

# Лекция 2

## Рекомендательные системы

Е. А. Соколов  
ФКН ВШЭ

9 февраля 2020 г.

## 1 Признаки в рекомендательных системах

### §1.1 Коллаборативная фильтрация

#### 1.1.1 Учёт неявной информации

Выше мы обсуждали, что интерес пользователя к товару может выражаться по-разному. Это может быть как явный (выставление рейтинга или лайк, написание рецензии с оценкой), так и неявный (просмотр видео, посещение страницы) сигнал. Неявным сигналам нельзя доверять слишком сильно — пользователь мог по многим причинам смотреть страницу товара. При этом неявной информации гораздо больше, и поэтому имеет смысл использовать её при обучении моделей.

Один из способов учёта неявной информации предлагается в методе Implicit ALS (iALS) [2]. Введём показатель неявного интереса пользователя к товару:

$$s_{ui} = \begin{cases} 1, & \exists r_{ui}, \\ 0, & \text{иначе.} \end{cases}$$

Здесь мы считаем, что даже если пользователь поставил низкую оценку товару, то это всё равно лучше ситуации, в которой пользователь совсем не поставил оценку. Это не очень сильные рассуждения — пользователь мог просто не найти товар, и в таком случае неправильно судить об отсутствии интереса. Поэтому введём веса  $c_{ui}$ , характеризующие уверенность в показателе интереса  $s_{ui}$ :

$$c_{ui} = 1 + \alpha r_{ui}.$$

Коэффициент  $\alpha$  позволяет регулировать влияние явного рейтинга на уверенность в интересе.

Теперь мы можем задать функционал:

$$\sum_{(u,i) \in D} c_{ui} (s_{ui} - \bar{s}_u - \bar{s}_i - \langle p_u, q_i \rangle)^2 + \lambda \sum_u \|p_u\|^2 + \mu \sum_i \|q_i\|^2 \rightarrow \min_{P,Q}$$

Как и раньше, обучать его можно с помощью стохастического градиентного спуска, ALS или HALS. Предложенные способы вычисления  $s_{ui}$  и  $c_{ui}$  могут изменяться в зависимости от специфики задачи.

### 1.1.2 Факторизационные машины

Рассмотрим признаковое пространство  $\mathbb{R}^d$ . Допустим, что целевая переменная зависит от парных взаимодействий между признаками. В этом случае представляется разумным строить полиномиальную регрессию второго порядка:

$$a(x) = w_0 + \sum_{j=1}^d w_j x_j + \sum_{j_1=1}^d \sum_{j_2=j_1+1}^d w_{j_1 j_2} x_{j_1} x_{j_2}.$$

Данная модель состоит из  $d(d-1)/2 + d + 1$  параметров. Если среди признаков есть категориальные с большим числом категорий (например, идентификатор пользователя), то после их бинарного кодирования число параметров станет слишком большим. Чтобы решить проблему, предположим, что вес взаимодействия признаков  $j_1$  и  $j_2$  может быть аппроксимирован произведением низкоразмерных скрытых векторов  $v_{j_1}$  и  $v_{j_2}$ , характеризующих эти признаки. Мы получим модель, называемую *факторизационной машиной* (factorization machine, FM) [3]:

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

Благодаря описанному трюку число параметров снижается до  $dr + d + 1$ , где  $r$  — размерность скрытых векторов.

Данная модель является обобщением моделей с матричными разложениями. Задачу LFM

$$\sum_{(u,i) \in R} (r_{ui} - \bar{r}_u - \bar{r}_i - \langle p_u, q_i \rangle)^2 \rightarrow \min_{P, Q} \quad (1.1)$$

можно сформулировать как задачу построения регрессии с двумя категориальными признаками: идентификатором пользователя и идентификатором товара. Целевым признаком является рейтинг  $r_{ui}$ . Для некоторого подмножества пар (пользователь, товар) мы знаем рейтинг; для остальных мы хотим его восстановить. После бинаризации признаков получим, что каждый объект  $x$  описывается  $|U| + |I|$  признаками, причём ненулевыми являются ровно два из них: один соответствует номеру пользователя  $u$ , второй — номеру товара  $i$ . Тогда факторизационная машина примет следующий вид:

$$a(x) = w_0 + w_u + w_i + \langle v_u, v_i \rangle.$$

Данная форма полностью соответствует модели (1.1). По сути, факторизационная машина позволяет строить рекомендательные модели на основе большого количества категориальных и вещественных признаков.

