Применение ансамблей алгоритмов в Рекомендательной системе

Дробин М.Е. (МФТИ ГУ)

<2020-02-12 Ср>

Contents

Abstract

1 Введение

В этой работе на задаче рекомендации фильма по списку уже просмотренных фильмов предложено несколько алгоритмов матричной факторизанции, несколько архитектур нейросетей и градиентного бустинга, и несколько ансамблевых моделей. Качество решения сравнивается с уже работающим алгоритмом в продакшене. Ставится задача улучшить результаты решения этой задачи за счёт использования нейросетевых решений последних лет и пнсамблирования алгоритмов.

2 Постановка задачи

Мы будем рассуждать в терминах пользователей (users, U) и товаров (items, I), но все методы подходят для рекомендаций любых объектов. Будем считать, что для некоторых пар пользователей $u \in U$ и товаров $i \in I$ известны оценки r_{ui} , которые отражают степень заинтересованности пользователя в товаре. Вычисление таких оценок — отдельная тема. Например, в интернет-магазине заинтересованность может складываться из покупок товара и просмотров его страницы, причём покупки должны учитываться с большим весом. В социальной сети заинтресованность в материале может складываться из времени просмотра, кликов и явного отклика (лайки, репосты); это всё тоже должно суммироваться с различными весами. Не будем сейчас останавливаться на этом вопросе, а перейдём к основной задаче. Требуется по известным рейтингам r_{ui} научиться строить для каждого пользо- вателя u набор из k товаров I(u), наиболее подходящих данному пользователю — то есть таких, для которых рейтинг r_{ui} окажется максимальным. Самый распространённый подход в данном случае — сформировать признаки, характеризующие пользователя, товар и их взаимодействия, и обучить модель, которая по данным признакам будет предсказывать рейтинг. Это может быть ранжирующая модель, которая сортирует все товары для данного пользователя; может быть и обычная поточечная модель. Ниже мы рассмотрим некоторые простые методы рекомендаций, оценки которых, как правило, используются в качестве признаков для итоговой модели

2.1 Коллаборативная фильтрация

Методы коллаборативной фильтрации строят рекомендации для пользователя на основе похожестей между пользователями и товарами

2.2 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 мы можем обладать дополнительными дан- ными — например, текстовыми описаниями или категориями товаров, данными из профиля пользователя. Из этих данных можно сформировать признаковое описание пары (пользователь, товар) и пытаться предсказывать рейтинг по этим признакам с помощью каких-либо моделей (линейных, композиций деревьев и т.д.)

3 Описание данных

Данные взяты с соревнования REKKO CHALLENGE на boosters 26.02.2019

Перед соревнование данные предобработали:

- Данные анонимизированы
- Для обучения представлена лишь некоторая выборка из всех данных по времени, пользователям и контенту
- Время выражено в абстрактных единицах, для которых сохранено расстояние и отношение порядка

В общем, данные разделили на несколько файлов:

- информация о транзакциях (покупках и просмотрах по подписке) со временем просмотра по каждои транзакции
- информация об оценках, поставленных пользователями
- информация о фильмах, добавленных пользователями в «Запомненное»
- метаинформация о контенте - анонимизированные признаки фильмов и сериалов

Доступ к контенту в Okko осуществляется через приложение на телевизоре или смартфоне, либо через веб-интерфейс. Контент можно взять в аренду®, купить(P) или посмотреть по оформленной подписке(S). Организатор соревнования предоставил данные о просмотрах за N дней ($N > 60$), дополнительно была доступна информация о проставленных рейтингах и добавлениях в закладки. Стоит иметь в виду одну важную деталь, если пользователь посмотрел один фильм несколько раз или несколько

серий сериала, то в табличке будет зафиксирована лишь дата последней транзакции и суммарное время потраченное на единицу контента.

Было предоставлено порядка 10 миллионов транзакций, 450 тысяч оценок и 950 тысяч фактов добавлений в закладки по 500 тысячам пользователей.

В выборке содержатся не только активные пользователи, но и пользователи посмотревшие пару фильмов за весь период.

