Кластеризация категорийных данных: масштабируемый алгоритм CLOPE

Data Mining, Big Data | **24 комментария** | Версия для печати

Введение и основные идеи

Задачи кластеризации больших массивов категорийных данных весьма актуальна для систем анализа данных. Категорийные данные встречаются в любых областях: производство, коммерция, маркетинг, медицина… Категорийные данные включают в себя и так называемые транзакционные данные: чеки в супермаркетах, логи посещений веб-ресурсов. Сюда же относится анализ и классификация текстовых документов (Text Mining).

Здесь и далее под категорийными данными понимаются качественные характеристики объектов, измеренные в шкале наименований. Напомним: при использовании шкалы наименований указывается только, одинаковы или нет объекты относительно измеряемого признака.

Применять для кластеризации объектов с категорийными признаками традиционные алгоритмы неэффективно, а часто – невозможно (подробнее см. материал "Алгоритмы кластеризации на службе Data Mining"). Основные трудности связаны с высокой размерностью и гигантским объемом, которыми часто характеризуются такие базы данных.

Алгоритмы, основанные на парном вычислении расстояний (k-means и аналоги) эффективны в основном на числовых данных. Их производительность на массивах записей с большим количеством нечисловых факторов неудовлетворительная. И дело даже не столько в сложности задания метрики для вычисления расстояния между категорийными атрибутами, сколько в том, что на каждой итерации алгоритма требуется попарно сравнивать объекты между собой, а итераций может быть очень много. Для таблиц с миллионами записей и тысячами полей это неприменимо.

Поэтому в последнее десятилетие ведутся активные исследования в области разработки масштабируемых (scalable) алгоритмов кластеризации категорийных и транзакционных данных. К ним предъявляются особые требования, а именно:

- минимально возможное количество "сканирований" таблицы базы данных;
- работа в ограниченном объеме оперативной памяти компьютера;
- работу алгоритма можно прервать с сохранением промежуточных результатов, чтобы продолжить вычисления позже;
- алгоритм должен работать, когда объекты из базы данных могут извлекаться только в режиме однонаправленного курсора (т.е. в режиме навигации по записям).

На сегодняшний день предложено свыше десятка методов для работы с категорийными данными, например, семейство иерархических кластерных алгоритмов. Но не всегда они удовлетворяют перечисленным выше требованиям. Одним из эффективных считается алгоритм LargeItem, который основан на оптимизации некоторого глобального критерия. Этот глобальный критерий использует параметр поддержки (в терминологии здесь много общего с алгоритмами для выявления ассоциативных правил). Вообще, вычисление глобального критерия делает алгоритм кластеризации во много раз быстрее, чем при использовании локального критерия при парном сравнении объектов, поэтому "глобализация" оценочной функции – один из путей получения масштабируемых алгоритмов.

Алгоритм CLOPE, который мы рассматриваем в данной статье, очень похож на LargeItem, но быстрее и проще в программной реализации. CLOPE предложен в 2002 году группой китайских

Для начала формализуем рассматриваемую задачу кластеризации для категорийных данных. Все изложение будет идти как будто бы у нас в наличии имеется база транзакционных данных, а в конце материала будет показано, как с помощью CLOPE разбивать на кластеры любые категорийные массивы, работая с ними как с транзакционными.

Под термином *транзакция* здесь понимается некоторый произвольный набор объектов, будь это список ключевых слов статьи, товары, купленные в супермаркете, множество симптомов пациента, характерные фрагменты изображения и так далее. Задача кластеризации транзакционных данных состоит в получении такого разбиения всего множества транзакций, чтобы похожие транзакции оказались в одном кластере, а отличающиеся друг от друга – в разных кластерах.

В основе алгоритма кластеризации CLOPE лежит идея максимизации глобальной функции стоимости, которая повышает близость транзакций в кластерах при помощи увеличения параметра *кластерной гистограммы*. Рассмотрим простой пример из 5 транзакций: {(a,b), (a,b,c), (a,c,d), (d,e), (d,e,f)}. Представим себе, что мы хотим сравнить между собой следующие два разбиения на кластеры:

(1) {{ab, abc, acd}, {de, def}}

(2) {{ab, abc}, {acd, de, def}}.

