

# Рекомендательные системы

и кто над ними работает

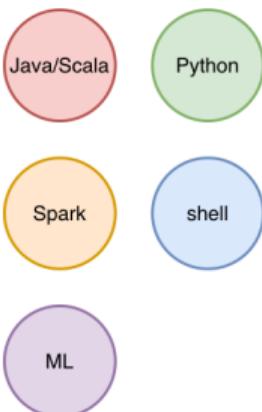
Николай Анохин

# Обо мне

## Академический опыт



## Навыки



## Индустриальный опыт



## Из этой лекции вы узнаете

- Как устроена рекомендательная система  
(на примере рекомендаций музыки)
- Что нужно знать и уметь, чтобы ее создать  
(это сложно, но вам точно по силам)



## Зачем нужны рекомендательные системы [RRSK10]

### Компаниям

- Увеличить продажи
- Добиться большей лояльности
- Улучшить пользовательский опыт

### Пользователям

- Найти лучший товар
- Найти все подходящие товары
- Залипнуть



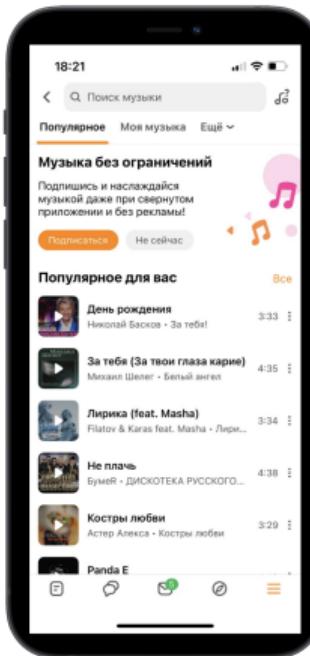
кинопоиск



## Рекомендации музыки

### Задача

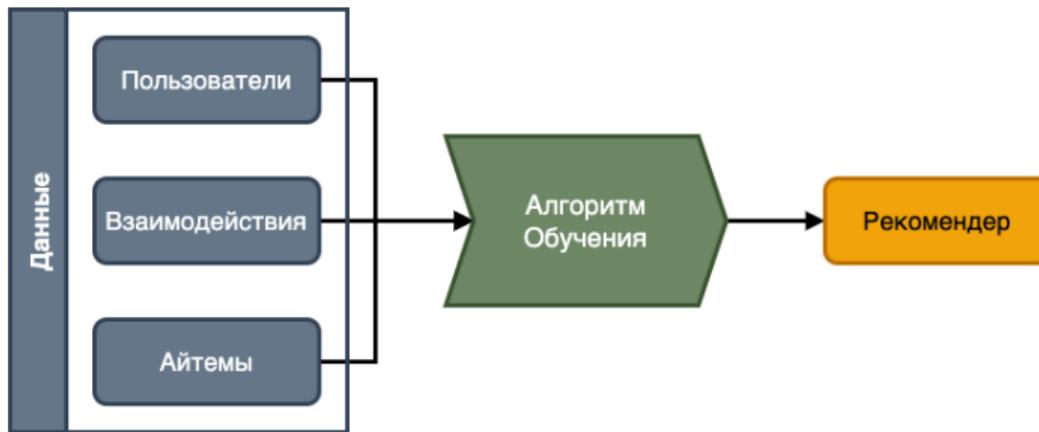
Рекомендовать пользователю исполнителей,  
так чтобы пользователь слушал их песни и  
как можно дольше оставался в приложении