Каталог Okko содержит контент трех типов: фильмы(movie), сериалы(series) и многосерийные фильмы($\text{multipart}_{\text{movie}}$), всего 10200 объектов. По каждому объекту был доступен набор анонимизированных атрибутов(attributes) и признаков(feature_1,... ,feature_5), доступность по подписке, аренде или покупке и длительность.

4 Целевая переменная и метрика

В задаче требовалось предсказать множество контента, который пользователь потребит за следующие 60 дней. Считается, что пользователь потребит контент, если он:

- Купит его или возьмет в аренду
- Посмотрит больше половины фильма по подписке
- Посмотрит больше трети сериала по подписке

$MNAP@20 = \frac{1}{|U|}\sum_u \frac{1}{min(n_u,20)}\sum_{i=1}^{20} r_u(i)p_u@i$, где

- $p_u@k = \frac{1}{k}\sum_{i=1}^{k} r_u(i)$
- $r_u i$ - потребил ли пользователь u контент, предсказанныиему на месте i (1 либо 0)
- n_u - количество элементов, которые пользователь потребил за тестовыи период
- U - множество тестовых пользователей

Большинство пользователь досматривают фильмы до конца, поэтому по транзакциям доля положительного класса составляет 65%. Оценка качества работы алгоритма производилась по подмножеству из 50 тысяч пользователей из представленной выборки.

На момент сбора выборки продуктовая модель рекомендация уже собрана и показывает на тестовой выборке 0.062 NMAP

5 Эксперименты

5.1 Описание первого решения

В качества начального решения воспользовались моделью K близжайших соседей с расстоянием tf-idf. Модель обучили только на данных про рейтинг. Получили предсказания для 27% пользователей из тестовой выборки. Результатат - 0.0046 NMAP

5.2 Некоторые улучшения

Попробовали обучить модели SAR, AlternatingLeastSquare [1] на полных данных - рейтинг, закладки, транзакции. Лучшая модель модель ALS 200 показала 0.0168 NMAP на тестовой выборке

Также улучшил результаты удаление дублирующихся предсказаний для каждого пользователя и повторное предсказанние удаленных другой моделью. Уверенность в том, какой из дубликатов удалить делаю на основе предсказаний лучшей модели - получился стекинг SAR и ALS. Результат - 0.0279 NMAP.

5.3 Нейросетевой подход

Задачу рекомендации релевантных фильмов и сериалов по историческим данным можно представить как задачу генерации последовательности чисел. Числа - это идентификаторы фильмов и сериалов. Последовательность идентификаторов отсортирована по дате просмотра.

С такой задачей справляются рекурентные нейросети RNN, LSTM, GRU, CuDNNGRU [2]

- Было произведена аугментация данных: отсеяны пользователи, у которых в истории меньше 10 просмотренных фильмов, а для оставшихся вся последовательность просмотренных фильмов была разделена на последовательности по 10 фильмов.

- Задача нейросети - по последовательности фильмов предсказать, какой из 10200 фильмов пользователь посмотрит следующим. Таким образом, получается задача классификации на 10200 классов. В нейросети это сформулировано в виде последнего линейного слоя с размерностью выхода в 10200 и softmax на конце.

- Функция потерь - кроссэнтропия

- К каждому фильму применяется обучаемый на этой же выборке эмбединг слой размерностью 128
- Размерность входных данных: (10, 128)
- Улучшило результат уменьшение коэффициента скорости обучения в 100 раз при попадании алгоритма на плато в пространстве весов [3]
- Получилось предсказать рекомендации для 31 тыс. пользователей из 50 тыс. в тестовой выборке
- Результат 0.032 NMAP на тестовой выборке

5.3.1 Архитектура нейросети

- Эмбединг слой
- Пространственный одномерный дропаут
- Двухслоная CuDNNGRU в 128 нейронов
- Конкатенация из map pool и avg pool и выхода CuDNNGRU
- Батчнормализация
- Выходной линейный слой с softmax

