не послушали

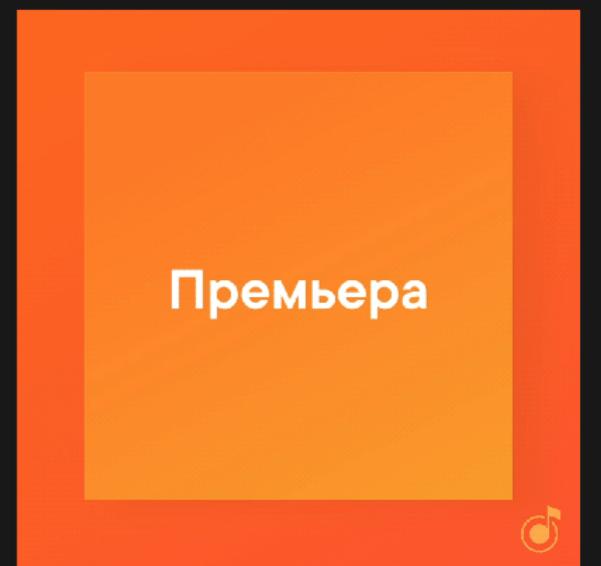
Обновлён сегодня



Дежавю

Вы ещё не слушали эти треки, но, похоже, вам они понравятся

Обновлён вчера



Премьера

Только новинки, подобранные по вашим предпочтениям

Обновлён 1 февраля

- Стreamинговые сервисы (музыка, фильмы, сериалы)

Примеры

Яндекс  Дзен

Лента

Подписки

Каналы



Персональная лента публикаций



Как устроена единственная в России кофейня только для женщин

В воскресенье, 3 февраля в Петербурге открылась единственная в России кофейня-коворкинг только для женщин...

THE VILLAGE





The Bell узнал имя возможного создателя Azino777 — это 33-летний разработчик из Татарстана

По данным издания, в создании Azino777 Альберта Валиахметова...

VC.RU — СТАРТАПЫ И БИЗНЕС





Число жертв пожара в центре Москвы возросло до восьми

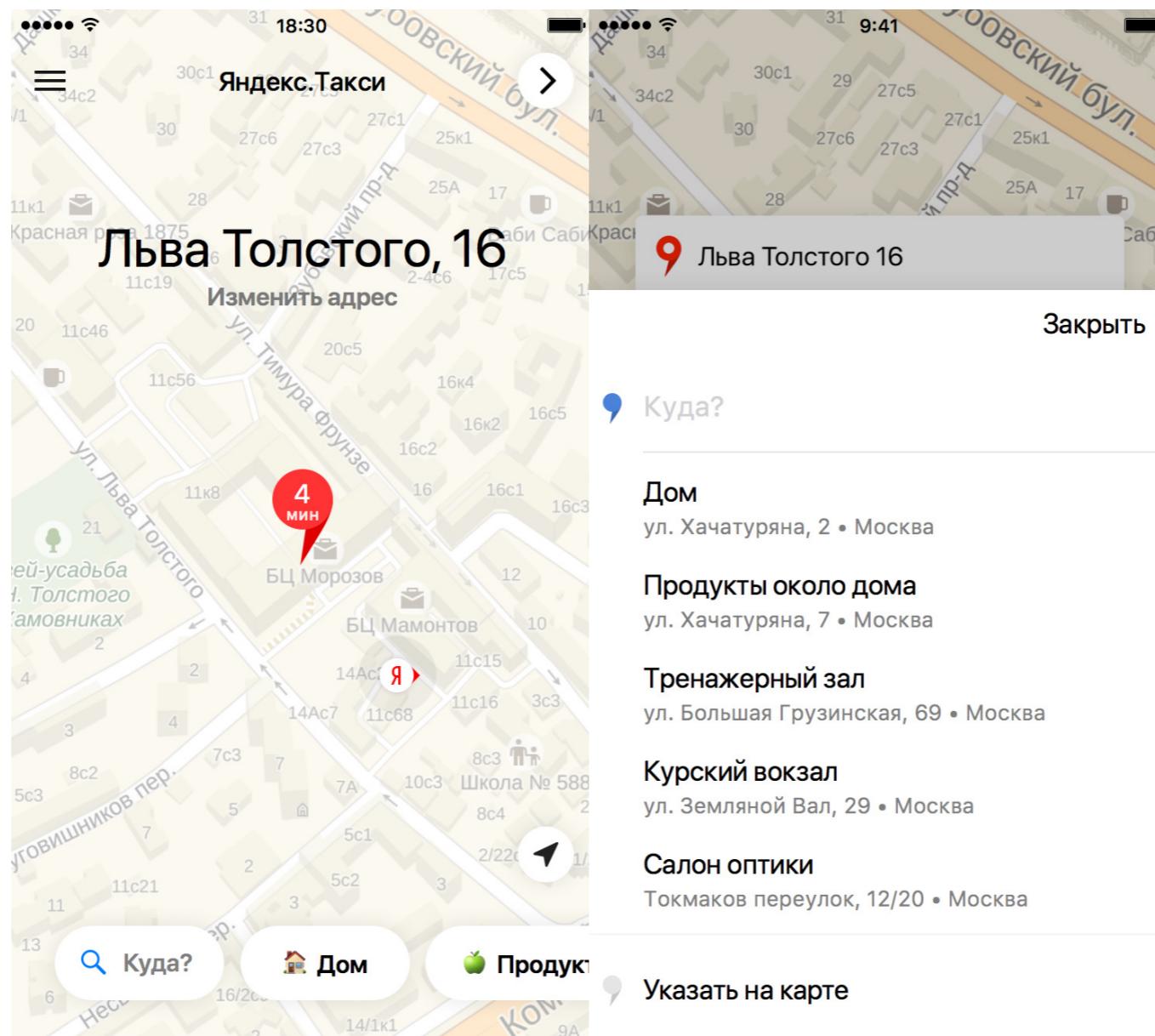
Москва. 4 февраля. INTERFAX.RU - На месте пожара в центре Москвы обнаружено тело еще одного погибшего, сообщил "Интерфаксу"...