Для первого и второго вариантов разбиения в каждом кластере рассчитаем количество вхождений в него каждого элемента транзакции, а затем вычислим высоту (H) и ширину (W) кластера. Например, кластер {ab, abc, acd} имеет вхождения a:3, b:2, c:2 с H=2 и W=4. Для облегчения понимания на рис. 1 эти результаты показаны геометрически в виде гистограмм.

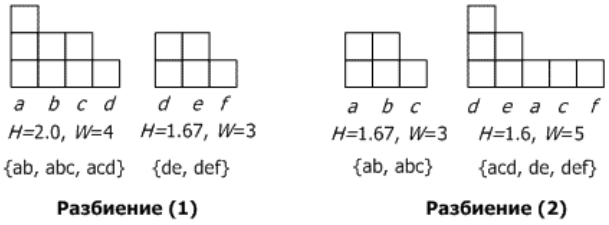

Рисунок 1. Гистограммы двух разбиений

Качество двух разбиений оценим, проанализировав их высоту H и ширину W. Кластеры {de, def} и {ab, abc} имеют одинаковые гистограммы, следовательно, равноценны. Гистограмма для кластера {ab, abc, acd} содержит 4 различных элемента и имеет площадь 8 блоков (H=2.0, H/W=0.5), а кластер {acd, de, def} – 5 различных элементов с такой же площадью (H=1.6, H/W=0.32). Очевидно, что разбиение (1) лучше, поскольку обеспечивает большее наложение транзакций друг на друга (соответственно, параметр H там выше).

На основе такой очевидной и простой идеи геометрических гистограмм и работает алгоритм CLOPE (англ.: Clustering with sLOPE). Рассмотрим его подробнее в более формальном описании.

Алгоритм CLOPE

Пусть имеется база транзакций D, состоящая из множества транзакций $\{t_1,t_2,\ldots,t_n\}$. Каждая транзакция есть набор объектов $\{i_1,\ldots,i_m\}$. Множество кластеров $\{C_1,\ldots,C_k\}$ есть разбиение множества $\{t_1,\ldots,t_n\}$, такое, что $C_1 \ldots C_k=\{t_1,\ldots,t_n\}$ и $C_i \neq \varnothing \wedge C_i \bigcap C_j = \varnothing$, для 1<=i, j<=k. Каждый элемент C_i называется *кластером*, n, m, k – количество транзакций, количество объектов в базе транзакций и число кластеров соответственно.

Каждый кластер C имеет следующие характеристики:

D(C) – множество уникальных объектов;

$S(C) = \sum_{i \in D(C)} Occ(i,C) = \sum_{t_i \in C} |t_i|$

W(C) = |D(C)|;

H(C) = S(C)/W(C).

Гистограммой кластера C называется графическое изображение его расчетных характеристик: по оси OX откладываются объекты кластера в порядке убывания величины Occ(i,C), а сама величина Occ(i,C) – по оси OY (рис. 2).

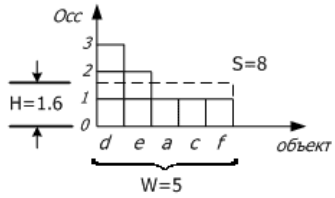

Рисунок 2. Иллюстрация гистограммы кластера

На рис. 2 S(C), равное 8, соответствует площади прямоугольника, ограниченного осями координат и пунктирной линией. Очевидно, что чем больше значение H, тем более "похожи" две транзакции. Поэтому алгоритм должен выбирать такие разбиения, которые максимизируют H.

