Ассоциативные правила

Виктор Китов

v.v.kitov@yandex.ru

Решаемая задача

Транзакции в магазине

TID	Items
1	{Bread, Milk}
2	{Bread, Diapers, Beer, Eggs}
3	{Milk, Diapers, Beer, Cola}
4	{Bread, Milk, Diapers, Beer}
5	{Bread, Milk, Diapers, Cola}

Пример правила:

$$\{\texttt{Diapers}\} \longrightarrow \{\texttt{Beer}\}.$$

Применения: анализ рыночных корзин (market basket analysis), медицинская диагностика, анализ данных в др. областях.

- Проблемы:
  - вычислительная сложность перебора всех правил.
  - статистическая значимость правил
  - совместная встречаемость не означает причинно следственной связи (correlation vs. causation)
    - (например, товары лежат рядом на одной полке)

Пример генерации правил в политике

Данные по голосованиям ("товары"):

1. Republican
2. Democrat
3. handicapped-infants = yes
4. handicapped-infants = no
5. water project cost sharing = yes
6. water project cost sharing = no
7. budget-resolution = yes
8. budget-resolution = no
9. physician fee freeze = yes
18. aid to Nicaragua = no
19. MX-missile = yes
20. MX-missile = no
21. immigration = yes
22. immigration = no
23. synfuel corporation cutback = yes
24. synfuel corporation cutback = no
25. education spending = yes
26. education spending = no

Определение принадлежности к партии (предпочтения)

Association Rule	Confidence
{budget resolution = no, MX-missile=no, aid to El Salvador = yes } $\longrightarrow$ {Republican}	91.0%
{budget resolution = yes, MX-missile=yes, aid to El Salvador = no } $\longrightarrow$ {Democrat}	97.5%
{crime = yes, right-to-sue = yes, physician fee freeze = yes} $\longrightarrow$ {Republican}	93.5%
{crime = no, right-to-sue = no, physician fee freeze = no} $\longrightarrow$ {Democrat}	100%

Бинарное представление

- $I = \{i_1, i_2, ...i_d\}$ - все товары, ищем наборы из этих товаров
- $T = \{t_1, t_2, ...t_N\}$ - все транзакции, t_i - подмножество I
- Поддержка (support)

$$\sigma(X) = \left|\{t_i | X \subseteq t_i,\ t_i \in T\}\right|$$

Ассоциативное правило

- Ассоциативное правило $X \to Y$, где X, Y - наборы товаров, $X \cap Y = \emptyset$

Свойства ассоциативного правила:

$$\text{Support, } s(X \longrightarrow Y) = \frac{\sigma(X \cup Y)}{N};$$

$$\text{Confidence, } c(X \longrightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}.$$

- $X \cup Y$ - наборы X и Y одновременно встретились (пересечение).
- Примеры расчетов.
- Правило с низкой поддержкой может появиться случайно, нет смысла делать промоакции для редких правил.
- Уверенность измеряет, насколько правило надежно.

Мера lift

Мера lift показывает, полезнее ли правило случайного угадывания:

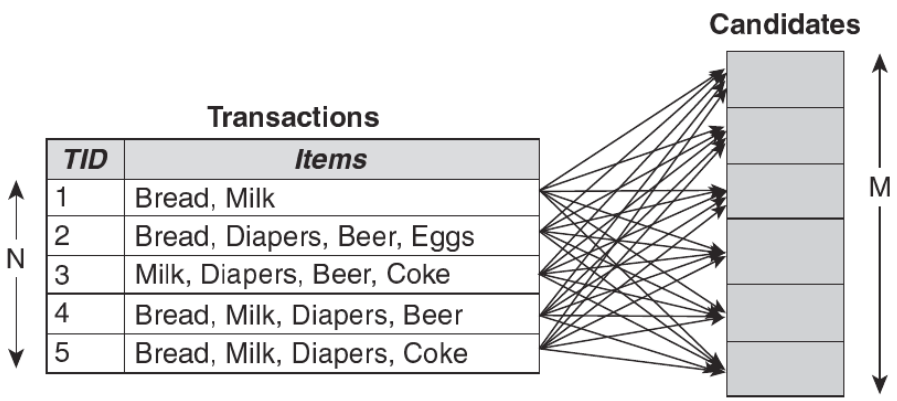
$$\text{lift}(A \to B) = \frac{\text{support}(A \cup B)}{\text{support}(A) \cdot \text{support}(B)} =$$

