УДК 004.827:510.635

МЕТОД ПОИСКА ОБОБЩЕННЫХ АССОЦИАТИВНЫХ ЗАВИСИМОСТЕЙ МЕЖДУ ДИСКРЕТНЫМИ ПРИЗНАКАМИ

к.т.н. Д.Э. Ситников, Е.В. Титова
(представил д.т.н., проф. О.И. Сухаревский)

Предлагается алгоритм нахождения обобщенных ассоциативных правил для признаков объектов в базах данных при условии, что признаки являются категорированными величинами. Основой алгоритма является построение дерева покрытий, которое позволяет генерировать ассоциации с соответствующей поддержкой, а также легко учитывать обновление записей в базе данных.

Установление зависимости между признаками объектов – перспективная область развития баз данных (БД), искусственного интеллекта и обучения машин. В отличие от традиционного подхода, который рассматривает каждый объект (запись в БД) отдельно, Knowledge Discovery in DataBases (KDD) – нахождение знаний в БД, оперирует с информацией, основанной на связи признаков объектов друг с другом. Обнаружение такой связи – вопрос построения (генерации) ассоциативных правил. Ассоциативное правило определяется как выражение вида $A \rightarrow B$, где A и B – некоторые множества признаков объектов. Для каждого ассоциативного правила вводятся понятия поддержки (Support) и уровня доверия (Confidence). Поддержка определяется как количество записей (объектов), удовлетворяющих данному правилу, а доверие – как отношение между количеством записей, удовлетворяющих правилу и количеством записей, удовлетворяющих его левой части.

В [1] предлагается алгоритм построения ассоциативных правил для случая, когда признаки объектов описываются бинарными переменными (т.е. объект или обладает данным признаком, или не обладает). В реальных же базах данных часто признаки могут принимать значения из некоторого набора (множества значений), т.е. являются категорированными величинами (разбиты на категории). Бинарные значения являются лишь частным случаем более общего подхода – признаки разбиты на две категории, множество значений состоит из двух элементов – 0 и 1.

Алгоритм, предложенный в [1], основан на нахождении покрытий, т.е. наборов признаков, имеющих поддержку большую или равную некоторому установленному порогу *minSupport*. Если известны все покрытия и их поддержка, то генерация ассоциативных правил сводится к нахождению всевозможных разбиений покрытия на подмножества: если S – покрытие, то для любых $S_1 \subset S$ и $S_2 = S - S_1$ ассоциативным правилом является $S_1 \rightarrow S_2$,

ISSN 1681-7710. Системи обробки інформації, випуск 6(22), 2002

если имеет необходимый уровень доверия:

$$Sup(S_1 \cup S_2) \geq minSup;$$

$$\frac{Sup(S_1 \cup S_2)}{Sup(S_1)} \geq minConf.$$

Кроме того, в [1] отмечено свойство монотонности покрытия с точки зрения сжатия множества признаков, т.е. если S является покрытием и $S'' \subset S$, то S'' также является покрытием. С другой стороны, ассоциативные правила не обладают свойством монотонности с точки зрения расширения и сжатия, т.е. если $S \to S''$ – ассоциативное правило, то $S \cup S'' \to S'$ не обязательно является таковым (может не обладать достаточным уровнем поддержки). Также, если $S \cup S'' \to S'$ – ассоциативное правило, то $S \to S'$ может не обладать достаточным уровнем доверия.

Мы предлагаем обобщение этого алгоритма на случай, когда признаки объектов являются категорированными величинами. Кроме того, наш алгоритм позволяет строить правила вида

$$\left(X^{a_1} \vee X^{a_2} \vee ... X^{a_i}\right)\left(Y^{b_3} \vee ... Y^{b_k}\right) \to \left(Z^{c_3} \vee .. Z^{c_j}\right)\left(t^{d_2} \vee ... \vee t^{d_m}\right)$$

т.е. если признак X объекта принимает значения из множества $\{a_1, a_2, ..., a_i\}$, и признак Y принимает значения из множества $\{b_3, ..., b_k\}$, то признак Z принимает значение из множества $\{c_3, ..., c_j\}$ и признак t принимает значения $\{d_2, ..., d_m\}$.

