УДК 004.9

Исследование алгоритмов рекомендательных систем

Е.Д. Чепикова, Е.О. Савкова, М.В. Привалов
Донецкий национальный технический университет

Чепикова Е.Д., Савкова Е.О., Привалов М.В. Исследование алгоритмов рекомендательных систем. *Выполнен анализ типов рекомендательных систем и алгоритмов использующихся для систем с коллаборативной фильтрацией. Разработана алгоритмическая часть гибридной рекомендательной системы на основе коллаборативной фильтрации. Предложено решение проблемы «холодного старта» для только что зарегистрировавшихся пользователей и для новых фильмов, добавленных в систему.*

Постановка проблемы

Сейчас пришло время развития рекомендательных систем (РС). Пользователи дорожат своим свободным временем и хотят потратить его с пользой. Этому способствуют системы рекомендаций, где система сама выбирает и предлагает пользователю музыку, фильмы, книги и прочее. Такие же системы используют магазины, предлагающие товары покупателям. Подбор товаров позволяет найти для пользователя подходящую продукцию и не потерять покупателя.

Экономия времени и стремление к удобству рождают потребность в таких системах. Для разработки математического обеспечения такой системы нужно выполнить следующие этапы:

– выбрать тип разрабатываемой РС;

– сравнить алгоритмы и выбрать наиболее быстрый, дающий более точные результаты (модификацию алгоритма);

– решить проблему подбора рекомендаций для новых пользователей и новых фильмов.

Цель работы

В данной работе целью является исследование типов РС, их алгоритмов с точки зрения скорости решения задачи и точности получаемых результатов.

Постановка задачи исследования

В данной работе необходимо определить последовательность алгоритмов для реализации рекомендательной Web-ориентированной системы, содержащей медиа-контент, а также решить проблему «холодного старта» для пользователей, только что прошедших регистрацию и новых фильмов, добавленных в систему.

Решение задач и результаты исследований

Рассмотрим основные типы РС. Традиционно их разделяют на четыре типа.

Коллаборативная фильтрация

Коллаборативная фильтрация (англ. collaborative filtering) вырабатывает рекомендации, основанные на модели предшествующего поведения пользователя. Эта модель может быть построена исключительно на основе поведения данного пользователя или — что более эффективно — с учетом поведения других пользователей со сходными характеристиками [1].

Плюсы**:** Теоретически высокая точность. Минусы**:** Высокий порог входа: не зная ничего об интересах пользователя, рекомендации практически бесполезны, многие пользователи будут просто сразу уходить [2].

РС, основанные на контенте (content-based)

В контентных рекомендательных системах вывод о полезности u(h,s) товара s для потребителя h делается, исходя из полезности (h,si), присвоенной потребителем товарам, сходным с товаром s. Например, в системе, рекомендующей фильмы, для того, чтобы рекомендовать фильмы потребителю h, контентная РС пытается найти сходство между фильмами, высоко оцененными потребителем ранее (общие актеры, режиссеры, жанры и т.д.) И только фильмы, обладающие высокой степенью общности с предпочтениями потребителя, будут рекомендованы [3].

Плюсы**:** Можно делать рекомендации даже незнакомым пользователям, тем самым вовлекая их в сервис. Возможность рекомендовать те объекты, которые еще не были никем оценены. Минусы**:** Точность сильно падает, время разработки немного возрастает.

РС, основанные на знаниях (knowledge-based)

Рекомендации, основанные на знаниях о предметной области. Часто предыдущий тип (content-based) определяют как частный случай knowledge-based, где знаниями являются сведения о товаре, но content-based имеет такую широкую распространенность, что имеет смысл выносить его в отдельный тип. Эти самые дополнительные знания позволяют делать рекомендации не основываясь на «похожести» чего-либо, а с более сложными условиями [2]. Плюсы**:** Возможность исключить ситуацию рекомендации товаров уже неактуальных для данного пользователя. Минусы**:** Высокая сложность разработки и сбора данных.