$$= \frac{|\{A \cup B \subseteq x \,|\, x \in X_{\text{train}}\}| \cdot |X_{\text{train}}|}{|\{A \subseteq x \,|\, x \in X_{\text{train}}\}| \cdot |\{B \subseteq x \,|\, x \in X_{\text{train}}\}|}$$

Нахождение правил

- Нахождение правил состоит из 2х этапов:
  1. генерация частых наборов ($support \geq minsup$)
  2. генерация уверенных правил по наборам ($confidence \geq minconf$)
- Первая задача вычислительно сложнее.
  - полный перебор: сложность $O(N(2^k - 1)w)$ для наборов длины k, #транзакций N и средней длины транзакции w.
  - решения: $\downarrow k$ (Apriori), $\downarrow$ число сравнений с транзакциями (FP-growth)

Полный перебор ($M = 2^k - 1$)



Apriori

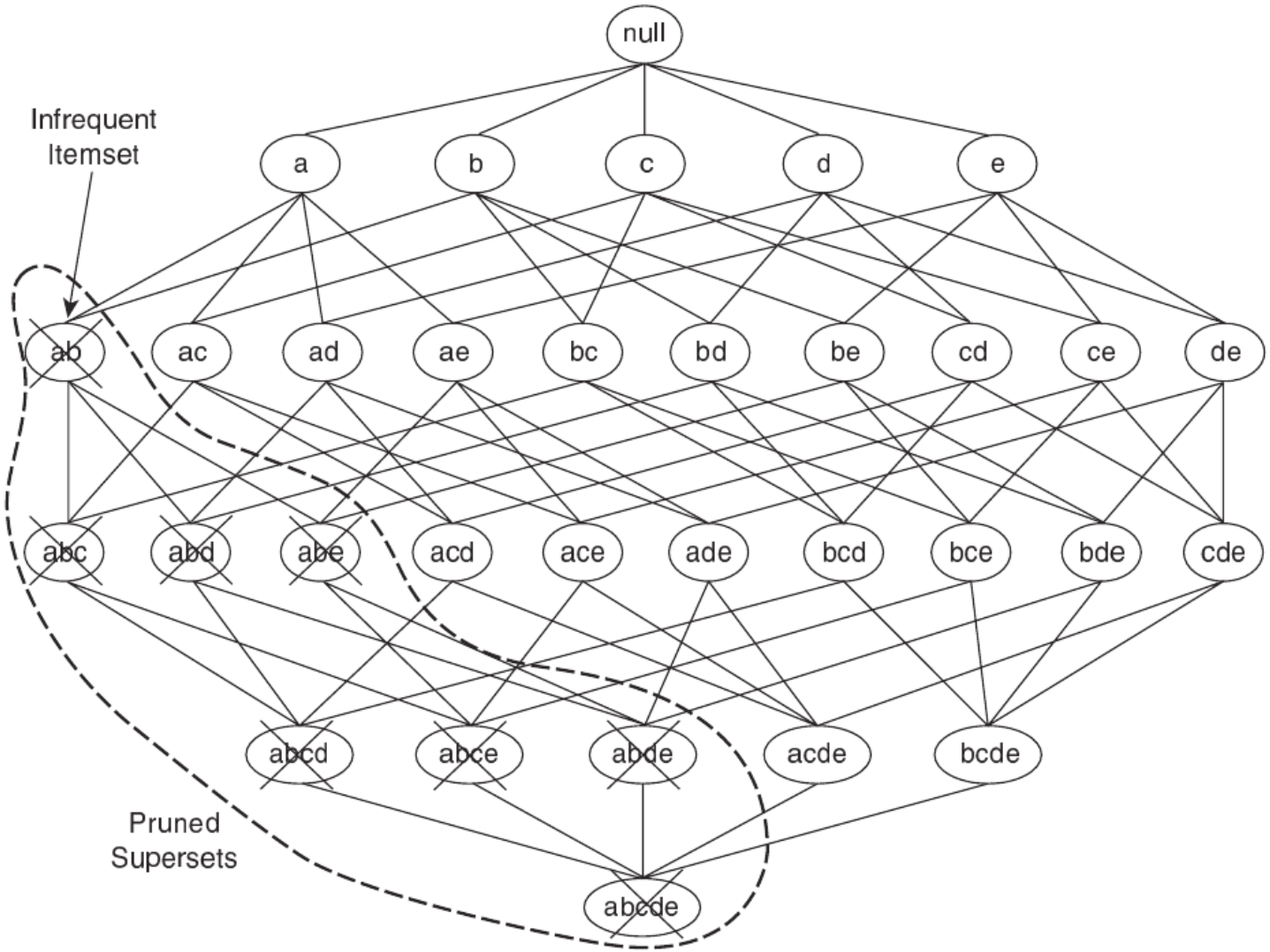
- У следующих правил - одинаковая поддержка:

$$\{\texttt{Beer, Diapers}\} \longrightarrow \{\texttt{Milk}\}, \quad \{\texttt{Beer, Milk}\} \longrightarrow \{\texttt{Diapers}\},$$
$$\{\texttt{Diapers, Milk}\} \longrightarrow \{\texttt{Beer}\}, \quad \{\texttt{Beer}\} \longrightarrow \{\texttt{Diapers, Milk}\},$$
$$\{\texttt{Milk}\} \longrightarrow \{\texttt{Beer,Diapers}\}, \quad \{\texttt{Diapers}\} \longrightarrow \{\texttt{Beer,Milk}\}.$$

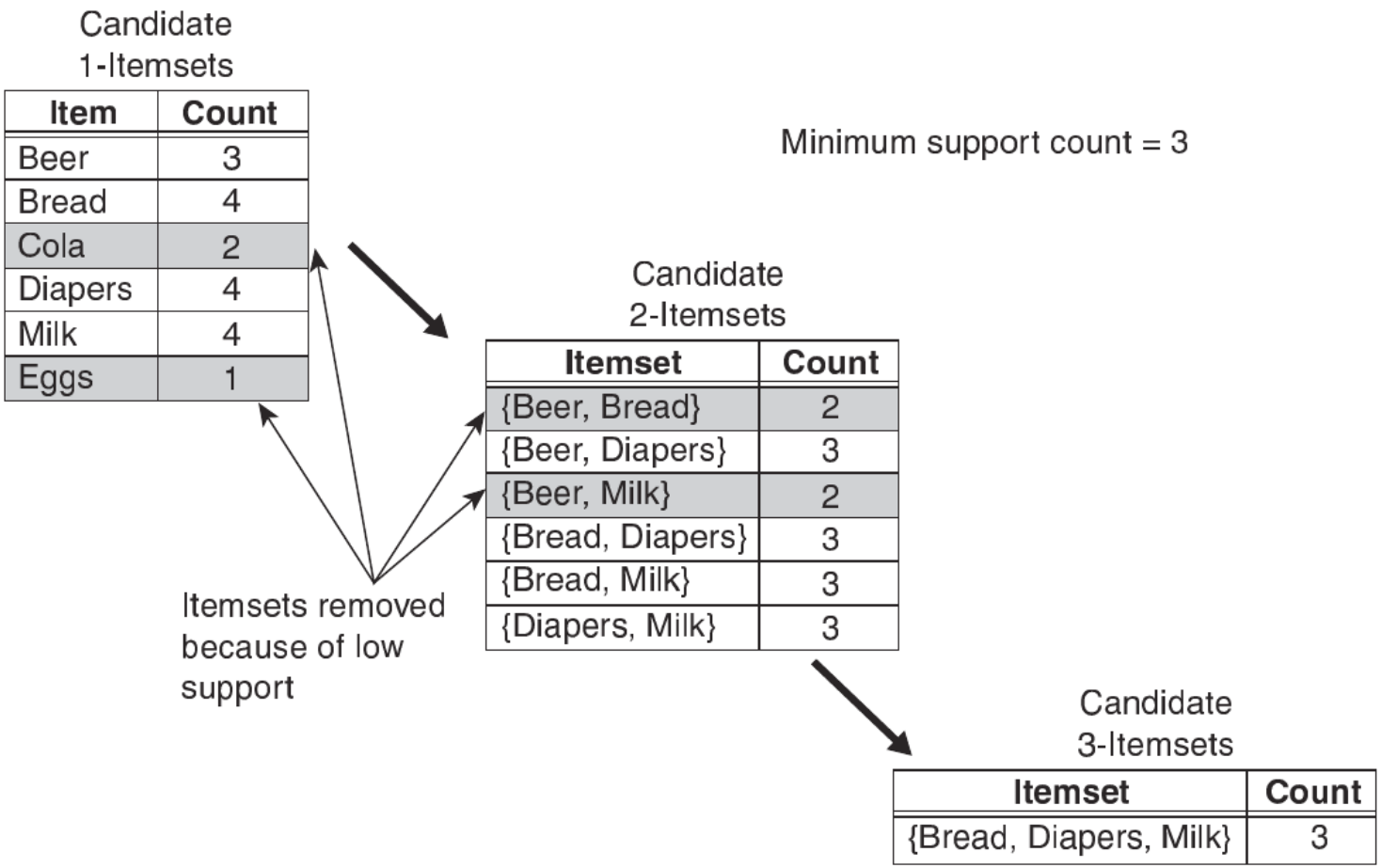
- Идея алгоритма Apriori - если набор частый, то все его поднаборы тоже частые.
- Математически это свойство антимонотонности поддержки:

$$X' \subset X \;\Rightarrow\; \sigma(X') \geq \sigma(X)$$

Идея Apriori



Поиск, использующий антимонотонность



Экономия вычислений

- Экономия перебора:

$$\binom{6}{1} + \binom{6}{2} + \binom{6}{3} \to \binom{6}{1} + \binom{4}{2} + 1$$

- Алгоритм:
  - проход наборам одного товара, выбор частых F_1
  - для $k = 1, 2, 3, \ldots$ пока $F_k \neq \emptyset$:
    - генерация F_{k+1} комбинацией $\{F_k\}_k$

Генерация F_k

- Не должны генерироваться кандидаты, содержащие нечастые k поднаборы.
- Перебор кандидатов должен быть полным (среди потенциально подходящих).
- Сгенерированные кандидаты не должны дублироваться.
  - $\{a,b,c,d\} = \{a,b\} + \{c,d\} = \{a,b,c\} + \{d\} = \{a\} + \{b,c,d\} = ...$
- Подходы генерации:
  - полный перебор, сложность C_k^d, d=#товаров, k-длина набора

