УДК 004.67

БОЛЬШИЕ ДАННЫЕ И РЕКОМЕНДАТЕЛЬНЫЕ СИСТЕМЫ

Диденко А.А., аспирант гр. ИВа-191, I курс
Научный руководитель: Пимонов А.Г., д.т.н., профессор
Кузбасский государственный технический университет имени Т.Ф. Горбачева
г. Кемерово

В современном обществе все чаще возникают проблемы анализа большого объема данных и предоставления рекомендаций на его основе. Для решения подобных проблемы служат так называемые рекомендательные системы. Основная задача рекомендательной системы – это информирование пользователя о товарах, решениях, услугах, которые могут быть ему наиболее интересны в данный момент времени. В основном подобные системы сейчас используются в сфере маркетинга и рекламы. Однако рекомендательные системы могут быть применены и в других сферах. Например, в медицине, образовании, производстве, менеджменте и многих других, где они могут быть использованы для предварительного анализа входных данных по той или иной проблеме и выводе информации по возможным вариантам ее решения, что упростит пользователю выбор окончательного способа решения. В этой статье будут рассмотрены задачи, основные характеристики, алгоритмы, проблемы и решения рекомендательных систем.

Любую рекомендательную систему можно описать по набору характеристик, таких как:

- **Объект рекомендации**, или что рекомендуется. В зависимости от рекомендательной системы предлагаться могут: различные товары (Amazon, AliExpress), новости (Yandex, Google), решения, статьи, мероприятия, изображения, видео (YouTube, Netflix), люди (Facebook, VK), музыка (VK), игры (Steam) и многое другое [1].
- **Цель рекомендации**, или зачем рекомендуется. В зависимости от рекомендательной системы основными целями могут быть: продажа того или иного товара или услуги, информирование пользователя, предоставление решений по проблеме, установка контактов, обучение.
- **Источник рекомендации**, или кто рекомендует. В зависимости от рекомендательной системы основными источниками рекомендаций могут быть: схожие по интересам пользователи (Steam, Netflix), в целом аудитория той или иной системы (Flamp, YouTube), небольшое сообщество экспертов для сложных специализированных товаров, услуг, решений (рекомендательные экспертные системы) [1].
- **Контекст рекомендации**, или в какой момент пользователь видит рекомендацию. Пользователи могут получать рекомендации во время просмотра товаров, общения с людьми, выполнения той или иной работы, прослушивания музыки, просмотра видео.

- **Формат рекомендации**, или в каком виде пользователь получает рекомендацию. Пользователи могут получать рекомендации в формате: всплывающих окон, ленты в краю экрана, отсортированного списка.
- **Степень персонализации**, или на сколько система анализирует каждого конкретного пользователя перед тем, как предложить ту или иную рекомендацию. Всего есть три степени персонализации, это: **неперсональные** рекомендации, когда система практически не анализируют пользователя, ограничиваясь только регионом и временем и рекомендует ему тоже, что и другим пользователям из того же региона, **сессионные** рекомендации, когда система анализирует вашу текущую сессию работы, например вы просматривали ноутбуки и в течении определенного времени вы будете видеть предложения о ноутбуках, и **персональные** рекомендации, когда система имеет полный доступ ко всей истории работы пользователя с системой и выдает предложения на основе анализа ее истории [1, 2].
- **Прозрачность системы**, или то, насколько система объясняет в результате чего была сделана та или иная рекомендация. Это важно в первую очередь для маркетинговых систем, потому что пользователи больше доверяют суждениям системы, если они могут убедиться, что на них не оказывает влияния та или иная компания, производитель товаров и услуг [1, 2].
- **Алгоритмы** расчета рекомендаций. Большинство существующих алгоритмов сводятся к нескольким основным подходам, которые будут описаны далее. К классическим алгоритмам относятся Content-based или модели, основанные на описании товаров, Summary-based неперсональные, Collaborative Filtering или коллаборативная фильтрация, Matrix Factorization или методы, которые основаны на матричном разложении, а также некоторые другие [1, 2].