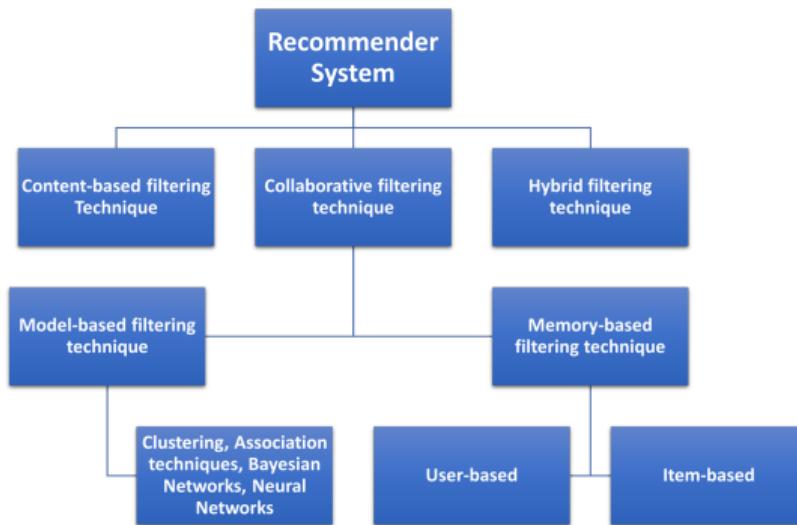
# Алгоритм рекомендаций



## Как строить рекомендации



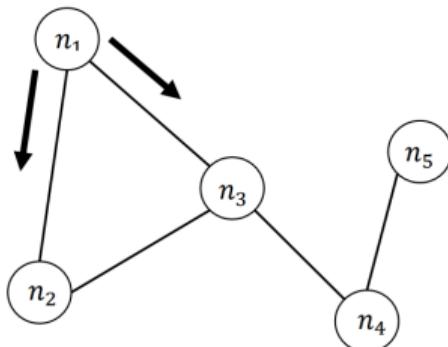
# Зоопарк алгоритмов рекомендаций



## Personalized Page Rank (PPR) [PLCL19]

Начальное состояние

$$\vec{s}_0 = (1 \ 0 \ 0 \ 0 \ 0)$$



Вероятность перехода

$$\vec{p}_{n+1} = c \vec{s}_n \cdot P + (1 - c) \vec{s}_0$$

$$P = \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 & 0 \\ 1/3 & 1/3 & 0 & 1/3 & 0 \\ 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 10 & 0 \end{pmatrix}$$

$(1 - c)$  – вероятность рестартра

Где мы окажемся через бесконечное число шагов?



## Строим граф исполнителей: создаем ребра I

### Задача

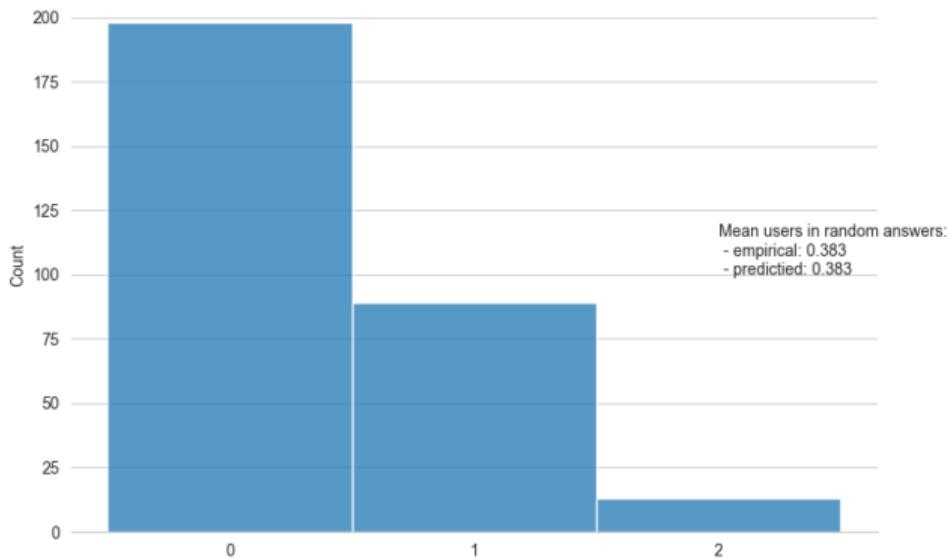
Есть  $K$  пользователей. Пользователь  $j$  случайным образом с равной вероятностью без возвращения выбирает  $n_j$  из  $N$  исполнителей. Сколько в среднем пользователей выберут каждую пару исполнителей?

