

Гальперина А.А.

студент

*4 курс, факультет «Дизайн и программная инженерия»
ФГБОУ ВО «Казанский национальный исследовательский
технологический университет»*

Россия, г. Казань

Махмутова И.И.

студент

*4 курс, факультет «Дизайн и программная инженерия»
ФГБОУ ВО «Казанский национальный исследовательский
технологический университет»*

Россия, г. Казань

*Руководитель: Тазиева Р.Ф., кандидат технических наук
доцент кафедры «Информатика и прикладная математика»*

*ФГБОУ ВО «Казанский национальный исследовательский
технологический университет»*

Россия, г. Казань

МЕТОДЫ ПОИСКА АССОЦИАТИВНЫХ ПРАВИЛ

Данная статья вкратце знакомит читателя с понятием ассоциативных правил. Вводит в историю их возникновения. Описывает и классифицирует методы поиска ассоциативных правил.

***Ключевые слова:** ассоциативные правила, методы поиска ассоциативных правил*

This article introduces the reader to the concept of associative rules. Enters into the history of their occurrence. Describes and classifies associative rule lookup methods.

***Key words:** associative rules, associative rule search methods.*

Для достижения высокой скорости развития информационных технологий требуется сохранять и преумножать объем данных. Ввиду большого объема информации, ее анализ стал весьма затруднителен для понимания. Эта проблема положила начало развитию методов автоматического исследования данных.

Значительно повысился уровень интереса к методам "обнаружения знаний в базах данных" (knowledge discovery in databases). Внушительные размеры бд оказывают влияние на алгоритмы анализа данных, требуя от них масштабируемости. Поэтому наибольшей популярностью в поиске знаний пользуются алгоритмы поиска ассоциативных правил.

Они используются для нахождения связей между событиями. Наличие ассоциативных отношений наблюдается между товарами, которые в большинстве сделок появляются вместе.

Ассоциативные правила поначалу применялись в рознице при поиске совместно приобретенных товаров. Но на этом область их использования не ограничилась. Правила широко применяются в разделении клиентов на группы путем изучения покупательских предпочтений, при помощи анализа, персональной рассылки, перекрестного маркетинга, в медицине - при изучении последствий пагубного воздействия наркотиков, при изучении переписи, при предупреждении отказов телекоммуникационных оборудования.

МЕТОДЫ ПОИСКА АССОЦИАТИВНЫХ ПРАВИЛ

Сегодня существует разнообразное количество методов поиска ассоциативных правил в разных источниках данных. Рассмотрим более подробно некоторые из них.

АЛГОРИТМ AIS

Создан исследователями научной организации IBM Almaden в 1993 году. Это самая первая разработка такого рода.

Здесь кандидаты наборов создаются и рассчитываются при сканировании базы данных. Каждая транзакция исследуется на содержание крупных наборов, найденных во время прошлого сканирования базы данных. Таким образом, формирование новых наборов происходит благодаря увеличению имеющихся

наборов. AIS признан одним из самых малоэффективных алгоритмов. Это случилось потому, что AIS учитывает достаточно небольшие и редко встречающиеся наборы кандидатов.

АЛГОРИТМ SETM

Его написанию способствовала заинтересованность специалистов в использовании технологий языка SQL для выявления повторяющихся групп товаров. Подобно ранее рассмотренному AIS, SETM "на лету" образует кандидатов, базируясь на перестройке базы данных. Для применения операции объединения - создания кандидата, данный алгоритм разграничивает подсчет кандидатов и их группировки.

Существенным недостатком вышеперечисленных алгоритмов является подсчет редко встречающихся кандидатов. Чтобы повысить качество работы рассмотренных выше алгоритмов, исследователи разработали новый алгоритм - Apriori, чья работа основана на двух этапах: формирования списка кандидатов и их подсчет.

АЛГОРИТМ APRIORY

Его написанием В 1994 г. занимались компании Rakesh Agrawal и Ramakrishnan Srikant, группа исследователей Almaden IBM. На сегодняшний день этот алгоритм пользуется наибольшей востребованностью среди других алгоритмов.

Алгоритм Apriori занимается поиском ассоциативных правил и применяется в бд, состоящим из количества транзакций.

Перед началом работы алгоритма нужно определить 3 параметра: 1. размер набора (состоит из 2 или большего количества элементов), 2. поддержку (транзакции, входящие в набор, разделенное на их общее число) и 3. достоверность - условную вероятность определенного товара оказаться в одной корзине с другими товарами.