ИНТЕРФАКС



- Персональные ленты

Примеры



История

The screenshot shows the Netflix Prize Leaderboard page. At the top, it says "Netflix Prize" and has a large red "COMPLETED" stamp. Below the header, there's a navigation bar with links for Home, Rules, Leaderboard, and Update. The main title "Leaderboard" is displayed in large blue text. A sub-instruction "Showing Test Score. Click here to show quiz score" is present. A dropdown menu indicates "Display top 20 leaders." The table below lists the top 5 teams with their names, best test scores, improvement percentages, and submission times.

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40
4	Opera Solutions and Vandelay United	0.8588	9.84	2009-07-10 01:12:31
5	Vandelay Industries !	0.8591	9.81	2009-07-10 00:32:20

- Netflix Prize (2006-2009)

История



The ACM Conference Series on
Recommender Systems

HOME

RECSYS 2019

PAST CONFERENCES

HONORS

BLOG

CONTACT

search...



ACM RecSys 2019

The 13th ACM Recommender Systems Conference will take place in Copenhagen, Denmark from 20, 2019.

RecSys 2018 (Vancouver)

RecSys 2017 (Como)

RecSys 2016 (Boston)

RecSys 2015 (Vienna)

RecSys 2014 (Silicon Valley)

RecSys 2013 (Hong Kong)

RecSys 2012 (Dublin)

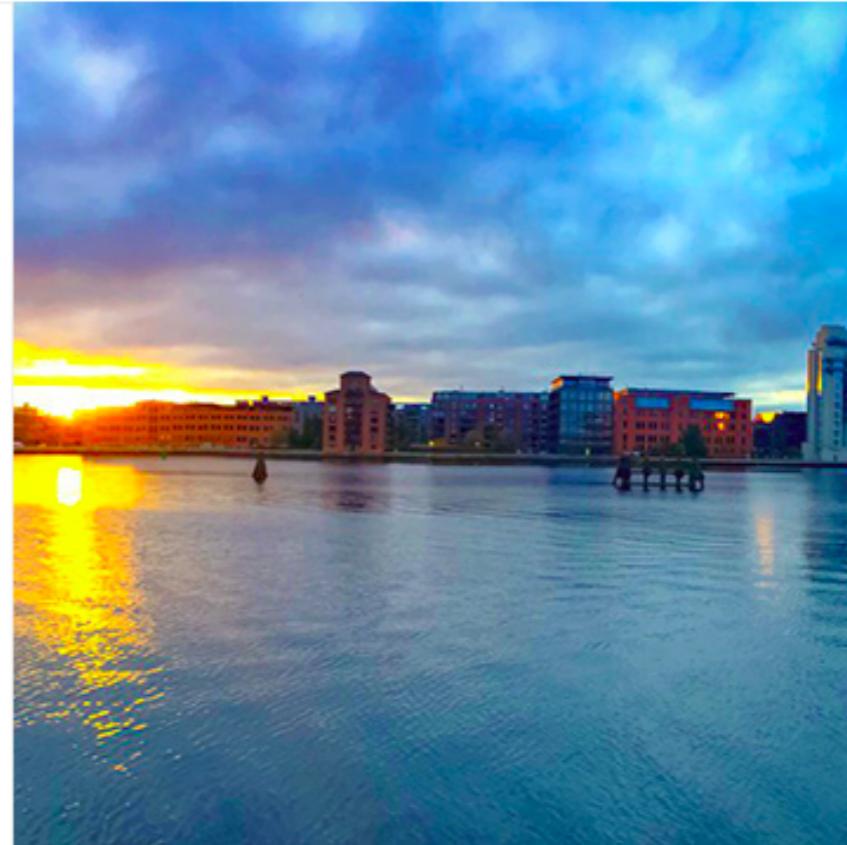
RecSys 2011 (Chicago)