1. Описание признаков объектов с помощью конечных предикатов. Если признаки объектов являются категорированными величинами, то удобным способом их описания являются конечные предикаты. Пусть признак X_j объекта может принимать значения из множества $A=\{a_1, a_2, ..., a_n\}$, тогда такой признак может быть описан с помощью конечного предиката следующим образом:

$$X_j^{a_i} = \begin{cases} 1, & \text{если } X_j = a_i; \\ 0 & \text{если } X_j \neq a_i. \end{cases}$$

Предикат $X_j^{a_i}$ как бы "узнает" значение a_i признака X_j среди всех возможных значений $a \in A$ [2].

Имеют место следующие тождества:

$$X^{a_1} \vee X^{a_2} \vee ... \vee X^{a_n} \equiv 1; \quad (1)$$

$$X^{a_i} X^{a_j} \equiv 0. \quad (2)$$

Действительно, признак X принимает какое-либо значение из возможного набора, а, следовательно, всегда один из дизъюнктивных членов, а вместе с ним и вся дизъюнкция (1) обращается в 1, и признак X не может принимать одновременно два различных значения.

Подобное описание позволяет не использовать отрицание признаков:

если признак X_j не принимает значение a_i, то он принимает любое значение из оставшегося набора признаков, т.е.

$$\overline{X^{a_i}} = X^{a_1} \vee .. X^{a_{i-1}} \vee X^{a_{i+1}} ... \vee X^{a_n} .$$

В случае бинарных признаков для их описания используется унарный предикат, различающий два значения – 0 и 1: X^0 и X^1.

Таким образом, не требуется строить отдельные алгоритмы для нахождения ассоциативных правил с отрицанием. Нахождение исключающей ассоциации (т.е. ассоциативного правила, включающего отрицание какого-либо признака), данной в [1], сводится к нахождению обычного покрытия, в которое могут входить предикаты вида X^0.

2. Построение дерева покрытий. Для нахождения покрытий в [1] предлагается использовать алгоритм построения дерева, который позволяет, во-первых, генерировать все покрытия, во-вторых, легко обновлять информацию при поступлении в БД новых записей или уничтожении старых записей. Кроме того, использование дерева позволяет считывать БД только один раз, что существенно сокращает время расчетов.

Определим дерево покрытий рекурсивно следующим образом.

Пусть существует множество признаков $X=\{X_1, X_2, ..., X_m\}$. Каждому признаку X_i соответствует множество его значений $\Sigma_i = \left\{\Sigma_{i1}, \Sigma_{i2}, \ldots, \Sigma_{in_i}\right\}$.

1) Корнем дерева является вершина, представляющая собой нулевую строку λ.

2) Вершины первого яруса (вершины, дочерние к λ): $\bigcup_{i=1}^{m} X_i^{\Sigma_{ji}}, \left(j = \overline{1, n_i}\right)$.

3) Вершины, дочерние к вершине $X_k^{\Sigma_{jk}}, \left(j = \overline{1, n_k}\right)$: $\bigcup_{i=k+1}^{m} X_i^{\Sigma_{ji}}, \left(j = \overline{1, n_i}\right)$.

Построение дерева.

1) считывание БД;

2) для каждой записи (объекта), обладающей k признаками:

for i = 1 to k do

для каждого подмножества признаков:

пусть S_i – строка длины i, состоящая из признаков объекта: $S_i = a_1 a_2 ... a_i$;

если существует путь $\lambda \to a_1 \to a_2 \to ... \to a_i$ то добавить к поддержке вершин $a_1, a_2, ..., a_i$ по 1,

иначе создать ветвь $\lambda \to a_1 \to a_2 \to ... \to a_i$ и добавить к поддержке вершин $a_1, a_2, ..., a_i$ по 1

end.

Любая строка, получаемая конкатенацией вершин при прохождении по любой ветви дерева от нулевой вершины до данной вершины $X_i^{\Sigma_{ji}}$, потен-

циально является покрытием. Его поддержка равна поддержке последней вершины в данной строке. Мы рассматриваем только те покрытия, поддержка которых больше или равна заданному уровню *minSupport*.

Рассмотрим пример, когда признак X объектов может принимать значения a_1, a_2, a_3, признак Y – значения b_1, b_2, b_3 и признак Z – значения c_1, c_2, c_3 (табл 1). Задана минимальная поддержка *minSup=2*.

Таблица 1

Фрагмент реляционной БД

	Объекты (записи в БД)							
Признаки	1	2	3	4	5	6	7	8
X	a_1	a_2	a_1	a_3	a_1	a_2	a_3	a_1
Y	b_1	b_3	b_2	b_2	b_1	b_3	b_2	b_2
Z	c_1	c_1	c_2	c_3	c_1	c_1	c_2	c_1

Построенное дерево показано на рис. 1. Возле каждой вершины указана ее поддержка в круглых скобках.