### Ответ

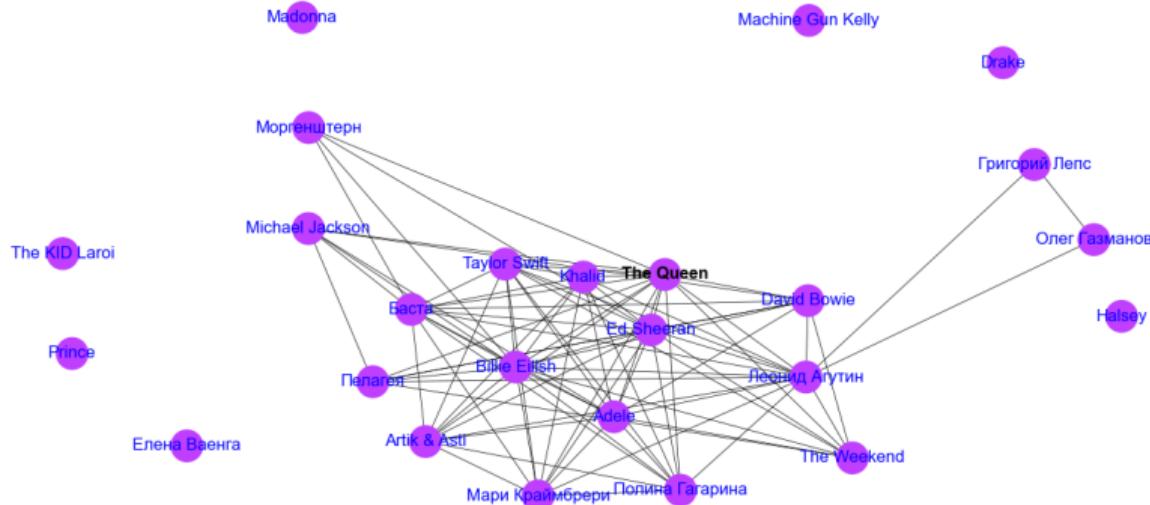
$$\frac{\sum n_j(n_j - 1)}{N(N - 1)}$$



## Строим граф исполнителей: создаем ребра II



## Строим граф исполнителей III



Алгоритм рекомендаций  
oooooooo●ooo

Продакшен  
ooooo

Данные для рекомендаций  
ooooo

Наука о рекомендациях  
oooo

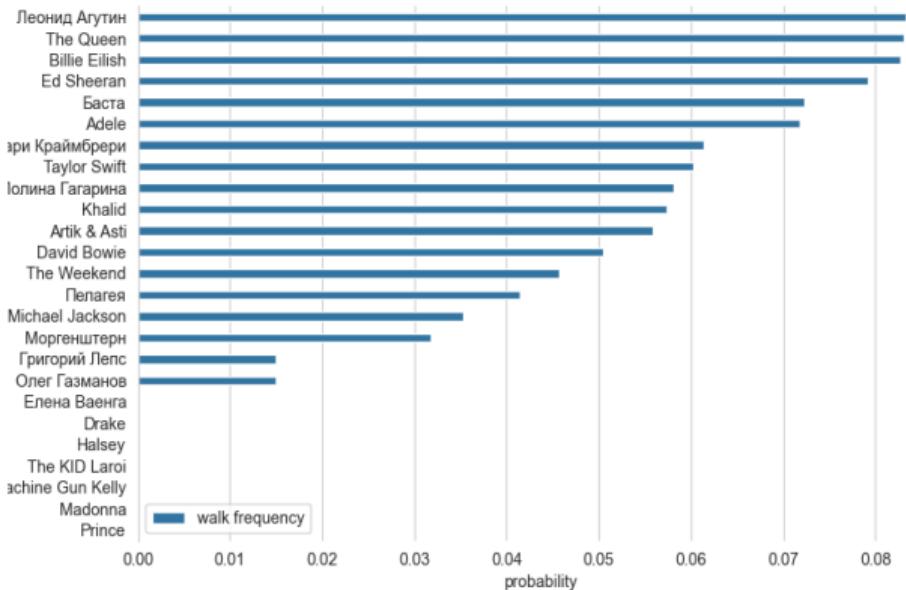
Итоги  
ooooo

# Случайные блуждания по графу исполнителей

[Ссылка на визуализацию](#)

