

Московский государственный университет имени М. В. Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра математических методов прогнозирования

Демин Георгий Александрович

**Методы построения рекомендательных систем**

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Научный руководитель:

к.ф-м.н., профессор

Дьяконов А. Г.

Москва, 2020

# Содержание

<b>1 Введение</b>	<b>3</b>
<b>2 Рекомендательных системы</b>	<b>3</b>
2.1 Общая постановка задачи . . . . .	3
<b>3 Специальные постановки задач</b>	<b>4</b>
3.1 Некоторые подходы к построению . . . . .	5
3.1.1 Memory-based model . . . . .	5
3.1.2 Model-based method . . . . .	5
3.1.3 Social network recommender system . . . . .	6
3.2 Нейросетевые методы . . . . .	7
<b>4 Постановка задачи</b>	<b>7</b>
<b>5 Рекомендательные системы на сетях</b>	<b>7</b>
5.1 GraphRec-WWW19 . . . . .	8
<b>6 Рекомендательные системы на группах</b>	<b>9</b>
6.1 Подходы к решению . . . . .	9
6.2 AGREE . . . . .	9
<b>7 Предлагаемый подход</b>	<b>9</b>
<b>8 Эксперименты</b>	<b>10</b>
<b>9 Результаты</b>	<b>11</b>
<b>10 Литература</b>	<b>11</b>

## **Аннотация**

В работе рассмотрены методы построения рекомендательных систем для разных типов данных. В данной работе решается задаче улучшения работы рекомендательной системы на группах с помощью моделирования графовой структуры и применения рекомендательной системы на сетях. Был предложен и реализован метод применения рекомендательных систем на сетях для данных, в которых пользователи объединены в группы, но не представляют собой граф (такие данные изначально рассчитаны на групповые рекомендательные системы). Проведены эксперименты, показывающие, что данный метод улучшает качество работы рекомендательной системы.

# 1 Введение

Рекомендательные системы - это класс моделей машинного обучения, предназначенных для решения задачи подбора товара (услуги) для пользователей. Постановки задач в конкретных случаях могут сильно отличаться, зависит это от предметной области и имеющихся данных. В работе будут рассмотрены различные примеры постановок задач и разработанные алгоритмы для их решения. Более подробно будут рассмотрены рекомендательные системы на сетях и группах пользователей и показано, как переходить от одной постановки задачи к другой.

## 2 Рекомендательных системы

### 2.1 Общая постановка задачи

Наиболее обще задачу можно поставить так: дано множество пользователей  $U$ , множество товаров  $I$ , а также  $R$  - множество взаимодействий пользователей и товаров, то есть существует не всюду определенное отображение  $f : (U \times I) \rightarrow R$ . Введём стандартные обозначения:

$f(u, i) = r_{ui}$  - известное взаимодействие пользователя  $u$  и товара  $i$

$\hat{r}_{ui}$  - предсказанное взаимодействие

$U_i$  - множество пользователей, оценивших товар  $i$

$I_u$  - множество товаров, оценённых пользователем  $u$

Задача состоит в том, чтобы восстановить значения  $f$  на множестве, на котором отображение не определено.

Обычно решается задача обучения с учителем:  $\{(X, y)\} = \{(u, i), r_{ui}\}$ . Множество  $R \in \mathbb{Z}_+^{n \times m}$ , представляет собой матрицу (ее называют матрицей интеракций), где на пересечении  $i$  столбца и  $u$  строки стоит число - оценка пользователя  $u$  товару  $i$  (здесь и далее под товарами будут пониматься любые сущности (посещенные рестораны, просмотренные фильмы, оказанные услуги), оценки которым пользователь может поставить. Также рассматриваются задачи, в которых матрица интеракции содержит не оценки товаров, а факт взаимодействия (1 - если пользователь что-либо сделал

с товаром, 0 - иначе). Такие модели называются неявными. [Подробнее про неявные модели](#)

	Товар 1	Товар 2	Товар 3	Товар 4	Товар 5
Клиент 1		3		5	
Клиент 2	1		1	1	
Клиент 3	2			3	2
Клиент 4		4			5
Клиент 5	5		2	3	4

Рис. 1: Пример матрицы интеракций

### 3 Специальные постановки задач

Далее будут рассмотрены различные постановки задач (в зависимости от наличия и характера дополнительной информации) и методы их решений (далее будет подразумеваться, что матрица интеракций известна всегда, задачи, в которых она не задана рассматриваться не будут)