Однако учитывать одно только значение высоты H недостаточно. Возьмем базу, состоящую из 2х транзакций: {abc, def}. Они не содержат общих объектов, но разбиение {{abc, def}} и разбиение {{abc}, {def}} характеризуются одинаковой высотой H=1. Получается, оба варианта разбиения равноценны. Но если для оценки вместо H(C) использовать градиент $G(C)=H(C)/W(C)=S(C)/W(C)^2$, то разбиение {{abc},{def}} будет лучше (градиент каждого кластера равен 1/3 против 1/6 у разбиения {{abc, def}}).

Обобщив вышесказанное, запишем формулу для вычисления глобального критерия – функции стоимости Profit(C):

$$Profit(C) = \frac{\sum_{i=1}^{k} G(C_i) \times |C_i|}{\sum_{i=1}^{k} |C_i|} = \frac{\sum_{i=1}^{k} \frac{S(C_i)}{W(C_i)^r} \times |C_i|}{\sum_{i=1}^{k} |C_i|}$$

$|C_i|$ – количество транзакций в i-том кластере, k – количество кластеров, r – положительное вещественное число большее 1.

С помощью параметра r, названного авторами CLOPE *коэффициентом отталкивания* (repulsion), регулируется уровень сходства транзакций внутри кластера, и, как следствие, финальное количество кластеров. Этот коэффициент подбирается пользователем. Чем больше r, тем ниже уровень сходства и тем больше кластеров будет сгенерировано.

Формальная постановка задачи кластеризации алгоритмом CLOPE выглядит следующим образом: для заданных D и r найти разбиение C: Profit(C,r) -> max.

Реализация алгоритма

Предположим, что транзакции хранятся в таблице базы данных. Лучшее решение ищется в течение последовательного итеративного перебора записей базы данных. Поскольку критерий оптимизации имеет глобальный характер, основанный только на расчете H и W, производительность и скорость алгоритма будет значительно выше, чем при попарном сравнении транзакций.

Реализация алгоритма требует первого прохода по таблице транзакций для построения начального разбиения, определяемого функцией Profit(C,r). После этого требуется незначительное (1-3) количество дополнительных сканирований таблицы для повышения качества кластеризации и оптимизации функции стоимости. Если в текущем проходе по таблице

1. // Фаза 1 – инициализация
2. Пока не конец
3. прочитать из таблицы следующую транзакцию [t, -];
4. положить t в существующий либо в новый кластер C_i, который дает максимум Profit(C,r);
5. записать [t,i] в таблицу (номер кластера);
6. // Фаза 2 – Итерация
7. Повторять
8. перейти в начало таблицы;
9. moved := false;
10. пока не конец таблицы
11. читать [t,i];
12. положить t в существующий либо в новый кластер C_j, который максимизирует Profit(C,r);
13. если C_i<>C_j тогда
14. записать [t,i];
15. moved := true;
16. пока (not moved).
17. удалить все пустые кластеры;

Как видно, алгоритм CLOPE является масштабируемым, поскольку способен работать в ограниченном объеме оперативной памяти компьютера. Во время работы в RAM хранится только текущая транзакция и небольшое количество информации по каждому кластеру, которая состоит из: количества транзакций N, числа уникальных объектов (или ширины кластера) W, простой хэш-таблицы для расчета Occ(i,C) и значения S площади кластера. Они называются *кластерными характеристиками* (CF – cluster features). Для простоты обозначим их как свойства кластера C, например, C.Occ[i] означает число вхождений объекта i в кластер C и т.д. Можно посчитать, что для хранения частоты вхождений 10 тыс. объектов в 1 тыс. кластерах необходимо около 40 Мб оперативной памяти.

Для завершения реализации алгоритма нам нужны еще две функции, рассчитывающие прирост Profit(C,r) при добавлении и удалении транзакции из кластера. Это легко сделать, зная величины S,W и N каждого кластера:

1. function DeltaAdd(C,t,r): double;
2. begin
3. S_new := C.S + t.ItemCount;
4. W_new := C.W;
5. for i:=0 to t.ItemCount–1 do
6. if (C.Occ[t.items[i]]=0) then W_new := W_new + 1;
7. result := $S\_new*(C.N+1)/(W\_new)^r – C.S*C.N/(C.W)^r$
8. end;

Николай Паклин

добавляемой транзакции t.