В ядре любой рекомендательной системы лежит **матрица предпочтений**, которая представляет из себя матрицу, по одной из осей которой отложены все **пользователи системы**, а по другой **объекты рекомендации**. На пересечении некоторых из пар пользователь/объект матрица заполнена оценками, это **показатель заинтересованности** пользователя в объекте рекомендации. Пример подобной матрицы представлен ниже в таблице 1 [2].

Таблица 1 – Пример матрицы предпочтений

	Item 1	Item 2	Item 3	Item 4
User 1	8		8	5
User 2	6	2	10	
User 3		1	7	5
User 4	4	1	8	8

Пользователи обычно оценивают только небольшую часть объектов рекомендации, одна из основных задач рекомендательной системы заключается

в обобщении имеющейся информации и предсказании отношения пользователя к остальным объектам рекомендации, про которые ничего не известно. То есть системе необходимо заполнить пустые пары пересечений пользователь/объект матрицы предпочтений.

Шаблоны использования системы у разных людей разные, поэтому, например в маркетинге, имеет смысл показывать пользователю не только новые товары, которые он никогда раньше не покупал, но и товары, которые он покупает периодически, по этому принципу выделяются две группы объектов рекомендации: повторяемые (продукты питания, гигиены, расходники) и неповторяемые (книги, фильмы, игры). Кроме того, для некоторых пользователей имеет смысл предлагать товары только из их любимой категории, а другим иногда предлагать некоторое количество нестандартных товаров, по этому принципу рекомендации делятся на консервативные и рисковые [1].

Также из-за различных шаблонов использования системы у разных групп людей имеет смысл собирать оценки объектов рекомендации поразному. Для одной группы имеет смысл **явной** сборки оценки, где пользователь самостоятельно проставляет рейтинг объекту, оставляет отзыв, дает ссылку на тот или иной рекомендуемый объект системы, для другой же эффективнее собирать такую информацию **неявно** из косвенного анализа действий пользователя, например покупок или задержек внимания на той или иной странице с рекомендуемым объектом.

Неперсонализированные рекомендации, такие рекомендательные системы являются самыми простыми в реализации, потому что не требуют анализа информации о каждом конкретном пользователе. В таких системах показатель заинтересованности пользователя в объекте рекомендации определяется простым средним рейтингом рекомендуемого объекта. По такому принципу работают большинство рекомендательных систем, где от пользователя не требуется авторизация. Предоставляться рекомендации могут разными способами, например в виде баннера или ленты рядом с описанием объекта рекомендации, или как результат поиска по определенному критерию в виде отсортированного списка. Рейтинг объекта рекомендации может быть представлен также разными способами, например в виде звездочек рядом с объектом рекомендации, количеством лайков и репостов, отрицательных и положительных оценок, или в виде графика оценок.

Основными проблемами данных систем являются **актуальность рекомендации** и **холодный старт** [1]. Проблема свежести или актуальности объекта рекомендации особенно актуальна в маркетинге и на форумах. Она заключается в том, что необходимо чтобы пользователям чаще показывались свежие объекты, которые недавно появились в системе, и по которым еще не набрана детальная статистика в матрице предпочтений, как для более старых объектов. Для решения данной проблемы применяют различные формулы вычисления показателя заинтересованности пользователя с учетом переменной времени создания объекта рекомендации. Такие формулы обычно выво-

дятся персонально для каждой отдельной системы и проверяются опытном путем, примеры подобных формул представлены ниже.

$Rank = \frac{P*(U-D-1)^{0.8}}{T^{1.8}}$, где T – время создания объекта, U – положительные оценки, D – отрицательные оценки, P – параметр корректировки.

$Rank = log_{10}(max(1, U - D)) - \frac{|U-D|T}{const}$, где T – время создания объекта, U – положительные оценки, D – отрицательные оценки, *const* – константа [1].

Проблема холодного старта заключается в том, что на момент старта системы матрица предпочтений еще практически не заполнена, и рекомендации на основе такой матрицы будут недостоверными. Для решения данной проблемы используют два различных способа.