Гибридные (hybrid) РС

Комбинируют различные подходы, что позволяет избежать ограничений, свойственных каждой системе.

Проанализировав плюсы и минусы можно сделать выбор в пользу гибридных систем. Для нашей задачи будем использовать метод коллаборативной фильтрации, и предложим своё решение проблемы «холодного старта» для пользователей и новых фильмов.

Рассмотрим основные алгоритмы, которые используется для реализации этого метода.

Алгоритм основанный на коэффициенте корреляции.

В [4] предложено следующее решение задачи коллаборативной фильтрации.

Имеется вектор предпочтений для каждого пользователя (строки матрицы R) и вектор оценок пользователей для каждого продукта (столбцы матрицы R). Прежде всего оставим в этих векторах только те элементы, для которых нам известны значения в обоих векторах, т.е. оставим только те продукты, которые оценили оба пользователя, или только тех пользователей, которые оба оценили данный продукт. В результате нам просто нужно определить, насколько похожи два вектора вещественных чисел. Для этого необходимо подсчитать коэффициент корреляции

$$w_{i,j} = \frac{\sum_a (r_{i,a}-\bar{r_i})(r_{j,a}-\bar{r_j})}{\sqrt{\sum_a (r_{i,a}-\bar{r_i})^2}\sqrt{\sum_a (r_{j,a}-\bar{r_j})^2}}, \quad (1)$$

где $\bar{r_i}$ — средний рейтинг, выставленный пользователем i. Иногда пользуются так называемой «косинусной похожестью», используя косинус угла между векторами

$$w_{i,j} = \frac{\sum_a r_{i,a} r_{j,a}}{\sqrt{\sum_a r_{i,a}{}^2}\sqrt{\sum_a r_{j,a}{}^2}}. \quad (2)$$

Но для того, чтобы косинус хорошо работал, желательно всё равно сначала вычесть среднее по каждому вектору, так что в реальности это та же самая метрика.

Для того чтобы воспользоваться этими коэффициентами схожести необходимо приблизить новый рейтинг как средний рейтинг данного пользователя плюс отклонения от среднего рейтингов других пользователей, взвешенных этими самыми весами

$$\dot{r}_{i,a} = \bar{r}_i + \frac{\sum_j (r_{j,a}-\bar{r_j}) w_{i,j}}{\sum_j |w_{i,j}|}. \quad (3)$$

Для ускорения поиска ближайших соседей автор работы [4] предлагает воспользоваться k-d-деревьями или локально-чувствительным хешированием (locally sensitive hashing).

Плюсом данного алгоритма является достаточно простая реализация. Однако этот алгоритм является устаревшим, на его основе работали первые РС, сейчас же существуют другие алгоритмы, которые быстрее справляются со своей задачей и дают более точные результаты.

Алгоритм наивных сетей Байеса

Теорема Байеса — одна из основных теорем элементарной теории вероятностей, которая определяет вероятность наступления события в условиях, когда на основе наблюдений известна лишь некоторая частичная информация о событиях.

Согласно [5] условная вероятность P(A|H) события A при условии наступления события H вычисляются по формуле

$$\mathrm{P}(A|H) = \frac{\mathrm{P}(A,H)}{\mathrm{P}(H)}, \quad (4)$$

где P (A) и P(H) – это вероятности каждого события по отдельности, а P (A,H) является совместной вероятностью A и H. Значит, совместную вероятность можно выразить двумя способами (5):

$$P(A,H) = P(A|H) * (H) = P(H|A) * P(A) \quad (5)$$

Тогда теорема Байеса имеет вид

$$\mathrm{P}(A|H) = \frac{\mathrm{P}(H|A)\mathrm{P}(A)}{\mathrm{P}(H)} \quad (6)$$

Перепишем теорему Байеса в других обозначениях

$$p(\theta|D) = \frac{p(\theta)p(D|\theta)}{p(D)} \quad (7)$$

$p(\theta|D)$— это то, что мы хотим найти, распределение вероятностей параметров модели