5.4 Ансамблевое решение

6 Обзор литературы

Рекомендательные системы - ключевые технологии для задачи предложения пользователю релевантный ему контент, услугу или товар на основе интересов пользователя и его покупательной истории [4] . Более чем десяти - летие - возросло использование рекомендательных систем и глобальные IT корпорации, такие как Google, Facebook, Netflix, Amazon начали активно использовать рекомендательные системы для увеличения продаж [5, 6, 7]. Рекомендательные системы разделяют на коллаборативную фильтрацию и контентные одели.[4, 8, 9]

В статье[10] описывается подход команды D2KLab к RecSysChallenge 2018, который фокусируется на задаче формирования плейлиста. Они предложили ансамблевую стратегию различных рекуррентных нейронных

сетей, использующих предварительно обученные вложения, представляющие треки, исполнителей, альбомов и заголовки в качестве входных данных. Они также использовали тексты песен, из которых они извлекли семантические и стилистические особенности, которые подали в сеть для творческого трека. RNN изучает вероятностную модель из этих последовательностей элементов в списке воспроизведения, которая затем используется для прогнозирования наиболее вероятных треков, которые будут добавлены в список воспроизведения. Что касается плейлистов без треков, то они реализовали резервную стратегию под названием title2rec, которая генерирует рекомендации, используя только названия песен. Они оптимизировали RNN, Title2Rec и ансамблевый подход для набора валидации, настраивая гиперпараметры, такие как optimizeralgorithm, скорость обучения и стратегию генерации. Этот подход эффективен при прогнозировании треков для плейлиста и гибко включает в себя различные типы входных данных, но он также является вычислительно сложным на этапе обучения.

В [2] описывается решение для Rec-Sys Challenge 2019, которое фокусируется на задаче прогнозирования последнего клика в сеансовом взаимодействии. Команда предложила ансамблевое решение, включающее матричную факторизацию для моделирования взаимодействия пользователя-элемента и сессионно-ориентированную модель обучения, реализованную с помощью рекуррентной нейронной сети. Этот метод, по-видимому, эффективен при прогнозировании последнего клика-аута, набравшего 0,60277 среднего значения в локальном тестовом наборе.

7 Список литературы

References

[1] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*, page nil, 12 2008.

[2] Andrea Fiandro, Giorgio Crepaldi, Diego Monti, Giuseppe Rizzo, and Maurizio Morisio. Predict your click-out: Modeling user-item interactions and session actions in an ensemble learning fashion, 2020.

[3] Leslie N. Smith. Cyclical learning rates for training neural networks, 2015.

[4] F.O. Isinkaye, Y.O. Folajimi, and B.A. Ojokoh. Recommendation sys-

tems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16(3):261–273, Nov 2015.

[5] Maryam Jallouli, Sonia Lajmi, and Ikram Amous. Designing recommender system: Conceptual framework and practical implementation. *Procedia Computer Science*, 112:1701–1710, 2017.

[6] Dietmar Jannach, Paul Resnick, Alexander Tuzhilin, and Markus Zanker. Recommender systems—. *Communications of the ACM*, 59(11):94–102, Oct 2016.

[7] Bhavik Pathak, Robert Garfinkel, Ram D. Gopal, Rajkumar Venkatesan, and Fang Yin. Empirical analysis of the impact of recommender systems on sales. *Journal of Management Information Systems*, 27(2):159–188, Oct 2010.

[8] and . : . , 17(3):85–103, 08 2012.

[9] Jieun Son, Seoung Bum Kim, Hyunjoong Kim, and Sungzoon Cho. Review and analysis of recommender systems. *Journal of Korean Institute of Industrial Engineers*, 41(2):185–208, Apr 2015.

[10] Diego Monti, Enrico Palumbo, Giuseppe Rizzo, Pasquale Lisena, Raphaël Troncy, Michael Fell, Elena Cabrio, and Maurizio Morisio. An ensemble approach of recurrent neural networks using pre-trained embeddings for playlist completion. In *Proceedings of the ACM Recommender Systems Challenge 2018 on - RecSys Challenge '18*, page nil, - 2018.

8 Ссылки

- Основные понятия и обозначения в машинном обучении. Воронцов К.В.
- Матричные разложения и рекомендательные системы
- Решение в Rekko Challenge. 2е место. Блендинг