Существует несколько методов настройки факторизационных машин, из которых наиболее совершенным считается метод Монте-Карло на основе марковских цепей; реализацию можно найти в библиотеке libFM.

**FFM.** Существует расширение факторизационных машин, позволившее, например, авторам победить в конкурсах Criteo и Avazu по предсказанию кликов по рекламным

объявлениям. В обычных факторизационных машинах у каждого признака имеется всего один скрытый вектор, отвечающий за взаимодействие с остальными признаками. Допустим, что признаки можно некоторым образом сгруппировать — например, в задаче рекомендации музыкальных альбомов в бинарном векторе, отвечающем за композиции, будет стоять несколько единиц, соответствующих всем композициям из альбома. Все единицы из этого вектора можно объединить в одну группу. Расширим модель, введя для каждого признака разные скрытые векторы для взаимодействия с разными группами:

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

где  $f_{j_1}$  и  $f_{j_2}$  — индексы групп признаков  $x_{j_1}$  и  $x_{j_2}$ . Данная модель носит название *field-aware factorization machines* (FFM) [4].

## §1.2 Контентные модели

В коллаборативной фильтрации используется информация о предпочтении пользователей и об их сходствах, но при этом никак не используются свойства самих пользователей или товаров. При этом может быть полезно находить товары, которые своим описанием похожи на товары из историю пользователя; особенно релевантно это может быть для рекомендательных систем контента (музыки, статей, видео), где пользователю, скажем, захочется познакомиться с музыкой, похожей на музыку его любимых исполнителей.

Как правило, это приводит к следующей идее: все товары описываются с помощью векторов (представлений, *embeddings*), и затем измеряется сходство между вектором нового товара и векторами товаров из истории пользователя. Можно вычислять минимальное или среднее расстояние до векторов из истории. Можно обучить линейную модель, которая для данного пользователя предсказывает целевую переменную на основе представления товара:

$$\sum_{i \in I: \exists r_{ui}} (\langle w_u, q_i \rangle - r_{ui})^2 \rightarrow \min_{w_u},$$

и затем с помощью этой модели оценивать, насколько пользователю подойдут другие товары. Можно обучить граф вычислений, который по всем данным о товаре и о пользователе пытается предсказать целевую переменную. Существует много методов, и какой из них подойдёт для данной задачи — заранее предсказать нельзя.

## §1.3 Статистические признаки

Важны и более простые типы факторов: конверсия просмотра данного товара в покупку за всю историю магазина, число покупок данного пользователя в категории данного товара, число покупок данного пользователя и т.д. Если товар или пользователь уже набрали достаточно статистики, то зачастую такие признаки оказываются самыми главными при принятии решения, поскольку уже содержат в себе достаточно информации о предпочтениях.

## Список литературы

- [1] *Gillis, Nicolas and Glineur, François* (2012). Accelerated Multiplicative Updates and Hierarchical Als Algorithms for Nonnegative Matrix Factorization. // *Neural Comput.*, 24, 4, p. 1085–1105.
- [2] *Hu, Yifan and Koren, Yehuda and Volinsky, Chris* (2008). Collaborative Filtering for Implicit Feedback Datasets. // *ICDM '08*.
- [3] *Rendle, S.* (2012). Factorization machines with libFM. // *ACM Trans. Intell. Syst. Technol.* 3, 3, Article 57.
- [4] Field-aware Factorization Machines:  
<http://www.csie.ntu.edu.tw/~r01922136/slides/ffm.pdf>