Реализация функции прироста Profit(C,r) при удалении транзакции похожа на DeltaAdd(C,t,r), поэтому опустим ее подробный код.

Следующая теорема гарантирует корректность использования функции DeltaAdd.

> **Теорема.** Если DeltaAdd(C_i,t) есть максимум, то перемещение t в кластер C_i максимизирует Profit(C,r).

Теперь можно оценить вычислительную сложность алгоритма CLOPE. Пусть средняя длина транзакции равна A, общее число транзакций N, максимально возможное число кластеров K. Временная сложность одной итерации равна O(N*K*A), показывающая, что скорость работы алгоритма растет линейно с ростом кластеров и размера таблицы. Это делает алгоритм быстрым и эффективным на больших объемах.

Рассказав о реализации алгоритма, мы ничего не сказали о виде таблицы транзакций, чтобы можно было применять алгоритм CLOPE. CLOPE позволяет решать задачи кластеризации не только транзакционных данных, но и любых категорийных. Главное, чтобы все признаки объектов были измерены в шкале наименований. Однако перед тем как запускать CLOPE, данные необходимо привести к нормализованному виду. Он может иметь вид бинарной матрицы образов, как в ассоциативных правилах, так и представлять собой взаимно однозначное отображение между множеством уникальных объектов {u_1,…u_q} таблицы и множеством целых чисел {0,1,2,…,q-1}.

Задача о грибах

Задача о грибах (The mashroom dataset) – популярный тест, который применяют для оценки алгоритмов кластеризации категорийных наборов данных (доступен на UCI machine learning repository). Тестовая выборка содержит 8124 записи с описанием 22 характеристик грибов двух классов: 4208 съедобных (e) и 3916 несъедобных (p) грибов. Файл выборки имеет следующий вид:

```
p,x,s,n,t,p,f,c,n,k,e,e,s,s,w,w,p,w,o,p,k,s,u
e,x,s,y,t,a,f,c,b,k,e,c,s,s,w,w,p,w,o,p,n,n,g
e,b,s,w,t,l,f,c,b,n,e,c,s,s,w,w,p,w,o,p,n,n,m
p,x,y,w,t,p,f,c,n,n,e,e,s,s,w,w,p,w,o,p,k,s,u
e,x,s,g,f,n,f,w,b,k,t,e,s,s,w,w,p,w,o,e,n,a,g
..., ..., ...
```

Общее количество уникальных характеристик объектов равно 116. 2480 записей имеют пропущенные значения в одном атрибуте.

Если такой набор данных представить в описанном выше нормализованном виде, то получится 8124 транзакции, из которых 2408 будут длиной 21, а остальные – 22 элемента (пропущенные значения игнорируются). И теперь можно применить алгоритм CLOPE. Результат работы CLOPE при r=2.6 для задачи о грибах после 1-ой итерации (фаза инициализации) представлен на рис. 3. При этом критерием качества работы алгоритма служит количество "грязных" кластеров, т.е. таких, в которых присутствуют как съедобные (e), так и несъедобные (p) грибы. Чем меньше таких кластеров, тем лучше. Из кросс-таблицы на рис. 3 видно, что уже после 1-ой итерации остался только 1 "грязный" кластер №18. Потребуется еще пару-тройку сканирований базы данных для получения финальной кластеризации. Очевидно, что кластер 12 исчезнет.

Детальное исследование работы алгоритма CLOPE, проведенное его авторами, показало высокое качество кластеризации в сравнении с другими алгоритмами, в т.ч. иерархическими. При этом по скорости работы и производительности он обгоняет их в несколько раз.

регрессии в медицине и скоринге

Логистическая регрессия и ROC-анализ — математический аппарат

Алгоритмы кластеризации на службе Data Mining

Непрерывные генетические алгоритмы — математический аппарат

Нечеткая логика — математические основы

Подписка
на материалы сайта

Ваш email...