после того, как мы приняли во внимание данные; это называется апостериорной вероятностью. В источнике [6] говорится, что эту вероятность, как правило, напрямую не найти, и здесь как раз и нужна теорема Байеса. $p(D|\theta)$ — это так называемое правдоподобие, вероятность данных при условии зафиксированных параметров модели; это как раз найти обычно легко, собственно, конструкция модели обычно в том и состоит, чтобы задать функцию правдоподобия. А $p(\theta)$ — априорная вероятность, она является математической формализацией нашей интуиции о предмете, формализацией того, что мы знали раньше, ещё до всяких экспериментов.

Алгоритм SVD

SVD (Singular Value Decomposition), переводится как сингулярное разложение матрицы. В теореме о сингулярном разложении утверждается, что у любой матрицы **A** размера $n \times m$ существует разложение в произведение трех матриц: U,D и V^T

$$A = U * D * V^T \quad . \qquad (8)$$

Матрицы U и V ортогональные, а D — диагональная (хотя и не квадратная). Алгоритм достаточно простой, но позволяет не только предсказывать оценки. С его помощью мы можем по истории пользователей выявлять скрытые признаки объектов и интересы пользователей.

Итак, чтобы предсказать оценку пользователя U для фильма I, мы берем некоторый вектор p_u (набор параметров) для данного пользователя и вектор для данного фильма q_i. Их скалярное произведение и будет нужным нам предсказанием

$$\dot{r}_{ui} = \langle pu, qi \rangle \qquad (9)$$

Однако мы не можем найти SVD-разложение матрицы, т.к. мы не знаем саму матрицу. Но мы хотим воспользоваться этой идеей и придумать модель предсказания, которая будет работать сходным с SVD образом. Хорошим образцом такой модели является модель, описанная в блоге компании Surfingbird [7].

Введем так называемые базовые предикторы $b_{i,a}$, которые складываются из базовых предикторов отдельных пользователей b_i, и отдельных продуктов b_a, а также просто общего среднего рейтинга по базе μ

$$b_{i,a}, = \mu + b_i + b_a \qquad (10)$$

где μ — средний рейтинг по базе; b_i — средний рейтинг каждого i пользователя; b_a — средний рейтинг каждого a продукта.

Для определения только базовых предикторов необходимо найти такие μ, b_i, b_a, для которых $b_{i,\,a}$ лучше всего приближают имеющиеся рейтинги. Затем можно будет добавить собственно факторы. Поскольку теперь, когда сделана поправка на базовые предикторы, остатки будут сравнимы между собой, вполне возможно будет получить разумные значения для факторов

$$\dot{r}_{i,a} = \mu + b_i + b_a + v_a^T u_i, \qquad (11)$$

где v_a — вектор факторов, представляющий продукт a; u_i — вектор факторов, представляющий пользователя i.

С учётом (10) формулировка исходной задачи принимает следующий вид: необходимо найти наилучшие предикторы, которые приближают величину $\dot{r}_{i,a}$.

Лучшими будут те предикторы, которые дают минимальную ошибку, определяемую по формуле

$$L(\mu, b_i, b_a, v_a, u_i) = \sum_{(i,a)\in D} (r_{i,a} - \dot{r}_{i,a})^2 = \sum_{(i,a)\in D} (r_{i,a} - \mu - b_i - b_a - v_a^T u_i)^2 \qquad (12)$$

Функцию L(μ, b_i, b_a, v_a, u_i) минимизируем методом градиентного спуска: берем частные производные по каждому аргументу и двигаемся в сторону, обратную направлению этих частных производных. Для компенсирования эффекта переобучения добавляется параметр регуляризации. Иными словами, накладывается штраф за слишком большие значения обучаемых переменных. Как показано в работе [8], для этого можно просто добавить в функцию ошибки сумму квадратов всех факторов и предикторов :

$$b_*, q_*, p_* = \arg\min = \sum_{(i,a)} (r_{i,a} - \mu - b_i - b_a - q_a^T p_i)^2 + \lambda(\sum_i b_i^2 + \sum_a b_a^2 + \|q_a\|^2 + \|p_a\|^2), \qquad (13)$$

где λ — параметр регуляризации.