1. Известна только матрица интеракций. Такие задачи называются collaborative filtering.
2. Информация о пользователях (например, социально-демографическая) или о товарах (категория товара, его цена, статистики продаж), задачи с такой информацией называются content-based filtering.
3. Информация о времени или месте покупки товара — context-aware collaborative filtering
4. Известна не только оценка пользователя, но и текстовый отзыв — Review-based recommender systems
5. Между пользователями существуют дружеские связи (Social networks recommender systems) или они объединены в группы (Group recommender systems)

## 3.1 Некоторые подходы к построению

### 3.1.1 Memory-based model

Это самый наивный метод рекомендации, опишем его подробно, чтобы дать представление как проектируются алгоритмы рекомендаций и от каких характерных проблем страдают.

Этот подход основан на нахождении похожих пользователей (или товаров) и усреднении их оценок. Интуиция метода такова: пользователю понравится товар, если этот товар нравится похожим пользователям (которые оценивают товары примерно также как данный пользователь). Будем находить

$$\hat{r}_{ui} = \frac{1}{|U_i|} \sum_{u' \in U_i} \text{sim}(u, u') r_{u'i} \quad (1)$$

Для товара  $i$  находятся пользователи, которые его оценили  $u'$ , для каждого ищется его мера схожести с  $u$  (это может быть косинусное расстояние, коэффициент Жакарра, посчитанные по матрице инеракций) и рейтинги этих пользователей складываются с весами, соответствующими похожести пользователя на  $u$ . Таким образом мы получаем искомые рейтинг как некоторую линейную комбинацию всех рейтингов этого товара. Этот подход очень прост: он не требует больших вычислений, хранения дополнительных данных. В силу своей примитивности метод выдает плохие результаты для реальных датасетов. Он также подвержен недостаткам, общих для многих моделей коллаборативной фильтрации: рекомендация самых популярных товаров и проблема холодного старта (для нового пользователя не можем посчитать его схожесть с другими). Последняя проблема характерна для многих моделей рекомендательных систем.

### 3.1.2 Model-based method

Основная идея данного метода - разложение разреженной матрицы интеракции на произведение матриц меньшего ранга (matrix factorization - MF).

$$\min_{P,Q} \|R - PQ^T\|_F^2 \quad (2)$$

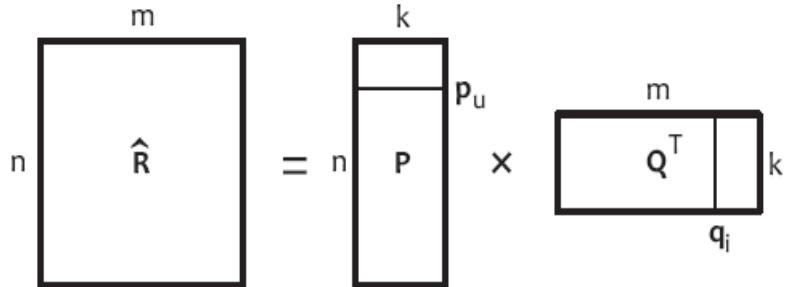


Рис. 2: Матричное разложение

В формуле (2)  $U \in \mathbb{R}^{n \times k}$ ,  $V \in \mathbb{R}^{m \times k}$  инимизируется норма Фробениуса. По сути для каждого товара и пользователя выучивается эмбеддинг размерности  $k$ . На практике вместо (2) используется (3): в этой оптимизационной задаче мы учим  $P$  и  $Q$  только по известным рейтингам в  $R$ , также она учитывает регуляризацию. Функционал в (3) соответствует функционалу Ridge-регрессии, для которого известно аналитическое решение, однако обычно эту задачу решают болчно-координатным спуском.

$$\sum_{u,i} (r_{u,i} - p_u q_i^T) + \lambda(\|p_u\|^2 + \|q_i\|^2) \rightarrow \min_{P,Q} \quad (3)$$

Эта модель требует дополнительного хранения  $P$  и  $Q$  и времени на обучение, однако как преимущество мы получаем эмбеддинги пользователей и товаров, которые можно использовать далее.

Рассмотренный метод построения эмбеддингов (как с помощью классических матричных разложений [1], так и с помощью нейросетей [2]) является одним из самых востребованных методов решения задачи колаборативной фильтрации при отсутствии дополнительной информации.