Подписаться

3	768	
4	96	
5	96	
6	192	
7	1 296	
8	432	
9		149
10		192
11		1 146
12		1
13		288
14	192	
15		223
16	48	
17		72
18	48	32
19		8
20		8
21		1 497
22	192	
23	288	
24	32	
25		36
26		8
27	16	
Итого	4 208	3 916

Рисунок 3. Результат работы CLOPE после 1 итерации

Области применения CLOPE

Алгоритм CLOPE предназначен для работы с транзакционными данными, но, как мы увидели, очень много наборов данных с категорийными атрибутами представляют собой транзакционные данные либо сводятся к ним. Ответы респондента в анкете, список ключевых слов документа, множество посещенных веб-ресурсов пользователя, симптомы больного, характеристики гриба – все это не что иное, как транзакция. Поэтому области применения CLOPE распространяются на все массивы категорийных баз данных.

Вообще, кластеризация транзакционных данных имеет много общего с анализом ассоциаций. Обе эти технологии Data Mining выявляют скрытые зависимости в наборах данных. Но есть и отличия. С одной стороны, кластеризация дает общий взгляд на совокупность данных, тогда как ассоциативный анализ находит конкретные зависимости между атрибутами. С другой стороны, ассоциативные правила сразу пригодны для использования, тогда как кластеризация чаще всего используется как первая стадия анализа.

В завершение подчеркнем преимущества алгоритма CLOPE:

1. Высокие масштабируемость и скорость работы, а так же качество кластеризации, что достигается использованием глобального критерия оптимизации на основе максимизации градиента высоты гистограммы кластера. Он легко рассчитывается и интерпретируется. Во время работы алгоритм хранит в RAM небольшое количество информации по каждому кластеру и требует минимальное число сканирований набора данных. Это позволяет применять его для кластеризации огромных объемов категорийных данных (large categorical data sets);
2. CLOPE автоматически подбирает количество кластеров, причем это регулируется одним единственным параметром – коэффициентом отталкивания.

ЛИТЕРАТУРА

1. Yang, Y., Guan, H., You. J. CLOPE: A fast and Effective Clustering Algorithm for Transactional Data In Proc. of SIGKDD'02, July 23-26, 2002, Edmonton, Alberta, Canada.

Масштабируемые алгоритмы | Кластеризация

Вам может быть интересно:

Быстродействие Deductor: файлы данных (до 10 млн. строк)

Быстродействие Deductor: файлы данных (до 1 млн. строк)

Быстродействие Deductor: файлы данных (до 100 тыс. строк)

Алгоритмы кластеризации на службе Data Mining

Комментарии

Войдите или зарегистрируйтесь для добавления комментариев.

Солеваров 27 июля 2006 11:24 Ссылка

Интересное переложение колмогоровской математики для K-струй.

Иван FXS 28 июля 2006 07:35 Ссылка

Вопрос к общей формуле Profit(C). Понятно, что формула «сконструирована», но не «с потолка», а – исходя из некоторой «интерпретации». Вопрос именно по интерпретации.

Я вижу, что «вес» (суммарный объем) каждого кластера учтен в формуле дважды: во-первых, добавление нового элемента в кластер увеличивает площадь (S) его гистограммы, и – следовательно – высоту (H). Но - дополнительно к этому – в формуле производится ПРЯМОЕ взвешивание вклада каждого кластера (его характеристики G) в общую сумму Profit(C), на «количество объектов в i-том кластере» C(i) …

Вопрос: зачем (в плане интерпретации алгоритма) нужно повторное - «грубое», посредством взвешивания слагаемых множителем C(i), - увеличение веса (вклада) крупных кластеров, если их характеристика G и без того будет больше, чем у мелких кластеров?

Николай Паклин 28 июля 2006 10:45 Ссылка

Если Вы внимательно посмотрите на формулы, то обнаружите, что добавление нового элемента в кластер не обязательно приводит к увеличению высоты H.