Подходы генерации

- Подходы генерации
  - $F_{k-1} \times F_1$: комбинация всех F_{k-1} и F_1, сложность $O(|F_{k-1}| \times |F_1|)$
  - могут генерироваться повторения: {Bread, Diapers}+{Milk}={Bread,Milk}+{Diapers}=...
    - решение: объединять только наборы по возрастанию в лексикографическом порядке: {Bread, Diapers}+{Milk}, ~~{Bread,Milk}+{Diapers}~~
    - $F_{k-1} \times F_{k-1}$: объединяются только при условии

$$a_i = b_i, i = 1, 2, ...k-2; \;\; a_{k-1} \neq b_{k-1}$$

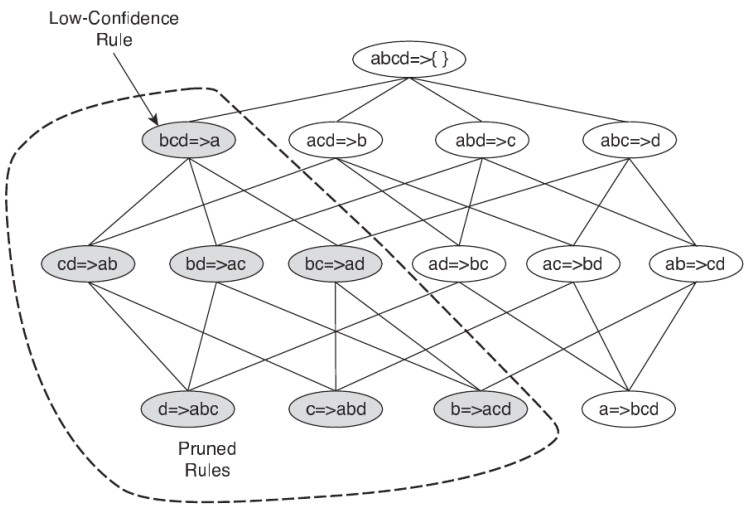
    - перебор полный и ↑эффективность от лексикографического упорядочивания.

Генерация правил

- Для набора длины k существует $2^k - 2$ правил
  - по набору X генерируем $Y \to X - Y$
    - X-частый, значит и поднаборы $Y, X - Y$-частые.
  - игнорируем $\emptyset \to Y$ и $Y \to \emptyset$
- Оптимизация перебора правил: если $X \to Y - X$ малой уверенности (confidence), то любое $X' \to Y - X'$, $X' \subset X$ - тоже малой уверенности, т.к. $\sigma(X') \geq \sigma(X)$
  - $conf(X \to Y - X) = \frac{\sigma(Y)}{\sigma(X)} \geq \frac{\sigma(Y)}{\sigma(X')} = conf(X' \to Y - X')$
  - т.е. если $conf(bcd => a)$ мало, то $conf(cd => ab)$ еще меньше.

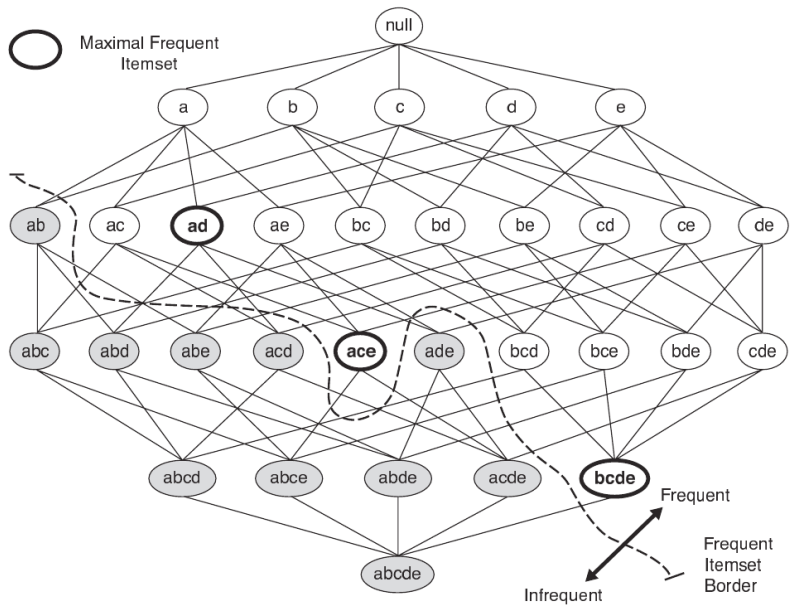
Оптимизация перебора правил в Apriori

Использование принципа для оптимизации перебора правил:



Компактное представление частых наборов

- Для компактного представления частых наборов достаточно знать только нерасширяемые частые наборы (maximal frequent itemsets).
- Все поднаборы нерасширяемые частые наборов - частые.Но теряется информация о поддержке.