Первый способ заключается в показывании не среднего значения, а сглаженного среднего показателя заинтересованности пользователя (Damped Mean). Смысл подхода заключается в показывании безопасного среднего показателя при малом количестве оценок, а при наборе достаточной базы оценок усредняющая корректировка перестает действовать.

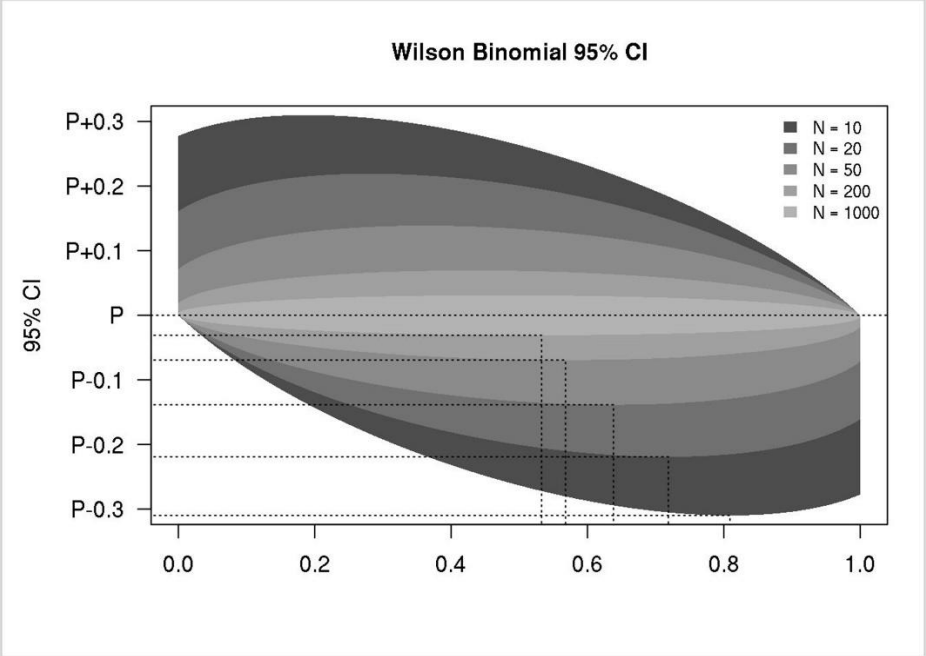


Рисунок 1 – Несимметричные интервалы

Второй способ заключается в расчете интервалов достоверности (confidence intervals) по каждому показателю его заинтересованности. Чем больше показателей в матрице предпочтений, тем меньше вариация среднего, а значит и больше вероятность его корректности. В качестве рейтинга объекта рекомендации можно выводить нижнюю границу интервала (Low CI Bound) [1, 3]. При таком подходе рекомендательная система будет консервативной, потому что она будет занижать оценки по новым объектам рекомендации. Так как показатели заинтересованности пользователя ограничены некоторой шкалой, например от 0 до 10, стандартный способ расчета интервала

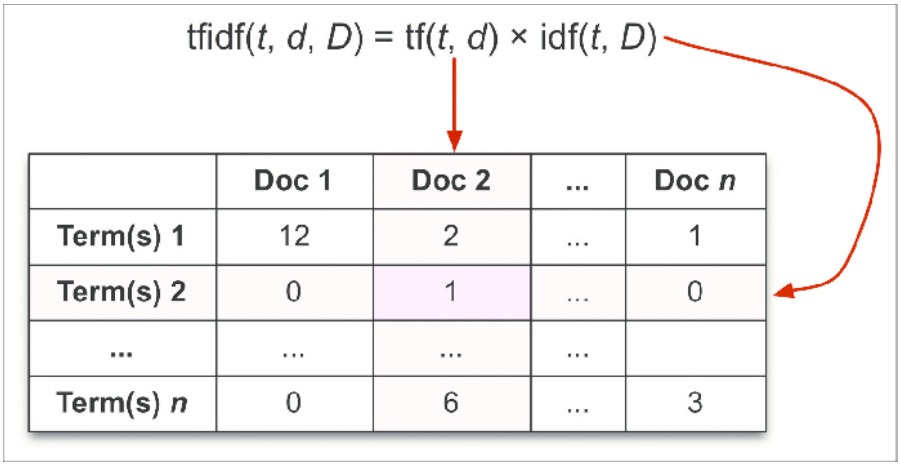
достоверности здесь практически не применим, из-за хвостов распределения, а также симметричности интервала. Но существует альтернативный способ для его расчета – Wilson Confidence Interval [3]. Вид несимметричных интервалов, которые получаются при этом, показаны на рис. 1. На рис. 1 по горизонтали отложена оценка среднего показателя заинтересованности пользователя, а по вертикали – разброс среднего значения.