Иван FXS 28 июля 2006 11:40 Ссылка

Да, поскольку может увеличиться ширина гистограммы (W). Но "штраф" за увеличение ширины и так вводится НЕПОСРЕДСТВЕННО - увеличением степени, в которую (в знаменателе) возводится W.

Есть "основная", - явно итерпретируемая! - "игра" на балансе между увеличением "массы" кластера (читай - площади S, читай - высоты H) и расширением его гистограммы …

Зачем-то ПОВЕРХ нее вводится еще одна "игра" на числе элементов в кластере … Я понимаю, так тоже можно, можно вообще конструировать любые фоормулы … Вопрос - в интерпретации их!

Николай Паклин 28 июля 2006 11:58 Ссылка

Алгоритм CLOPE больше эвристический. Если Вам так интересно, то можете поразбираться в первоисточнике. Если нужен более интерпретируемый алгоритм, то рекомендую LargeItem. Но учтите, что он гораздо сложнее в реализации, а эффективность не лучше.

Иван FXS 28 июля 2006 12:11 Ссылка

На самом деле, алгоритм мне очень понравился: он (или какая-то его модификация) явно позволит мне продвинуться в задаче, которая стоит (застопорилась!) передо мной уже около года как. Это – задача кластеризации графов (конкретнее – Ассоциативных Семантических Сетей, см. об этом http://forum.aicommunity.org/viewtopic.php?p=19069).

Поэтому я пытаюсь «вжиться» в описаный Вами алгоритм, прочувствовать его.

Вот еще один мысленый эксперимент. Предположим, мы расссматриваем очередной элемент, чтобы отнести его к одному из кластеров, и оказывается, что он целиком (без необходимости расширения гистограмм) вписывается в несколько (два) кластеров …

Увеличение площади (S) он даст одно и то же, куда бы мы его ни положили … Увеличение высоты (H), и тем более – градиента (G), будет больше если мы положим его в кластер с меньшим основанием (W), иначе говоря – в более компактный …

Но! Из-за множителя C(i) нам может оказаться более выгодно положить его не в более компактный, а в более «массивный» кластер … хорошо ли это?

Николай Паклин 28 июля 2006 12:21 Ссылка

Точно ответить не могу, нужно на примере разобрать ситуацию, но с ходу думается, что при "втором" прогоне этот "массивный" кластер проглотит тот "компактный".

Иван FXS 28 июля 2006 12:52 Ссылка

Нет, у них могут быть мало-пересекающиеся "основания" ... а если и проглотит, то только в силу того самого "грубого" притяжения массивных кластеров, т.е. - множителя C!

Cyberian 18 января 2007 05:17 Ссылка

В формуле Profit(C), скорее всего очепятка. Судя по коду функции DeltaAdd, |C(i)| это не количество объектов i-том кластере, а количество транзакций в нем.

Alex 4 августа 2006 12:09 Ссылка

Народ, я извиняюсь, но все же .... А есть реализация этой штуковины на VB/VBA хоть какая-то? Или на другом языке каком нибудь? Я вот в программировании не силен, но имя хоть какую-то реализацию попросил бы переделать ее для моих нужд в Excel.

AgentGES 19 марта 2007 23:24 Ссылка

Есть реализация алгоритма CLOPE на Delphi.
Кто заинтересован - пишите agentges@bk.ru

Agentges 11 декабря 2010 00:10 Ссылка

Есть реализация алгоритма CLOPE на Delphi.
Кто заинтересован - пишите agentges@bk.ru

Пытаюсь применить данный алгоритм в одной задачке.

Еще раз перечитал статью, и поймал себя на мысли, что не понимаю формулировки: "|Ci| – количество объектов в i-том кластере".

Для ПОЛНОГО количества объектов в кластере (гистограмме кластера) уже введено обозначение S(Ci) ...

Для количества УНИКАЛЬНЫХ объектов в кластере (гистограмме кластера) уже введено обозначение W(Ci) ...

Что же такое |Ci|? Если оно равно одной из величин W(Ci) или S(Ci), тогда - вопрос - какой из них? И второй вопрос: зачем удвоены сущности (обозначения)?