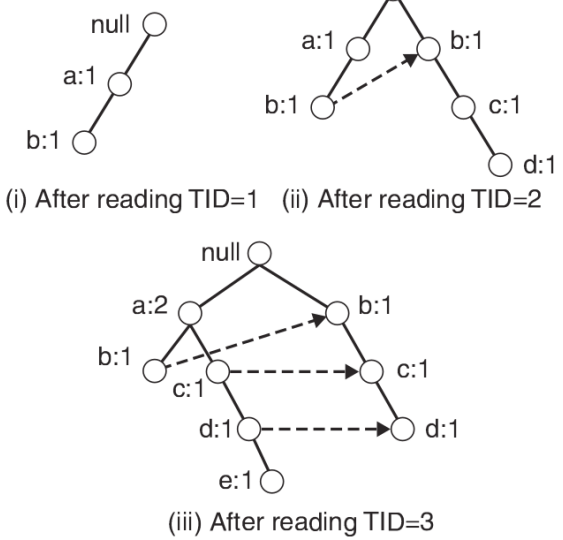
FP-growth

FP-growth алгоритм использует структуру данных для компактного представления наборов и их поддержек.

дерево FP-growth:

Transaction Data Set

TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}



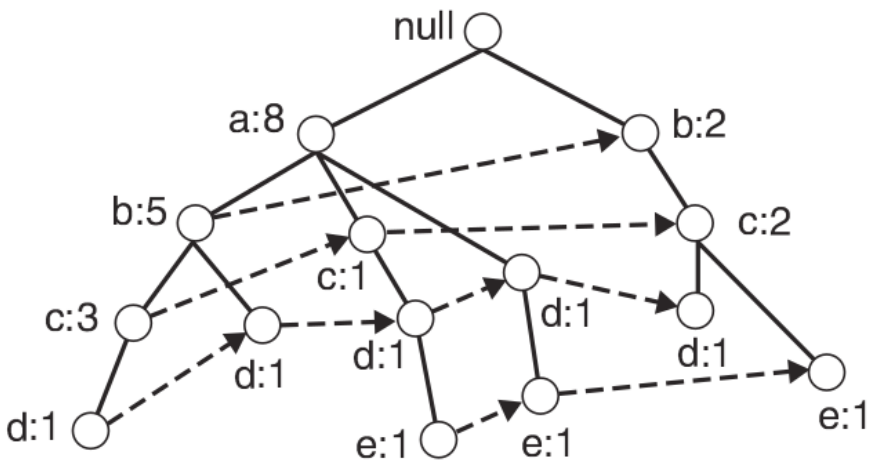
- Узлы - частые товары, упорядоченные по $\downarrow$поддержки.
- Узлы, отвечающие одинаковым товарам соединены указателем (уровень дерева)

