АЛГОРИТМЫ АНАЛИЗА ПОЛЬЗОВАТЕЛЬСКИХ ПРЕДПОЧТЕНИЙ И ГЕНЕРАЦИИ РЕКОМЕНДАЦИЙ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

Александров Н. А., Нестеренков С. Н.
Кафедра программного обеспечания информационных технологий,
Белорусский государственный университет информатики и радиоэлектороники
Минск, Республика Беларусь
E-mail: nekitale@gmail.com, s.nesterenkov@bsuir.by

В работе рассмотрены алгоритмы, позволяющие проанализировать пользовательские предпочтения, что позволит составить список персональных рекомендаций. Также рассмотрена классификация алгоритмов и возможные варианты их использования.

Введение

Рекомендательные системы предназначены для точного прогнозирования предпочтений пользователей и предоставления персонализированных рекомендаций по товарам или услугам. Они помогают потребителям находить именно те предложения, которые наиболее соответствуют их интересам и потребностям.

Сегодня такие системы внедрены повсеместно и стали неотъемлемой частью большинства крупных интернет-сервисов. Онлайн-магазины, стриминговые платформы, новостные порталы и социальные сети активно используют алгоритмы рекомендаций для улучшения пользовательского опыта, предлагая контент или товары, которые могут заинтересовать конкретного пользователя на основе его предыдущих действий, предпочтений и анализа поведения.

I. Виды рекомендательных систем

Алгоритмы анализа пользовательских предпочтений и генерации рекомендаций можно разделить на 3 категории.

1. Коллаборативная фильтрация – это один из самых популярных методов рекомендаций, основанный на анализе предпочтений и поведения пользователей. Существует два основных типа коллаборативной фильтрации:
   1.1. User-based: Рекомендации строятся на основе сходства между пользователями. Например, если пользователи A и B ставят похожие оценки одинаковым фильмам, то фильмы, которые понравились A, могут быть рекомендованы B. Предполагается, что пользователи с похожими вкусами будут одинаково оценивать похожие элементы. Такой подход используется, например, в системах рекомендаций книг на основе того, что другие пользователи с аналогичными предпочтениями уже прочитали и оценили.
   1.2. Item-based: В этом случае рекомендации строятся на основе сходства между элементами. Если, например, фильмы X и Y часто получают одинаковые оценки, то пользователю, которому понравился фильм X, может быть предложен фильм Y. Такой подход применим в музыкальных сервисах, где песни рекомендуются на основе того, что другие пользователи часто прослушивают их вместе.

   Коллаборативная фильтрация обладает рядом преимуществ, в том числе независимостью от анализа содержимого элементов, что делает её универсальной. Однако метод сталкивается с проблемой «холодного старта» – нехваткой данных для новых пользователей или элементов.
2. Матричная факторизация (Matrix Factorization) – это метод, используемый для сокращения размерности данных и выявления скрытых факторов, влияющих на предпочтения пользователей. Одним из известных алгоритмов этого типа является SVD (Разложение сингулярных значений). Матричная факторизация позволяет обнаружить скрытые закономерности в больших матрицах, где строки – это пользователи, а столбцы – элементы. Это улучшает качество рекомендаций, анализируя взаимодействия между пользователями и элементами на более глубоком уровне.

II. Алгоритм коллаборативной фильтрации на основе сходства элементов

В ситуации, когда имеются данные о предпочитаемых пользователем товарах, но недостаточно информации о схожести пользователей для применения методов факторизации можно использовать коллаборативную фильтрацию на основе схожести товаров. Для этого потребуется представить характеристики товаров в виде текстового описания товара. В дальнейшем, тексто-

вые описания преобразуются в вектора, используя алгоритм TF-IDF (см. формула 1).

$$TF(t,d) = \frac{number\ of\ times\ t\ appears\ in\ d}{total\ number\ of\ terms\ in\ d}$$
$$IDF(t) = \log(\frac{N}{1+df}) \quad (1)$$
$$TF - IDF(t,d) = TF(t,d) * IDF(t)$$

После проведения векторизации требуется вычислить косинусное расстояние. Данное значение лежит в диапазоне от -1 до 1 и чем оно больше, тем больше схожесть двух сравниваемых товаров:

$$cosine\ similarity = S_c(A,B) = \frac{A \cdot B}{\|A\|\|B\|} = \\ = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}} \quad (2)$$

Данный подход не требует больших вычислительных мощностей и может работать с любыми данными, однако ему требуется наличие большого количества данных для точной работы и он чувствителен к семантике описания товаров.