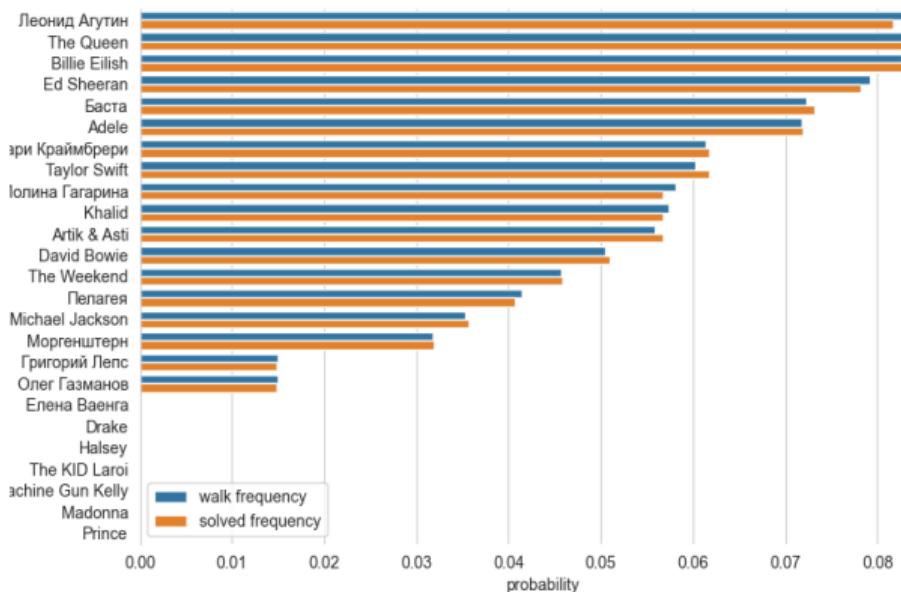


## Рекомендации на основе блужданий



## Рекомендации на основе точного решения

$$\vec{r} = c\vec{r}P + (1 - c)\vec{s} \quad \rightarrow \quad \vec{r} = (1 - c)\vec{s}(I - cP)^{-1}$$