Если взять у функции ошибки в формуле (12) частные производные по каждой из оптимизируемых переменных, получим простые правила для градиентного (стохастического) спуска

$$\begin{aligned} b_i &= b_i + \gamma(e_{i\,a,} - \lambda b_i), \\ b_a &= b_a + \gamma(e_{i\,a,} - \lambda b_a), \\ q_{a,j} &= q_{a,j} + \gamma(e_{i,a} p_{i,j} - \lambda q_{a,j}), \\ p_{i,j} &= p_{i,j} + \gamma(e_{i,a} p_{i,j} - \lambda p_{i,j}), \end{aligned} \qquad (14)$$

для всех j, где $e_{i,a} = r_{i,a} - \dot{r}_{i,a}$ — ошибка на данном тестовом примере, а γ — скорость обучения

Для компенсирования эффекта переобучения добавляется параметр регуляризации. Иными словами, накладывается штраф за слишком большие значения обучаемых переменных [9].

Исходя из данных источника [7], SVD является самым быстрым алгоритмом РС. Существуют три его разновидности: SVD, SVD++, timeSVD. При этом SVD немного проигрывает в скорости двум другим. В статье описывалась реализация SVD++, его мы и предлагаем использовать в нашей системе.

Решение проблемы «холодного старта»

Проблема «холодного старта» делится на «холодный старт» для пользователей и «холодный старт» для фильмов. Рассмотрим решение обоих вариантов. Проанализировав источник [10], предлагается собственное решение проблемы.

«Холодный старт» для пользователей. Эта проблема состоит в том, что только что зарегистрировавшимся пользователям нечего порекомендовать, так как они не успели выставить оценок фильмам. Логично, что чем больше оценок пользователь поставит, тем точнее система будет выдавать результаты.

Для решения возьмём набор обычных базовых данных при регистрации – возраст пользователя (из даты его рождения), местоположение пользователя, и его гендерную принадлежность. Система должна осуществить подбор векторов пользователей, которые имеют схожее геоположение (одну и ту же страну, один и тот же регион), такой же пол и входят в те же возрастные рамки. Возрастные рамки предлагается поделить на промежутки: до 12 лет, 12-18, 18-25, 25-40, 40-55, старше 55. После этого будут отобраны фильмы которым данные пользователи поставили максимальные оценки и предложены к просмотру данному, только что вошедшему в систему пользователю. После проставления собственных оценок эти данные будут заменяться результатами алгоритма SDV++.

Проблема «холодного старта» для новых фильмов, добавленных в систему, заключается в том, что у этих фильмов пока ещё нет рейтинговых оценок, следовательно их некому рекомендовать.

Для решения этой проблемы следует провести корреляционно-регрессионный анализ для таких данных, как жанры, актёры и режиссёры. Для рекомендаций новых фильмов нужно выделить те из них, которые наиболее коррелируют по выше названным данным с тем, что предпочитает пользователь. Результаты по подобранным новинкам системы следует выводить перед результатами алгоритма SVD++.

Выводы

В результате проведенного анализа типов рекомендательных систем для применения в Web-ориентированной системе подбора фильмов, был выбран гибридный метод, дающий наиболее точные результаты. Метод включает в себя коллаборативную фильтрацию и собственную разработку решения проблемы подбора медиа-контента для новых пользователей системы. После исследования алгоритмов коллаборативной фильтрации был выбран алгоритм SVD++, который является модификацией самого быстрого и широко используемого алгоритма SVD, разработанного для разреженных матриц рейтингов. Предложено решение для проблемы «холодного старта» новых пользователей, заключающееся в подборе фильмов пользователям на основе геоположения и возрастной категории, и новых фильмов, заключающееся в подборе наиболее коррелированных по жанру с предпочитаемыми жанрами пользователя.