Если не равно ни одной, то - что это и как его считать??

Николай Паклин 5 ноября 2007 14:25 Ссылка

|Ci| - количество транзакций в кластере. Возьмем пример: 4 транзакции и 1 разбиение (т.е. кластер единственный) {{abc, abcd, bcde, cde}}. В нем S=14, W=5, |Ci|=4, сумма |Ci|=4. И Profit=2.8 (при r=1).

Иван FXS 13 ноября 2007 00:05 Ссылка

О, спасибо! В самом деле, это очень разумная интерпретация ...

Дима 3 марта 2008 00:10 Ссылка

Будьте добры напишите реализацию этого алгоритма, а то уже целую неделю мучаюсь.

Александр Смирнов 8 апреля 2008 15:08 Ссылка

Реализовал CLOPE на java в качестве дополнения к Weka.
Могу выслать если интересно.

Елена 28 ноября 2010 17:04 Ссылка

Мне очень интересно!!!
на малых выборках работал нормально, а на выборке покрупнее уже несколько дней пытаюсь подобрать коэффициент, а он все складывает в один кластер.
было бы очень интересно посмотреть ваш вариант.

idag 3 апреля 2011 21:27 Ссылка

Реализация по примере weka дает в задаче о грибах 9 кластеров. Я уже перепробовал все условия, наилучший результата я добился это 16 кластеров. Кто знает конкретно что нужно сравнивать и как?

Иван FXS 21 ноября 2007 18:01 Ссылка

Продолжаю разбираться с CLOPE, и вот что меня сейчас смущает: похоже, что если транзакция хотя бы краешком "цепляет" кластер, то системе выгоднее, чтобы эта транзакция находилась в этом кластере, чем в "свободном состоянии". Это верно при любом значении параметра R.

Это, на самом деле, довольно неприятное свойство ...

Александр 29 ноября 2007 10:38 Ссылка

После знакомства с алгоритмом CLOPE, пришел к выводу, что у него нет недостатков:
- минимум обращений к базе данных
- не высокая вычислительная сложность

Заметил только проблему с оптимальным подбором коэффициента отталкивания (repulsion), но с этим вроде отлично справляется Fuzzy Clope.
Хотелось бы услышать мнения о возможных недостатках данного алгоритма, границах использования?

Тимур 14 января 2011 21:47 Ссылка

Здравствуйте
Написал программу и работает она на первый взгляд верно. Но если, например, подать на вход те же транзакции, но в другом порядке, результаты немного отличаются.
Вопрос: данный алгоритм зависит от последовательности обработки транзакций или нет? Просто хочу знать правильно ли я составил программу.

Руслан Русланов 16 сентября 2016 18:40 Ссылка

Относительно уникальных характеристик.
«Общее количество уникальных характеристик объектов равно 116».
Пересчитал три раза. Получил 126.

class {e,p} //класс транзакции e,p
cap-shape b,c,x,f,k,s 6
cap-surface f,g,y,s 4
cap-color n,b,c,g,r,p,u,e,w,y 10
bruises t,f 2
odor a,l,c,y,f,m,n,p,s 9
gill-attachment a,d,f,n 4
gill-spacing c,w,d 3
gill-size b,n 2
gill-color k,n,b,h,g,r,o,p,u,e,w,y 12
stalk-shape e,t 2
stalk-root b,c,u,e,z,r,? 7
stalk-surface-above-ring f,y,k,s 4
stalk-surface-below-ring f,y,k,s 4
stalk-color-above-ring n,b,c,g,o,p,e,w,y 9
stalk-color-below-ring n,b,c,g,o,p,e,w,y 9
veil-type p,u 2
veil-color n,o,w,y 4
ring-number n,o,t 3
ring-type c,e,f,l,n,p,s,z 8
spore-print-color k,n,b,h,r,o,u,w,y 9
population a,c,n,s,v,y 6
habitat g,l,m,p,u,w,d 7