ML Инженер



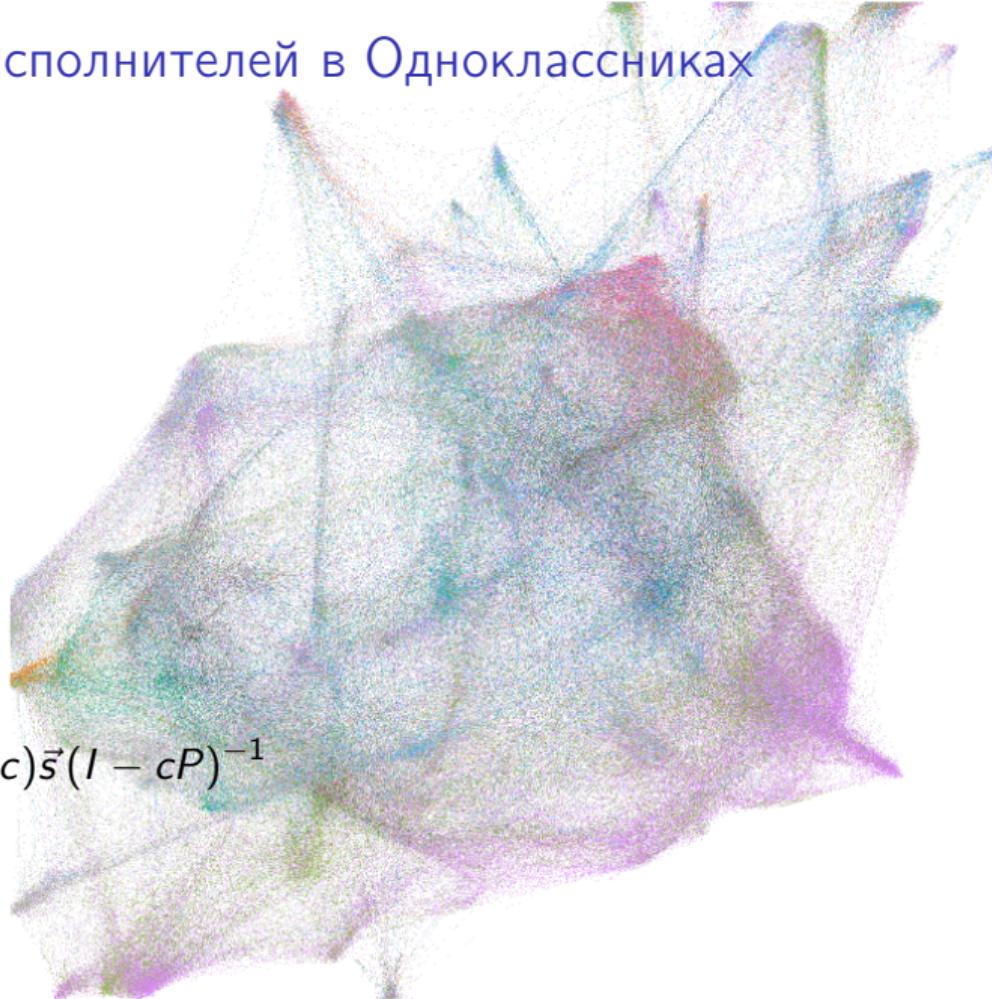
# Продакшен



## Сервис рекомендаций



# Граф исполнителей в Одноклассниках



## Параметры графа

20000 вершин

750000 ребер

## Память

$O(N_{nodes}^2)$

## Сложность

Прямое решение  $\vec{r} = (1 - c)\vec{s}(I - cP)^{-1}$

$O(N_{nodes}^3)$

## Оптимизируем вычисление рекомендаций

- Оптимизированное прямое решение
- Оптимизированный power iteration
- Алгоритм Bookmark coloring
- Динамическое программирование
- Сэмплирование монте-карло