Получены следующие покрытия:

1) $X^{a_1}Y^{b_1}$ *Support=2*;
2) $X^{a_1}Y^{b_1}Z^{c_1}$ *Support=2*;
3) $X^{a_1}Y^{b_2}$ *Support=2*;
4) $X^{a_1}Z^{c_1}$ *Support=3*;
5) $X^{a_2}Y^{b_3}$ *Support=2*;
6) $X^{a_2}Y^{b_3}Z^{c_1}$ *Support=2*;
7) $X^{a_2}Z^{c_1}$ *Support=2*;
8) $X^{a_3}Y^{b_2}$ *Support=2*;
9) $Y^{b_1}Z^{c_1}$ *Support=2*;
10) $Y^{b_2}Z^{c_2}$ *Support=2*;
11) $Y^{b_3}Z^{c_1}$ *Support=2*.

Для построения ассоциативного правила используем следующий алгоритм.

Для каждого покрытия $S = s_1s_2...s_k$ выделяем подстроку s_i и находим его поддержку $Sup(S - s_i)$ путем прохождения по ветви дерева, соответствующей этому покрытию. Если $\frac{Sup(S)}{Sup(S-s_i)} \geq \min Conf$, то $S - s_i \to s_i$ является ассоциативным правилом.

Например:

1) $X^{a_1}Y^{b_1} \to Z^{c_1}$ *Confidence=1*;
2) $X^{a_1} \to Y^{b_1}Z^{c_1}$ *Confidence=0,5*;
3) $Z^{c_1} \to X^{a_2}$ *Confidence=0,4*;
4) $X^{a_1}Z^{c_1} \to Y^{b_1}$ *Confidence=2/3* и т.д.

Отметим, что при задании уровня доверия *minConf* = 0,6 правила 2 и 3

не отвечают поставленным условиям и должны быть отброшены.

Оценим сложность построения дерева покрытий. Пусть БД состоит из n объектов, каждому из которых соответствует m признаков. В бинарном случае, т.е. когда признаки принимают значения 0 и 1, общее количество вершин дерева $L \le n \cdot \sum_{i=1}^{m} C_m^i$.

Если количество категорий больше двух, то при выборе любых j признаков, количество вершин увеличивается, соответственно в $k_1 k_2 \ldots k_j$ раз, где k_i – количество категорий для i-го признака.

Таким образом, предлагаемый алгоритм имеет достаточно большую сложность, что накладывает некоторые ограничения на область его применения: количество признаков объектов в БД должно быть невелико. Это же ограничение касается и категорий признаков. Однако, количество объектов (записей в БД) может представлять собой достаточно большое число.

3. Построение обобщенных правил вида

$$\left(X^{a_1} \vee X^{a_2} \vee \ldots X^{a_i}\right)\left(Y^{b_3} \vee \ldots \vee Y^{b_k}\right) \to \left(Z^{c_3} \vee \ldots \vee Z^{c_j}\right)\left(t^{d_2} \vee \ldots \vee t^{d_m}\right).$$

В случае, когда признаки объекта являются категорированными величинами, интерес представляют обобщенные правила, когда признаки принимают значения из некоторого множества возможных.

Для построения обобщенных правил используется алгоритм нахождения обобщенных покрытий.

Из всех ветвей дерева мы отмечаем те, которые содержат все признаки. В нашем примере таких ветвей 6:

1) $X^{a_1}(4), Y^{b_1}(2), Z^{c_1}(2)$;

2) $X^{a_1}(4), Y^{b_2}(2), Z^{c_1}(1)$;

3) $X^{a_1}(4), Y^{b_2}(2), Z^{c_2}(1)$;

4) $X^{a_2}(2), Y^{b_3}(2), Z^{c_1}(2)$;

5) $X^{a_3}(2), Y^{b_2}(2), Z^{c_2}(1)$;

6) $X^{a_3}(2), Y^{b_2}(2), Z^{c_3}(1)$.

Обобщенное покрытие строится путем объединения этих ветвей следующим образом.

Если мы хотим объединить 2 или более ветвей, мы начинаем двигаться по ним сверху вниз (от начала до конца). Между вершинами, отмеченными разными значениями одного признака, ставиться знак $\vee$. Между вершинами, отмеченными разными признаками, ставится знак $\wedge$.