Направлением дальнейших исследований в данной работе является модификация метода Singular Value Decomposition с целью повышения точности результатов, что планируется достигнуть за счёт усложнённого формирования правил базовых предикторов.

Литература

1. Тим Джонс - Рекомендательные системы: Часть 1. Введение в подходы и алгоритмы [Электронный ресурс] // Режим доступа:http://www.ibm.com/developerworks/ru/library/os-recommender1/index.html

2. Рекомендательные системы [Электронный ресурс] // Режим доступа: http://vas3k.ru/blog/355/

3. Gediminas Adomavicius, Alexander Tuzhilin – IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 6, 2005

4. Блог компании Surfingbird – Рекомендательные системы: user-based и item-based [Электронный ресурс] // Режим доступа: http://habrahabr.ru/company/surfingbird/blog/139518/

5. Вентцель Е.С. Теория вероятностей: Учеб. для вузов. — 6-е изд. стер. — М.: Высш. шк., 1999.— 576 с.

6. Блог компании Surfingbird – Рекомендательные системы: теорема Байеса и наивный байесовский классификатор [Электронный ресурс] // Режим доступа: http://habrahabr.ru/company/surfingbird/blog/15020/

7. Блог компании Surfingbird – Рекомендательные системы: SVD и базовые предикторы [Электронный ресурс] // Режим доступа: http://habrahabr.ru/company/surfingbird/blog/140555/

8. Королева Д.Е., Филиппов М.В. – Анализ алгоритмов обучения коллаборативных рекомендательных систем, 2013

9. Рекомендательные системы: You can (not) advise [Электронный ресурс] // Режим доступа: http://habrahabr.ru/post/176549

10. Блог компании Surfingbird – Рекомендательная система: введение в проблему холодного старта [Электронный ресурс] // Режим доступа: http://habrahabr.ru/company/surfingbird/blog/168733

Чепикова Е.Д., Савкова Е.О., Привалов М.В. Исследование алгоритмов рекомендательных систем.*Выполнен анализ типов рекомендательных систем и алгоритмов использующихся для систем с коллаборативной фильтрацией. Разработана алгоритмическая часть гибридной рекомендательной системы на основе коллаборативной фильтрации. Предложено решение проблемы «холодного старта» для только что зарегистрировавшихся пользователей и для новых фильмов, добавленных в систему.*

Ключевые слова. Алгоритмы рекомендательных систем, гибридный метод, коллаборативная фильтрация, проблема «холодного старта», применение алгоритма SVD (Singular Value Decomposition) в рекомендательных системах.

Чепікова О.Д., Савкова О.Й., Привалов М.В. Дослідження алгоритмів рекомендаційних систем. *Виконано аналіз типів рекомендаційних систем і алгоритмів,що використуються для систем з коллаборативною фільтрацією. Розроблено алгоритмічну частину гібридної рекомендаційної системи на основі коллаборативної фільтрації. Запропоновано вирішення проблеми «холодного старту» для користувачів, що щойно зареєструвалися і для нових фільмів, доданих в систему.*

Ключові слова: Алгоритми рекомендаційних систем, гібридний метод, коллаборативна фільтрація, проблема «холодного старту», застосування алгоритму SVD (Singular Value Decomposition) в рекомендаційних системах.

Elena Chepikova, Elena Savkova, Maksim Privalov, Research of recommender systems' algorithms. *Made analysis of the existing types of recommender systems and performed investigation of the algorithms used in the systems which are based on collaborative filtering approach. Developed set of the algorithms designed to be used in hybrid recommender system. Proposed solution of the "cold start problem" that could be applied to initially registered users and new films added to the system.*

Keywords: Algorithms of recommender systems, hybrid method, collaborative filtering, the problem of "cold start", using of the algorithm SVD (Singular Value Decomposition) in recommendation systems.

Статья поступила в редакцию 20.05.2016
Рекомендована к публикации д-ром техн. наук В.Н. Павлышом