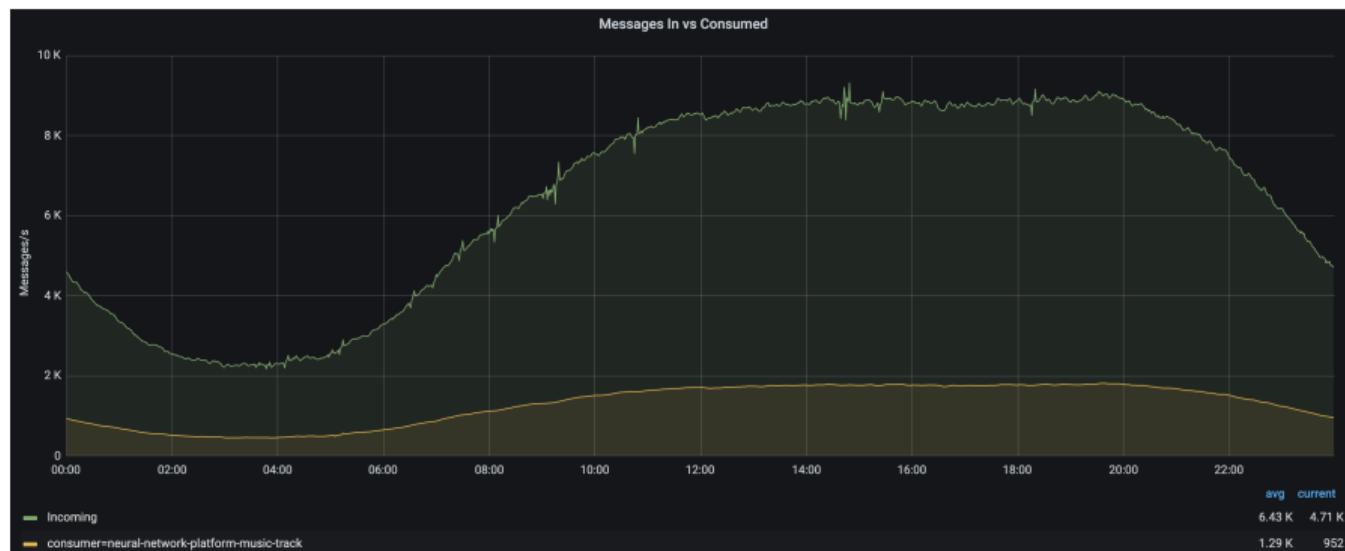
# Backend Разработчик



# Данные для рекомендаций



# Транспорт данных между сервисами

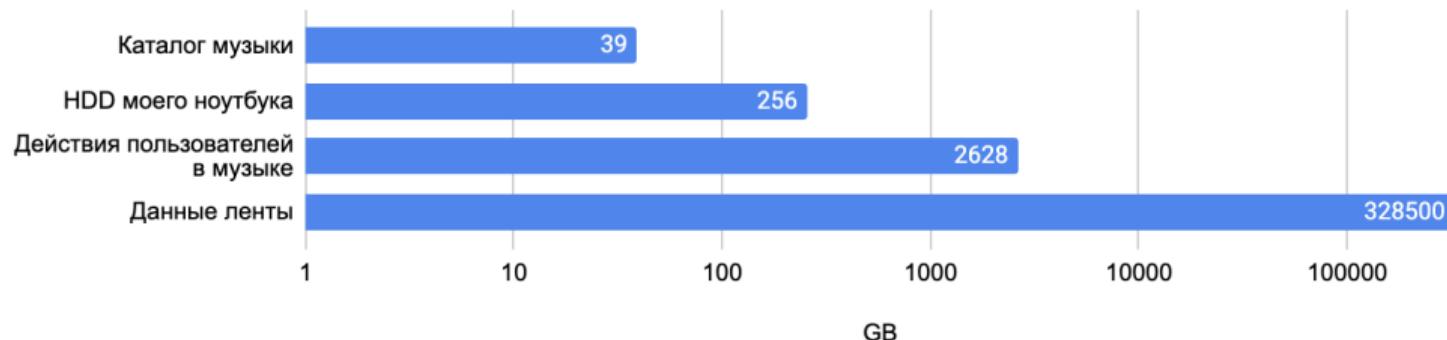


- Потоковая обработка данных (Spark Streaming, Apache Samza)
- Хранение данных для анализа



## Хранение и анализ данных

### Размер с репликацией



- Распределенная файловая система: Hadoop HDFS
- Анализ данных: Apache Spark
- Контроль выполнения: Apache Airflow