III. Алгоритм матричной факторизации

User-item matrix – это таблица, в которой каждая строка представляет собой пользователя, каждый столбец – товар, а каждая ячейка содержит информацию о взаимодействии между пользователем и товаром. Например, в ячейке может быть указано количество раз, которое пользователь купил товар, или оценка, которую пользователь поставил товару. User-item matrix используется в рекомендательных системах для определения предпочтений пользователей и предложения им наиболее подходящих товаров. Очевидно, что данная матрица - очень большая и разряженная. Эту проблему можно решить при помощи матричной факторизации.

Сингулярное разложение матриц (SVD-разложение) – это представление прямоугольной матрицы в виде произведения нескольких матриц особого вида. Цель этого разложения – это упростить некоторые вычисления, которые будут осуществляться над матрицей. Цель в контексте рекомендательных систем – чтобы по полученному представлению мы могли получить прогноз оценки пользователя.

Любая прямоугольная матрица размеров N на M представляется в виде: $Mat_{N*M} = U \sum V^*$, где U и V – унитарные матрица порядка N и M соответственно (при этом матрица $V*$ – это сопряженно-транспонированная матрица к V), $\sum$ – матрица размеров $N \times M$, на главной диагонали которой лежат неотрицательные числа.

Для изучения факторных векторов (p_u и q_i) система минимизирует ошибку регуляризованного квадрата на множестве известных рейтингов:

$$Mat_{N*M} = U \sum V^* min_{u_i,v_j} \sum_{i,j} (mat_{i,j} - u_i * v_j)$$

, где u_i,v_j строки и столбцы матриц U и $V*$ соответственно, $mat_{i,j}$ – известные элементы матрицы.

Минимизация может производиться различными математическими методами, например, с помощью градиентного спуска.

Однако у SVD есть существенные недостатки: из-за большого количества пропусков в матрице полученное решение будет слишком шумным, а кроме того, его придется каждый раз рассчитывать заново при добавлении новых пользователей или объектов.

IV. Заключение

В завершении отметим, что выбор алгоритма анализа зависит он имеющихся данных и требований. Каждый из них обладает своими плюсами и минусами. Зачастую, компании прибегают к комбинированию разных методов для реализации более адаптивной системы, которая сможет выбирать методы реализации на основе текущего состояния данных.

Следует также отметить, что внедрение системы рекомендаций требует не только выбора подходящего алгоритма, но и тщательной настройки модели, работы с данными и постоянной валидации результатов. Важно не только разработать систему, но и проводить регулярные эксперименты с целью её оптимизации, чтобы она максимально точно удовлетворяла нужды пользователей и бизнеса. В перспективе, использование продвинутых методов машинного обучения, таких как глубокое обучение и нейронные сети, может ещё больше улучшить качество рекомендаций, особенно в сложных и динамичных доменах, таких как электронная коммерция или потоковое видео.

V. Список литературы

1. Recommender Systems Handbook 3rd ed. / F. Ricci, L. Rokach, B. Shapira // Springer – 2022.
2. Practical Recommender Systems / K. Falk // Manning – 2019.
3. Recommender Systems: Legal and Ethical Issues (The International Library of Ethics, Law and Technology, 40) / S. Genovesi, K. Kaesling, S. Robbins // Springer – 2023.
4. Использование алгоритмов Big Data для формирования индивидуальных музыкальных рекомендаций / И. П. Наденeнко, С. Н. Нестеренков // Восьмая Международная научно-практическая конференция «BIG DATA and Advanced Analytics. BIG DATA и анализ высокого уровня» – Минск – 2022 год.
5. Поиск визуально подобных изображений на основе машинного обучения / М. М. Гресик, С. Н. Нестеренков // Девятая Международная научно-практическая конференция «BIG DATA and Advanced Analytics. BIG DATA и анализ высокого уровня» – Минск – 2023 год.