### 3.1.3 Social network recommender system

Рассматривается задача рекомендации на сетях. Пусть помимо матрицы интеракции задан граф  $G = \langle U, E \rangle$ , вершинами в котором являются пользователи, а ребрами помечаются связи между ними (дружеские, родственные или другие). Таким образом  $e_{u_1 u_2} \in E$  если пользователи  $u_1$  и  $u_2$  как-то связаны. Задача — восстановить неизвестные рейтинги  $\hat{r}_{ui}$ . В последние годы было выпущено множество статей, предлагающих различные алгоритмы для решения данной задачи. В данной работе

подробно будет рассмотрена и использована модель, предложенная в [3]. Эта работа выбрана, так как является достаточно новой с одной стороны, а с другой существует ее реализация, доступная для использования и модификаций. [сократить](#)

### 3.2 Нейросетевые методы

[дописать](#)

## 4 Постановка задачи

Выше приведено много различных постановок задачи рекомендательных систем и методов их решений. В связи с этим становится актуальной задача преобразования входных данных для обработки их различными моделями. Предлагается вместо разработки новой модели для входных данных, преобразовать их неким образом, чтобы иметь возможность сделать предсказание с помощью модели, которая работает с другими структурами данных и засчет ансамблирования 2 принципиально различных моделей получить лучший результат.

В частности в данной работе требуется подход, с помощью которого в данных, содержащих информацию о группах пользователей моделируется графовая структура. Таким образом, эти данные становятся доступны для рекомендательных систем на сетях.

[Улучшить стиль](#)

Рассмотрим подробнее задачи рекомендации на сетях и групповые рекомендации.

## 5 Рекомендательные системы на сетях

Пусть, кроме матрицы интеракции, известны связи между пользователями (дружеские, родственные и т.д.). Формально это означает, что задан граф  $\Gamma = \langle U, E \rangle$  и  $\exists e_{uv} \in E$ , если пользователи  $u$  и  $v$  связаны. Предполагается, что пользователи связаны симметрично, то есть граф ненаправленный, также предполагается, что граф невзвешенный (то есть все связи между пользователями равнозначны). Задача рекомендательной системы не изменяется: необходимо найти  $r_{ui}$ . На сегодняшний день

для решения этой задачи используют нейросетевые модели. Рассмотрим основные идеи, применяемый в них

1. Эмбеддинги
2. Механизм внимания
3. Случайное семплирование

## 5.1 GraphRec-WWW19

Рассмотрим конкретную модель графовых рекомендаций - Graph Neural Networks for Social Recommendation [3]. Это модель колаборативной фильтрации с вниманием, ее архитектура показана на рис. 3. Важной особенностью данной модели является то, что эмбеддинги строятся практически для каждой сущности:

1. эмбеддинг каждого уникального рейтинга  $r \in \{1, 2, 3, 4, 5\}$
2. эмбеддинг пользователя состоит из эмбеддинга относительно других пользователей (его соседей в графе) и эмбеддинга относительно товаров (которые он оценил)
3. эммбеддинг товара (линейная комбинация по все пользователям, оценившим товар, эмбеддинга пользователя сконкатенированного с эмбеддингом рейтинга) [может, здесь лучше формулами](#)

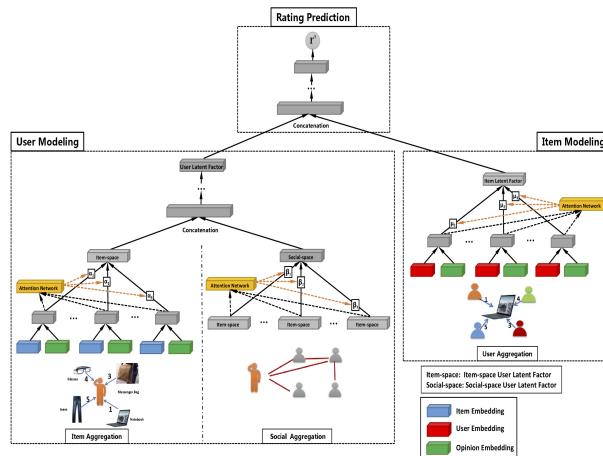


Рис. 3: Архитектура модели GraphRec

Таким образом модель может обучаться и предсказывать результат даже в случае, когда никакой дополнительной информации о пользователях и товарах не задано. Механизм внимания используется, чтобы определить с каким весом

1. каждый пользователь влияет на товар
2. каждый товар влияет на пользователя
3. каждый пользователь влияет на пользователя, который является его соседом

blueучитывает схожесть неявно

## 6 Рекомендательные системы на группах

Пусть имеются пользователи и они распределены по некоторым группам (возможно, пересекающимся), группа также может поставить оценку товару. Требуется восстановить неизвестные рейтинги  $\hat{r}_{ui}$ . [дополнить аналогично сетям](#)