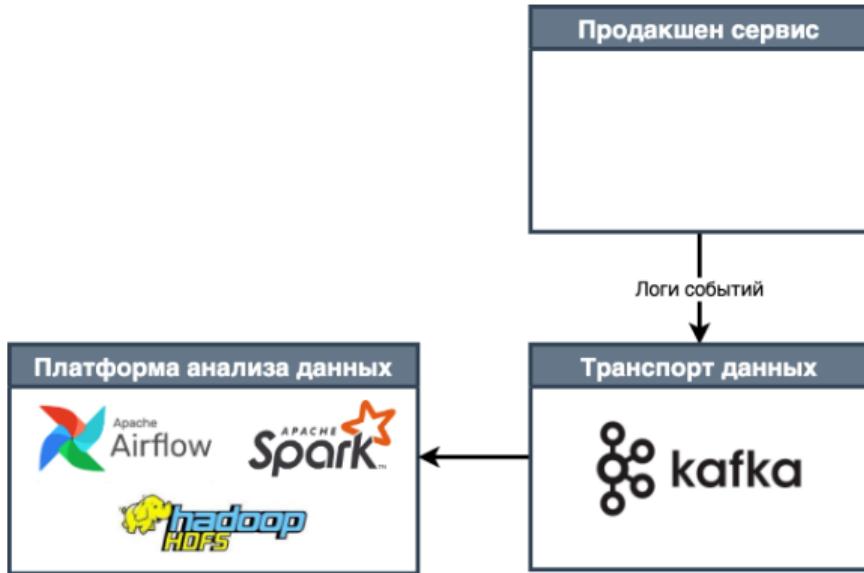
Алгоритм рекомендаций  
oooooooooooo

Продакшен  
ooooo

Данные для рекомендаций  
ooo•o

Наука о рекомендациях  
oooo

Итоги  
ooooo



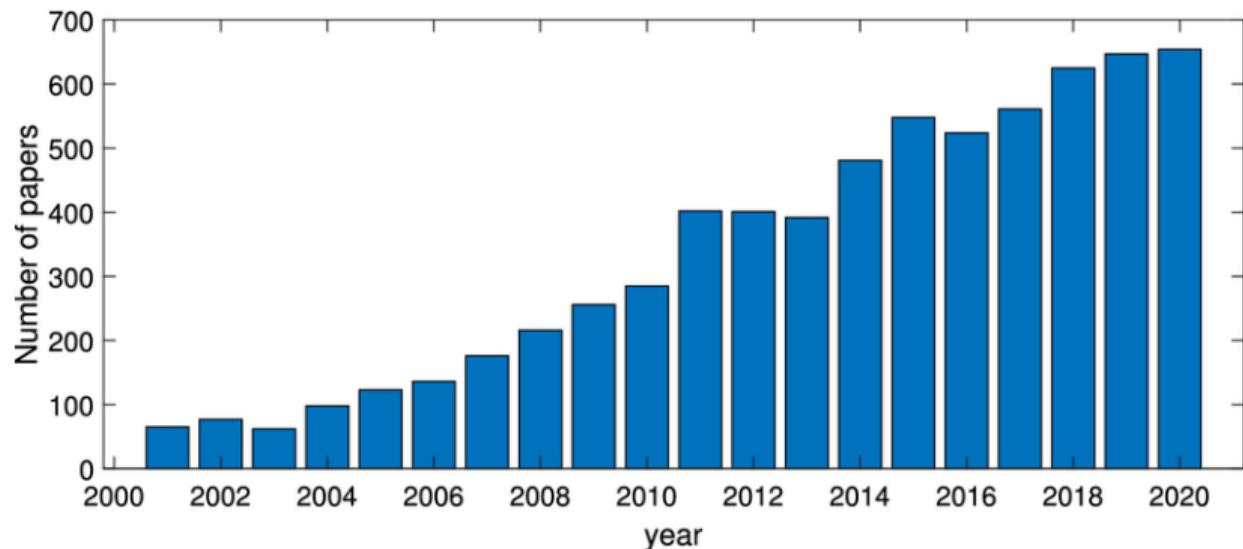
# Big Data Инженер



# Наука о рекомендациях



## Количество статей с “recommender system” в названии



## Актуальные задачи в науке о рекомендациях

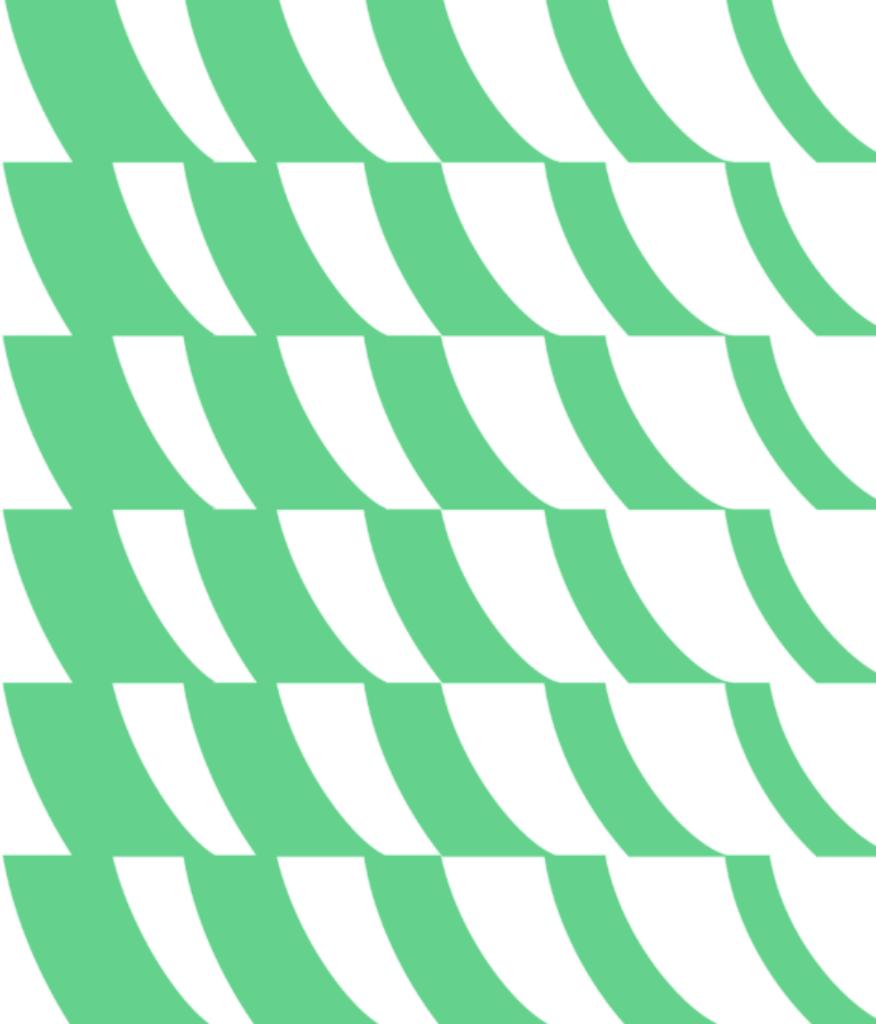
- Новые алгоритмы
- Счастье пользователей
  - Долгосрочный эффект
  - Удовлетворение всех сторон
  - Честность рекомендаций
- Выйти из пузыря
  - Разнообразие рекомендаций
  - Causal recommendations
  - Reinforcement learning
- Пользовательский опыт
  - Объяснения рекомендаций
  - Настройки рекомендаций
  - Оптимизация UI