Полное FP-дерево

Transaction Data Set

TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}

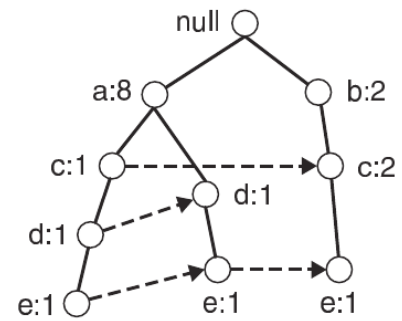
Полное FP-дерево



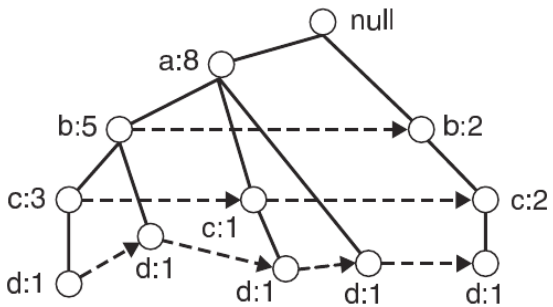
(iv) After reading TID=10

Поиск частых наборов

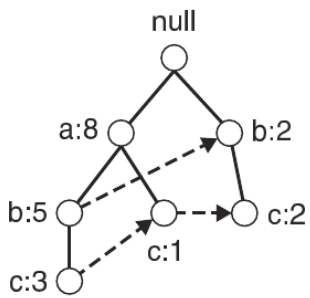
Можем найти частые наборы, оканчивающиеся на заданный суффикс, например "e":



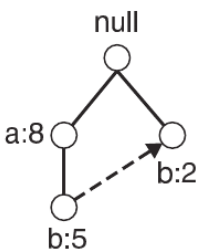
(a) Paths containing node e



(b) Paths containing node d



(c) Paths containing node c

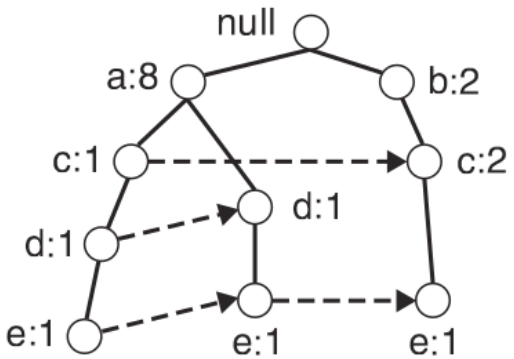


(d) Paths containing node b

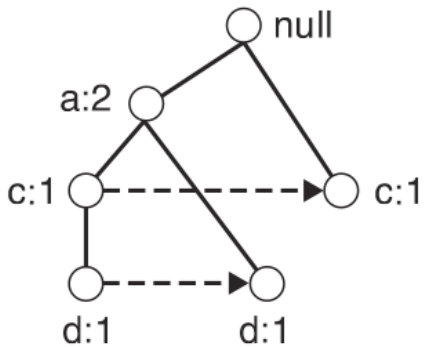
(e) Paths containing node a

Извлеченные наборы

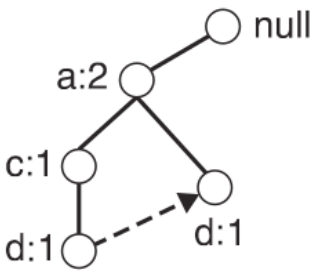
- Находим поддержку {e}: сумируем поддержки всех путей, заканчивающихся на e, получим 3.



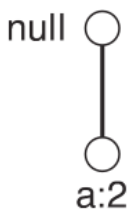
(a) Prefix paths ending in e



(b) Conditional FP-tree for e



(c) Prefix paths ending in de



(d) Conditional FP-tree for de

Извлеченные наборы

- Т.к. {e} - частый, то ищем частые поднаборы с расширенным суффиксом: de,ce,be,ae.
- Рассмортим de. Строим условное FP-дерево - при условии окончания на e.
  1. Обновляем счетчики: например, путь null->b:2->c:2->e:1 содержит {b,c}. Т.к. условие=окончание на e, то пересчитываем поддержку:

$$null->b:1->c:1->e:1$$

  2. некоторые товары могут перестать быть частыми в условном дереве, например "b", т.к. #{b,e}=1.
  3. убираем узлы с "e" (т.к. счетчики выше уже учитывают наличие "e" в конце)

Заключение

- Ассоциативные правила - эффективный метод извлечения интерпретируемых зависимостей в дискретных данных.
  - анализ покупательских корзин, политика, экология и др.
- Нужно быстро
  - искать частотные наборы (Apriori, FP-growth)
  - правила (Apriori)
- FP-growth быстрее Apriori.
- Развернутый обзор темы.