RecSys 2010 (Barcelona)

RecSys 2009 (New York)

RecSys 2008 (Lausanne)

RecSys 2007 (Minnesota)



- RecSys Conference (2007–...)

Виды

- Коллаборативная фильтрация
 - Похожие пользователи – похожие товары
- Контентные подходы
 - Рекомендации по описанию товаров
- Гибридные подходы

План изучения

- Коллаборативная фильтрация
- Контентные походы
- Факторизационные машины
- Метрики в задачах рекомендаций
- Отбор кандидатов
- Свойства рекомендательных систем
- Нейросетевые подходы к рекомендациям

Common ML vs RecSys

- Common ML задача: объект и целевая переменная
- Рекомендации:
 - Что такое целевая переменная?
 - Откуда брать данные?
 - Что такое отрицательные примеры?
 - Как оценить качество системы?

Формализация

- user — пользователь
- item — «товар» (песня, фильм, книга, статья, ...)
- r_{ui} — «оценка» пользователем u товара i
- Хотим найти i для пользователя u с максимальным \hat{r}_{ui}

Коллаборативная фильтрация

- Идея в рекомендациях через поиск похожести между пользователями и товарами
- Прочие свойства пользователей и товаров не учитываются
- Memory-based подход и модели со скрытыми переменными

Memory-based

- Идея: если два пользователя похожи друг на друга, то одному из них надо рекомендовать то, что понравилось второму

- $I_{uv} = \{i \in I \mid \exists r_{ui} \& \exists r_{vi}\}$

- $w_{uv} = \frac{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_{uv}} (r_{vi} - \bar{r}_v)^2}}$

- — схожесть пользователей u и v

Memory-based

- Нашли w_{uv} , как сделать предсказания?
- $U(u_0) = \{v \in U \mid w_{u_0v} > \alpha\}$
- Рекомендуем к товаров с наибольшим весом:

$$p_i = \frac{|\{u \in U(u_0) \mid \exists r_{ui}\}|}{|U(u_0)|}$$

Memory-based

- Аналогично вместо схожести пользователей можно использовать схожесть товаров

$$U_{ij} = \{u \in U \mid \exists r_{ui} \& \exists r_{uj}\}$$

$$w_{ij} = \frac{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_i)(r_{uj} - \bar{r}_j)}{\sqrt{\sum_{u \in U_{ij}} (r_{ui} - \bar{r}_i)^2} \sqrt{\sum_{u \in U_{ij}} (r_{uj} - \bar{r}_j)^2}}$$

$$I(u_0) = \{i \in I \mid \exists r_{u_0 i}, w_{i_0 i} > \alpha\}$$

$$p_i = \max_{i_0: \exists r_{u i_0}} w_{i_0 i}$$

Memory-based

- Нужно хранить разреженную большую матрицу с оценками — тяжело хранить и обрабатывать
- Функции вычисления схожести выбраны «наобум» (можно взять другие) — не заточено под какую-либо функцию потерь

Модели со скрытыми переменными

- Идея: каждый пользователь и товар можно описать через его интерес/принадлежность к некоторыми категориям (вектор небольшой длины), скалярное произведение покажет их сходство

$$r_{ui} \approx \langle p_u, q_i \rangle$$

Модели со скрытыми переменными

- Будем явно учиться предсказывать оценки:

$$\sum_{(u,i) \in R} (r_{ui} - \bar{r}_u - \bar{r}_i - \langle r_u, q_i \rangle)^2 \rightarrow \min_{P, Q}$$

- Можно записать через матричное разложение
- Можно добавить регуляризацию (Latent Factor Model)

Модели со скрытыми переменными

- Обучение стохастическим градиентным спуском (для некоторой пары пользователь-товар)
- ALS (alternating least squares) – обновляем Р при фиксированной матрице Q и Q при фиксированной матрице Р

Холодный старт

- Новый пользователь или новый товар не имеют истории оценок, поэтому не можем использовать коллаборативную фильтрацию
- Можно использовать контентный подход (далее) или решать без машинного обучения
- После сбора оценок можно воспользоваться коллаборативной фильтрацией