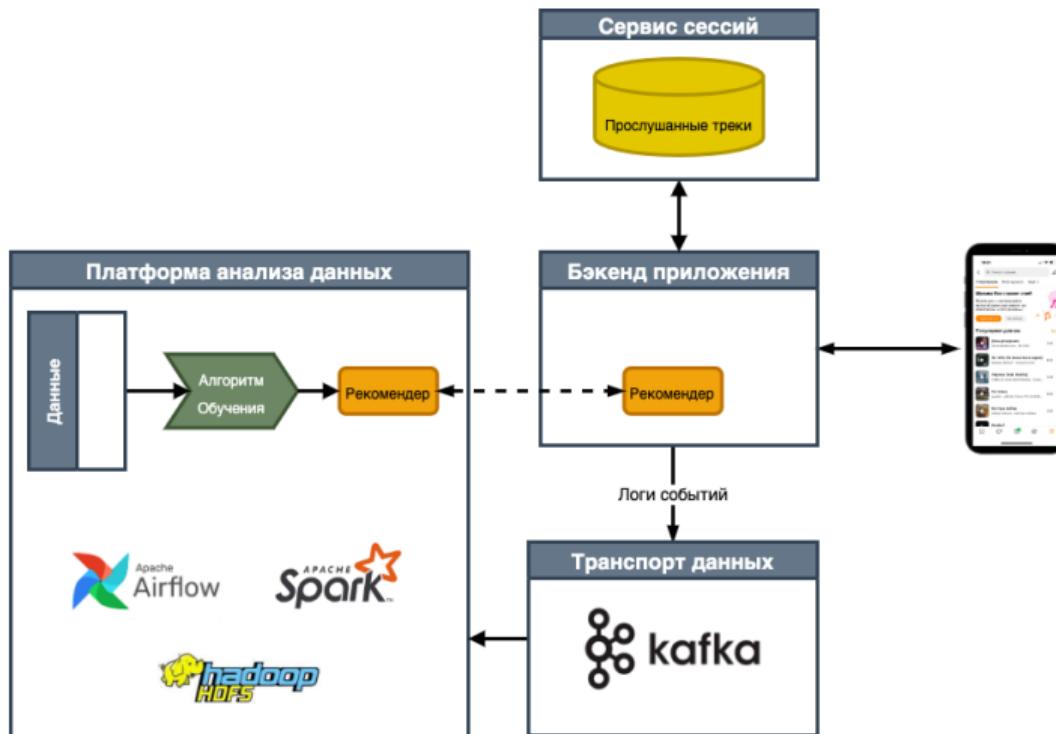
# Учёный



# Итоги



## Готовая рекомендательная система



Разработчики приложений

Менеджер проектов

Тестировщик

ML инженер

Фронтенд разработчик

Бэкенд разработчики

Big Data  
инженер

Директор по  
проектам

Ученых у  
нас нет :(





Спасибо за  
внимание



<https://t.me/anokhinn>

## Литература I

-  Sungchan Park, Wonseok Lee, Byeongseo Choe, and Sang-goo Lee, *A survey on personalized pagerank computation algorithms*, IEEE Access **PP** (2019), 1–1.
-  Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, *Recommender systems handbook*, 1st ed., Springer-Verlag, Berlin, Heidelberg, 2010.

