Часть 2. Анализ проблем и ограничений рекомендательных систем

а) Проблемы, специфичные для разных подходов RecSys

1. Проблема: Холодный старт (Cold Start)

Ситуация: Система не может рекомендовать новые товары (нет взаимодействий) или новым пользователям (нет истории)

Решения:

1) Анкета/опрос при регистрации: собираем интересы, категории, бренды, но опросы вызывают повышенный отток на старте
2) Эвристики: рекомендации популярных товаров (baseline), но this is неперсонализированный опыт
3) Гибридные модели: комбинируют контент и поведение других пользователей, но сложная реализация
4) Использование контекстных признаков (гео, устройство, время и др.)

Наверное лучшие решения: implicit feedback (время просмотра, клики), умные UI-шаги (например, лайк/дизлайк при первом запуске), и предварительное обучение модели на «похожих» юзерах

2. Проблема: Этические и репутационные риски

Ситуация: рекомендации на основе спорного поведения (алкоголь, товары 18+) и рекомендации, усиливающие предвзятости (bias): по полу, расе, возрасту

Решения:

1) Постпроцессинг рекомендаций с модерацией или фильтрацией
2) Блокировка чувствительных категорий при отсутствии согласия
3) Введение ярлыков-метрик fairness (рейтинг качества или справедливости) на рисковые товары
4) Использование контекстных признаков (гео, устройство, время и др.)

Из самого трудного, что может предстать перед тем, как мы получим результат – вопрос: “есть ли баланс между персонализацией и цензурой? ”

Этические системы RS точно требуют согласия пользователя, прозрачности алгоритма и возможности “пояснить” рекомендацию при претензиях

3. Проблема: Размывание пользовательского профиля

В Collaborative Filtering профиль пользователя может загрязняться шумными взаимодействиями (например, просмотр без интереса)

Решения:

1) Весовые коэффициенты по типу взаимодействия (покупка > клик)

2) Фильтрация временных/однократных действий
3) Введение decay-функций (старые интересы менее важны)

НО всё это усложняет модель и может появится некая непредсказуемость картины поведения пользователя

б) Проблемы и ограничения при развертывании RecSys в продакшене

1. Медленные отклики и масштабируемость

Допустим по причине больших связей в матрице взаимодействий или просто медленные запросы к бд

Решения:

1) Предрасчёт и кэширование рекомендаций (например, топ-10 товаров для каждого пользователя)
2) Использование технологий поиска по вектору (FAISS – “”Facebook AI Research Similarity Search – разработка команды Facebook AI Research для быстрого поиска ближайших соседей и кластеризации в векторном пространстве. Высокая скорость поиска позволяет работать с очень большими данными – до нескольких миллиардов векторов.””)
3) Асинхронные очереди и batch-обновления

Недостатки:

1) Падает свежесть рекомендаций
2) Усложнение архитектуры

2. Обновление модели и данных

Поведение пользователей быстро меняется, старые данные устаревают

Решения:

1) Online learning (модели обновляются на лету)
2) Mini-batch retraining + cron job (обновления каждый день/час)
3) Sliding window (например, только последние 30 дней)

Но всё это требует ресурсов и возможна регрессия качества при частом обновлении

Часть 3. Мониторинг и оценка RecSys

Проблемы и ограничения при развертывании RecSys в продакшене

ML-метрики (возможные)

Для оценки качества user experience и system efficienty:

1) Precision@K / Recall@K – насколько полезны топ-K рекомендаций

$$\text{precision@k} = \frac{\text{number of recommended items that are relevant @k}}{\text{number of recommended items @k}} \quad (1)$$

$$\text{recall@k} = \frac{\text{number of recommended items that are relevant @k}}{\text{number of all relevant items}} \quad (2)$$

2) NDCG@K – учёт порядка рекомендаций

2. **NDCG@K** – метрика для оценки качества ранжирования, которая учитывает порядок элементов в списке рекомендаций. Метрика нам интересна, т. к. мы стремимся к тому, чтобы наиболее релевантные товары оказались вверху блока с рекомендациями:

$$NDCG@K\,(u_j) = \frac{\sum_{k=1}^{K} p_k / \log_2(k+1)}{\sum_{k=1}^{\min\left(K, |T_{u_j}|\right)} 1/\log_2(k+1)}$$

где:

- $p_k = 1$, если k-й товар из рекомендованной корзины присутствует в T_{u_j}, и $p_k = 0$ в противном случае,
- знаменатель нормирует значение NDCG@K, чтобы оно лежало в диапазоне от 0 до 1,
- k – позиция товара в ранжированном списке рекомендаций P_{u_j}.

3) MAP (Mean Average Precision) – средняя точность по всем пользователям

4) Coverage – сколько товаров вообще попало в рекомендации

5) Diversity / Novelty – разнообразие рекомендаций - разнообразие+ новизна

Бизнес-метрики:

1) CTR (Click-Through Rate) – кликабельность рекомендаций
2) Conversion Rate – сколько пользователей совершили целевое действие
3) Revenue per User / per Impression – влияние на выручку от отного пользователя и от выручки за показ рекламы

4) Retention / DAU/WAU/MAU – удержание и вовлечённость (отслеживание дневной (DAU), недельной (wau) и месячной активности (mau)

5) Churn Rate – снижение оттока после внедрения RS, дает ответ на вопрос, какая доля пользователей перестала пользоваться сервисом

Мониторинг в продакшене:

1) Дашборды с метриками по:

    - Перформансу модели (время ответа, число запросов)
    - Поведенческой аналитике (влияние на метрики вовлечения)
2) A/B-тестирование при внедрении новых моделей
3) Алерты: резкое падение качества или метрик (например, CTR ↓ 50%)