Проблема холодного старта актуальна и для персонализированных рекомендаций. Общий способ решения проблемы здесь заключается в замене того, что на данный момент не может быть рассчитано, различными эвристиками.

Персонализированные рекомендации. Такие рекомендательные системы предполагают использование пользовательской информации, в первую очередь о его истории работы с системой. Content-based filtering – является одним из первых появившихся подходов к реализации такого типа систем. Суть данного подхода заключается в том, что описание товара (content) сравнивается с интересами конкретных пользователей, которые были получены из его предыдущих показателей заинтересованности. Чем выше соответствие объекта рекомендации интересам пользователя, тем выше рейтинг данного объекта в системе. Очевидным требованием для данного подхода является наличие описания у каждого объекта рекомендации системы.

Примерами объектов систем, основанных на подходе Content-based, являются товары с неструктурированным описанием, такие как различные статьи, книги, фильмы, игры. Признаками таких объектов могут быть, например, рецензии, текстовые описания, состав актеров и многое другое. Но можно использовать и обычные числовые или категориальные признаки. Неструктурированные признаки описываются в таких системах, как векторы в пространстве слов (Vector-Space model) [1]. Элементы такого вектора являются признаками, характеризующими интересы пользователей. По мере использования системы пользователем векторные описания его удачно рекомендованных объектов объединяются (суммируются и нормализуются) в единый вектор, формируя вектор его интересов. После чего достаточно найти объект рекомендации, у которого описание наиболее близко к вектору интересов пользователя (решить задачу поиска n ближайших соседей).

Однако, не все элементы описания одинаково значимы, например, союзные слова, не несут никакой полезной информации в себе. Для решения этой проблемы необходимо при определении числа элементов, которые совпали в двух векторах, все измерения необходимо взвесить по их значимости. Для решения этой задачи используется преобразование TF-IDF [4], назначающее больший вес более редким интересам. Из-за чего совпадение таких интересов имеет гораздо большее значение во время определения близости двух векторов, чем совпадение более популярных. На рис. 2 изображен расчет tfidf, где tf – мера значимости атрибута для пользователя, idf – мера редкости атрибута [1, 4].



	Doc 1	Doc 2	...	Doc *n*
Term(s) 1	12	2	...	1
Term(s) 2	0	1	...	0
...	...	...	...	
Term(s) *n*	0	6	...	3

Рисунок 2 – TF-IDF

Коллаборативная фильтрация, данный вид рекомендательных систем использует других похожих пользователей для генерации рекомендаций. Реализация таких систем основана на принципе k ближайших соседей. То есть для каждого пользователя алгоритм ищет k наиболее схожих с ним по предпочтениям пользователей и дополняет информацию об основном пользователе данными его соседей. На рис. 3 изображен принцип работы данного метода, желтым выделен основной пользователь, синим его ближайшие соседи [1, 5].