Итого: 126

Руслан Русланов 28 сентября 2016 11:19 Ссылка

Требуется совет от профессионалов по кластеризации, на примере задачи о грибах. Столкнулся с вопросом нормализации таблицы:
p,x,s,n,t,p,f,c,n,k,e,e,s,s,w,w,p,w,o,p,k,s,u
e,x,s,y,t,a,f,c,b,k,e,c,s,s,w,w,p,w,o,p,n,n,g
e,b,s,w,t,l,f,c,b,n,e,c,s,s,w,w,p,w,o,p,n,n,m
…

А точнее с первой нормальной формой, которая говорит об атомарности.
На первом шаге есть два варианта:
«mushroom class» «property_1» «property_2» … «property_21» «property_22»
и
«mushroom class» «property_num» «property_value»

Но сначала рассмотрим пример на машинах в гараже. Есть список машин, которые находятся в гараже в таблице с колонками: «garage», «car1», «car2», «car3». Такой вид таблицы нарушает 1NF, т.к. колонки «car1», «car2», «car3» нарушают атомарность. Это может привести к той аномалии, когда в гараж добавляется четвертая машина, и тогда придется переделывать все, и БД и прикладную программу. Поэтому, необходимо делать так: «ragare», «carnumber», «car». Здесь колонка «carnumber» указывает номер машины 1, 2, или 3. И, если появляется четвертая машина, то нет необходимости добавлять новую колонку «car4», а всего лишь добавляется одна запись в таблицу, без изменения ее архитектуры. Это избавляет от указанной аномалии и придает атомарность.
Аналогично с грибами.

разные атрибуты, в отличие от приведенного примера с гаражом.
Поэтому имеет смысл с грибами оставить первый вариант:
«mushroom class» «property_1» «property_2» … «property_21» «property_22»

Но при этом с другой стороны, характеристики грибов закодированы односимвольными значениями. И для универсальности работы прибегнуть к варианту номер два:
«mushroom class» «property_num» «property_value»
Что исключает аномалии добавления новой характеристики и добавляет удобство для построения прикладных программ. Есть же разница писать 22 строки с обращением к каждому свойству, или конструкцию типа: for each in «mushroom table».

С третьей стороны биологи не каждый день добавляют новые характеристики в классификацию грибов, и формат поставляемой базы грибов является устоявшимся.

И опять же, приложение должно быть универсальным. Некий биолог разработал свою таблицу характеристик грибов. Которая похожа на указанную. Только в ней, допустим, не 22 характеристики, а 20 или 24.

Какой бы вариант выбрали вы и почем? Спасибо за ответы.

P.S. Я лично склоняюсь к варианту два: «mushroom class» «property_num» «property_value»

Подпишитесь на нашу рассылку

Ваш email... OK

Скачать Deductor

О КОМПАНИИ РЕКВИЗИТЫ НОВОСТИ ПРЕСС-РЕЛИЗЫ ВАКАНСИИ МЕРОПРИЯТИЯ ОБРАТНАЯ СВЯЗЬ

BaseGroup Labs — профессиональный поставщик программных продуктов и решений в области бизнес-аналитики. Мы специализируемся на разработке систем для глубокого анализа данных, охватывающих вопросы сбора, интеграции, очистки данных, построения моделей и визуализации.

Информация о BaseGroup Labs

© 2017 BaseGroup Labs
ООО «Аналитические технологии»
Пользовательское соглашение

ГОЛОВНОЙ ОФИС

Россия, 390023, Рязань, ул. Новая 53в

+7 (4912) 24-09-77

sale@basegroup.ru

МОСКОВСКИЙ ОФИС

Россия, Москва, Садовническая улица, д. 82, стр. 2, подъезд 6, 2-й этаж

+7 (495) 222-71-17

moscow@basegroup.ru

Кампус

Партнёрский портал

Портал преподавателей

ПРОДУКТЫ

Deductor

Deductor Credit Pipeline

Deductor Credit Scorecard Modeler

Deductor Data Quality

Deductor Demand Planning

Создание сайта —