Простой Apriori представлен объединением (частотой вхождения отдельных товаров), отсечением (переход удовлетворяющих уровню поддержки

и достоверности наборов на следующую итерацию с двухкомпонентными наборами) и повторения.

Алгоритм делает несколько проходов по базе данных. При первом прохождении учитывается поддержка отдельных элементов. Элементы, имеющие поддержку, уровень которой превосходит или равняется уровню минимальной поддержки, рассматриваются как крупные предметы. Любой дальнейший проход k после этого, большие наборы элементов в предыдущем проходе сгруппированы в наборы из k элементов, это и есть набором кандидатов. Расчет поддержки для различных наборов кандидатов подсчитывается, и, если установлено, что поддержка превышает минимальный уровень, то такая группа элементов считается большой. Этот процесс продолжается до тех пор, пока большой набор элементов в конкретном проходе не станет пустым.

В этом алгоритме число проходов по бд напрямую зависит от размера самого длинного часто встречающегося набора.

Рассмотрим несколько подвидов алгоритма Apriori, которые являются его улучшенной версией.

AprioriTid

Здесь бд не применяется для подсчета поддержки после первого прохода. Группа подходящих наборов элементов используется для этой цели при $k > 1$. Если транзакция не имеет какого-либо набора k -элементов-кандидатов, тогда группа наборов-кандидатов не будет иметь какой-либо записи для этой транзакции, что в конечном итоге уменьшит число транзакций в наборе, состоящий из наборов-кандидатов, по сравнению с базой данных. При увеличении значения k каждая запись будет меньше, чем соответствующие транзакции, поскольку число кандидатов в транзакциях будет уменьшаться. Apriori работает лучше, чем AprioriTid на начальных этапах, но на более поздних этапах AprioriTid имеет лучшую производительность, чем Apriori.

AprioriHybrid

Основанный операции генерации наборов кандидатов, алгоритм AprioriHybrid используется для объединения лучших качеств Apriori и

AprioriTid. Он повторяет алгоритм Apriori на начальных этапах реализации, в дальнейшем переходя к алгоритму AprioriTid. Смена алгоритма происходит в то время, когда начинается проверка соответствия закодированного набора источника, установленного в самом конце фрагмента, возможностям памяти. Это происходит потому, что переключение с одного алгоритма на другой является непростым процессом и требует подключения дополнительных ресурсов.

ДРУГИЕ АЛГОРИМЫ ПОИСКА

DHP

Алгоритм хеширования разработанный и спроектированный в 1995 г. (J. Park, M. Chen and P. Yu). Он основывается на стохастическом подсчете наборов-кандидатов, предназначенном для уменьшения числа подсчитываемых кандидатов на каждом шаге алгоритма Apriori. Снижение происходит из-за того, что каждый из k -элементных наборов-кандидатов помимо шага сокращения проходит шаг хеширования. На $k-1$ этапе при выборе кандидата организовывается хеш-таблица. Ее записями являются счетчики всех поддержек k -элементных наборов, соответствующих этой записи в хеш-таблице. DHP применяет эту информацию на k -этапе для сокращения множества k -элементных наборов-кандидатов. После сокращения подмножества, аналогично Apriori, алгоритм может убрать набор-кандидат, если его значение в хеш-таблице меньше порогового значения, установленного для обеспечения.

PARTITION

Алгоритм разбиения (разделения). Основной его задачей является сканирование транзакционной базы данных, которое осуществляется путем ее разбиения на непересекающиеся разделы, каждый из которых может поместиться в оперативной памяти. Начальный этап заключается в определении часто встречающихся наборов данных при помощи алгоритма Apriori. Второй этап подсчитывает поддержку каждого такого набора относительно всей базы данных. То есть, второй шаг заключается в определении множества всех потенциально встречающихся наборов данных.

ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ

1. Алгоритм Apriori. [Электронный ресурс]. URL: <http://datascientist.one>. (дата обращения: 23.12.2019).
2. APRIORI, APRIORITID and APRIORI HYBRID. [Электронный ресурс]. URL: <http://associationrule.blogspot.com> (дата обращения: 21.12.2019).
3. Методы поиска ассоциативных правил. [Электронный ресурс]. URL: <https://www.intuit.ru> (дата обращения: 24.12.2019).
4. Методы поиска ассоциативных правил. [Электронный ресурс]. URL: <https://studbooks.net> (дата обращения: 21.12.2019).