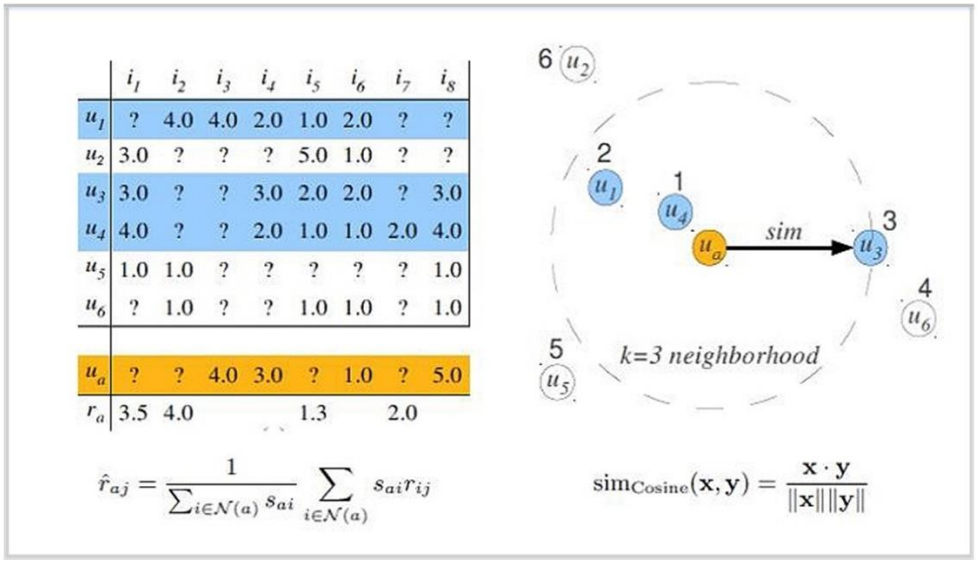


Рисунок 3 – Коллаборативная фильтрация

Похожесть в данном случае – это синоним корреляции интересов пользователей и может рассчитываться множеством способов, например корреляцией Пирсона, косинусным расстоянием, расстоянием Жаккара, расстоянием Хэмминга [1, 2, 5].

У классической реализации такого алгоритма есть одна проблема, он плохо применим из-за квадратичной сложности, метод требует расчета всех попарных расстояний между пользователями. Данная проблема может быть решена с помощью мощного железа или корректировок алгоритма, таких как:

- обновление расстояний производить не при каждом взаимодействии с системой, а порциями, например раз в час или раз в день;
- не пересчитывать матрицу расстояний в целом, а инкрементально;
- рассчитывать только приближенные значения.

Для использования данного подхода также необходимо сделать несколько допущений:

- вкусы людей не меняются со временем;
- если вкусы людей совпадают, то они совпадают абсолютно во всем.

Кроме того, для использования алгоритма крайне важно нормализовать значения показателей заинтересованности пользователей, поскольку они могут оценивать объекты рекомендации по-разному, например один активно проставляет максимальные оценки всем объектам рекомендации, а второй крайне редко использует оценки выше средней. Приведение таких оценок к единой шкале значительно повышает эффективность алгоритма.

Заключение. В данной статье были рассмотрены рекомендательные системы, а в частности их задачи, основные характеристики, алгоритмы, проблемы и способы их решения. В результате было показано, что все рекомендательные системы имеют свои проблемы, которые в основном связаны со сбором и интерпретацией пользовательских предпочтений, а также сложностями работы систем при недостатке таких оценок при холодном старте работы системы.

Список литературы:

1. Анатомия рекомендательных систем [Электронный ресурс]. – URL: https://habr.com/ru/company/lanit/blog/420499/, свободный (дата обращения: 10.03.2020).
2. Кутянин, А.Р. Рекомендательные системы: обзор основных постановок и результатов. 2017 [Электронный ресурс]. – URL: http://www.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=ista&paperid=26&option_lang=rus, свободный (дата обращения: 10.03.2020).
3. Википедия Binomial proportion confidence interval [Электронный ресурс]. – URL: https://en.wikipedia.org/wiki/Binomial_proportion_confidence_interval, свободный (дата обращения: 10.03.2020).
4. Википедия tf–idf [Электронный ресурс]. – URL: https://en.wikipedia.org/wiki/Tf%E2%80%93idf, свободный (дата обращения: 10.03.2020).
5. Коллаборативная фильтрация [Электронный ресурс]. – URL: https://habr.com/ru/post/150399/, свободный (дата обращения: 10.03.2020).