### 6.1 Подходы к решению

#### 6.2 AGREE

В качестве примера рекомендательной системы на группах будем рассмотрим модель, описанную в [4], Архитектура этой модели принципиально схожа с GraphRec-WWW19 (также используется Attention-слой и эмбеддинги), с той лишь разницей, что в этой модели также строятся и эмбеддинги групп. [подробнее](#)

## 7 Предлагаемый подход

Смоделируем графовую структуру следующим образом: пусть все пользователи, входящие в одну группу связаны. [еще 2 подхода: случайные связи и полностью случайная структура](#) для предсказаний неизвестных рейтингов.

Далее предлагается обогатить матрицу интеракций пользователь-товар. Рассмотрим всех пользователей, входящих в группу  $U(g)$  и рейтинги, которыми группа оценила товары  $\{r_{gv_1}, \dots, r_{gv_k}\}$ , и для каждого пользователя из группы  $u \in U(g)$  добавим

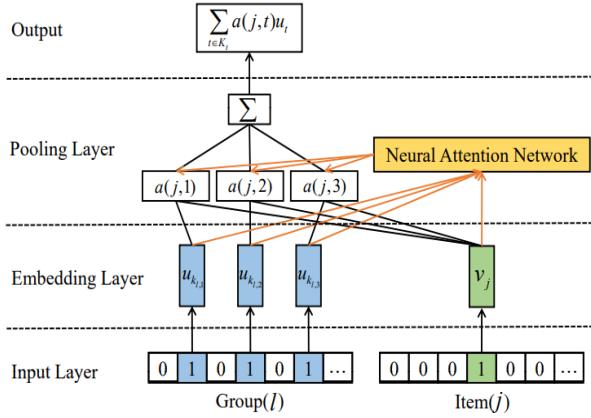


Рис. 4: Архитектура модели Agree

в матрицу интеракций рейтинг  $r_{uv_j} = r_{gv_j}$ . То есть, будем считать, что, если человек состоит в группе, то он оценил соответствующий товар той же оценкой, что и группа. Таким образом, мы добавим большое количество рейтингов в матрицу интеракций, что облегчит процесс обучения сетевой рекомендательной системы.

## 8 Эксперименты

Для тестирования предлагаемого подхода применяется датасет CAMRa2011 (он был использован в оригинальной статье [4])

1. 600 пользователей
2. 200 групп (в каждой группе в среднем 2.5 пользователя)
3. 7000 товаров
4. 140000 рейтинги группа-товар
5. 110000 рейтинги пользователь-товар

В таблице 1 представлено сравнение метрик качества для каждой модели в отдельности и в случае их агрегации (в качестве агрегации используется простое среднее арифметическое предсказаний 2 моделей). Мы видим, что качество при агрегировании моделей намного выше (метрики RMSE и MAE соответственно меньше), чем

качество каждой модели в отдельности. Это позволяет говорить о том, что предлагаемый метод действительно улучшает качество моделей. [еще эксперименты - применить простой MF: ALS LifgtFM также датасет Movie Lens](#)

	GraphRec-WWW19	AGREE	aggregated
RMSE	0.9944	0.9860	0.9703
MAE	0.7416	0.7542	0.7314

Таблица 1: Сравнение метрик качества моделей

## 9 Результаты

Предложен подход, позволяющий использовать датасеты, хмодержащие группы пользователей для моделей рекомендаций на сетях. Этот подход увеличивает качество предсказаний [дополнить после экспериментов](#)

## 10 Литература

[дополнить, статьи есть](#)

## Список литературы

- [1] Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges Y Shi, M Larson, A Hanjalic - ACM Computing Surveys (CSUR), 2014 - dl.acm.org
- [2] Deep learning based recommender system: A survey and new perspectives S Zhang, L Yao, A Sun, Y Tay - ACM Computing Surveys (CSUR), 2019 - dl.acm.org
- [3] Wenqi Fan, Yao Ma , Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. Graph Neural Networks for Social Recommendation. In Proceedings of the 28th International Conference on World Wide Web (WWW), 2019.

- [4] Da Cao, Xiangnan He, Lianhai Miao, Yahui An, Chao Yang, and Richang Hong. 2018. Attentive Group Recommendation. In The 41st International ACM SIGIR Conference on Research Development in Information Retrieval (SIGIR '18). ACM, New York, NY, USA, 645-654.