Контентные модели

- Присутствуют текстовые описания, картинки, звук, ... для товаров
- Присутствует описание пользователей + момента времени
- Строим признаки решаем классическую задачу (классификация или регрессия)
- Негативные примеры явные или получены сэмплированием

Факториационные машины

- Предположим, что целевая переменная зависит от парных взаимодействий между признаками
- Если рассмотреть полиномиальную регрессию второго порядка, то в количества параметров будет пропорционально квадрату числа признаков
- Если присутствуют категориальные признаки и будет сделано бинарное кодирование, то число параметров окажется слишком большим

Факторизационные машины

- Пусть вес взаимодействия пары признаков будет выражаться произведением низкоразмерных скрытых векторов, характеризующих эти признаки.

$$a(x) = w_0 + \sum_{j=1}^d w_j x_j + \sum_{j_1=1}^d \sum_{j_2=j_1+1}^d \langle v_{j_1}, v_{j_2} \rangle x_{j_1} x_{j_2}$$

- Параметров станет пропорционально количеству признаков

Факторизационные машины

- Являются обобщением моделей с матричными разложениями
- Реализацию можно найти, например, в libFM

Факторизационные машины

- Предположим, что признаки объединяются в группы и есть взаимодействие с каждой другой группой описывается своим вектором.

$$a(x) = w_0 + \sum_{j=1}^d w_j x_j + \sum_{j_1=1}^d \sum_{j_2=j_1+1}^d \langle v_{j_1, f_{j_2}}, v_{j_2, f_{j_1}} \rangle x_{j_1} x_{j_2}$$

- Field-aware factorization machines (FFM)

Отбор кандидатов

- В конкретной задаче может быть много товаров
- Нужно выдать topK рекомендаций в нужный момент для конкретного пользователя
- В случае тяжёлой модели не можем применить её на всех товарах
- Нужно уменьшать список товаров для скоринга моделью
 - Простыми моделями
 - Эвристиками

Метрики качества рекомендаций

- Онлайн метрики – важные для бизнеса (продукта) показатели
 - Можем оценить только в работающей системе
 - Сравнивать модели в А/Б тестировании
- Оффлайн метрики – привычные в машинном обучении метрики
 - Выбираем те, которые коррелируют с нужными онлайн метриками

Предсказание рейтингов

- Рейтинги — числа в некотором диапазоне — задача регрессии
- MSE, RMSE, MAE, ...
- Не обязательно рейтинг — может быть другая вещественная величина (время просмотра)

Предсказание событий

- Событие (клик, просмотр, покупка, ...) – задача классификации
- LogLoss, ROC-AUC, PR-AUC, F-мера – есть недостаток

Предсказание событий

- Идея: показываем только k рекомендаций, поэтому будем оценивать только их
- Наличие верной рекомендации ($\text{hitrate}@k$), точность ($\text{precision}@k$), полнота ($\text{recall}@k$)

Ранжирование

- С другой стороны, нам не нужно предсказывать точные рейтинги или события — нужно лишь правильно ранжировать

$$DCG@k(u) = \sum_{p=1}^k g(r_{ui_p})d(p)$$

$$g(r) = 2^r - 1, d(p) = \frac{1}{\log(p + 1)}$$

$$nDCG@k(u) = \frac{DCG@k(u)}{\max DCG@k(u)}$$

Недостатки

- Хорошее качество по оффлайн метрикам ничего не гарантирует
 - Тройственность метрик машинного обучения
- Хорошее качества в онлайне не отражает качество рекомендаций
 - Пользователь и без нас мог купить этот же товар

Свойства рекомендательных систем

- Может получиться так, что система будет рекомендовать только популярные товары — **покрытие каталога**
 - Доля товаров, рекомендованных хотя бы раз
 - Энтропия системы
- Аналогично может быть так, что не всем пользователям что-либо рекомендуем — **покрытие пользователей**

Свойства рекомендательных систем

- Хотим показывать людям товары, которые они ещё не видели — **новизна**
 - Добавить кнопку в интерфейс
 - Промоделировать ситуацию удалением из выборки части товаров
 - Добавить веса для рекомендаций обратные популярности товара

Свойства рекомендательных систем

- **Прозорливость** – способность системы рекомендовать товары, не похожие на уже купленные
 - Например, доля рекомендаций, которые далеки от всех оценённых пользователем товаров

Свойства рекомендательных систем

- К ноутбуку в рекомендациях полезнее показать не 10 мышек, а набор из чехла, мышки, очистителя, зарядки – **разнообразие**
- Нужно измерять расстояние между товарами (в дереве каталога или по аналогии со схожестью в рекомендациях)