Например, объединим ветви 1 и 2:

$$\left(X^{a_1} \vee X^{a_1}\right)\left(Y^{b_1} \vee Y^{b_2}\right)\left(Z^{c_1} \vee Z^{c_1}\right) = X^{a_1}\left(Y^{b_1} \vee Y^{b_2}\right)Z^{c_1}.$$

Мы получили покрытие $X^{a_1}\left(Y^{b_1} \vee Y^{b_2}\right)Z^{c_1}$.

Поддержка такого покрытия считается так.

При движении по ветвям дерева, которые мы объединяем: если значения признаков для первых *k* вершин совпадают, то поддержка объединенной вершины равна поддержке исходных вершин. Если значение признаков для вершины *i* не совпало, то с этого момента поддержка объединенной вершины получается суммированием поддержек вершин, ее составляющих:

$$X^{a_1}(4)\left(Y^{b_1} \vee Y^{b_2}(2+2)\right)Z^{c_1}(2+1) = X^{a_1}(4)\left(Y^{b_1} \vee Y^{b_2}(4)\right)Z^{c_1}(3).$$

Общая поддержка покрытия равна минимуму из поддержек его вершин: $\min\{4,4,3\}=3$, *Support=3*.

Ассоциативные правила получаются из данных покрытий аналогично описанному ранее: путем разбиения покрытия на подмножества. Уровень доверия (Confidence) также считается аналогично, т.е. равен отношению поддержки обобщенного покрытия к поддержке его правой части. Для приведенного покрытия составляем следующие правила:

$X^{a_1}\left(Y^{b_1} \vee Y^{b_2}\right) \to Z^{c_1}$ *Confidence = 3/4*;

$\left(Y^{b_1} \vee Y^{b_2}\right)Z^{c_1} \to X^{a_1}$ *Confidence = 3/3=1*;

$X^{a_1}Z^{c_1} \to \left(Y^{b_1} \vee Y^{b_2}\right)$ *Confidence = 3/3 =1*.

Используя свойство монотонности покрытия, мы можем построить также правила следующего вида:

$X^{a_1} \to Z^{c_1}$ *Confidence = 3/4*;

$\left(Y^{b_1} \vee Y^{b_2}\right) \to Z^{c_1}$ *Confidence = 3/6=0,5*;

$X^{a_1} \to \left(Y^{b_1} \vee Y^{b_2}\right)$ *Confidence = 4/4 = 1*.

Обобщенные покрытия, таким образом, включают в себя простые покрытия (покрытие $X^{a_1}\left(Y^{b_1} \vee Y^{b_2}\right)Z^{c_1}$ включает в себя покрытие $X^{a_1}Z^{c_1}$, полученное при построении простых правил).

Продемонстрируем работу алгоритма на других примерах.

Объединим ветви 1, 5 и 6:

$$\left(X^{a_1} \vee X^{a_3}\right)\left(Y^{b_1} \vee Y^{b_2}\right)\left(Z^{c_1} \vee Z^{c_2} \vee Z^{c_3}\right).$$

Согласно тождеству (1) $Z^{c_1} \vee Z^{c_2} \vee Z^{c_3} \equiv 1$. Следовательно,

$$\left(X^{a_1} \vee X^{a_3}\right)\left(Y^{b_1} \vee Y^{b_2}\right)\left(Z^{c_1} \vee Z^{c_2} \vee Z^{c_3}\right)=$$
$$=\left(X^{a_1} \vee X^{a_3}\right)\left(Y^{b_1} \vee Y^{b_2}\right) \wedge 1=\left(X^{a_1} \vee X^{a_3}\right)\left(Y^{b_1} \vee Y^{b_2}\right).$$

Получено покрытие $\left(X^{a_1} \vee X^{a_3}\right)\left(Y^{b_1} \vee Y^{b_2}\right)$. *Support = 6*.

Строим правила:

$\left(X^{a_1} \vee X^{a_3}\right) \to \left(Y^{b_1} \vee Y^{b_2}\right)$ *Confidence = 1*;

$\left(Y^{b_1} \vee Y^{b_2}\right) \to \left(X^{a_1} \vee X^{a_3}\right)$ *Confidence = 1*.

Объединим ветви 1 и 6:

$\left(X^{a_1} \vee X^{a_3}\right)\left(Y^{b_1} \vee Y^{b_2}\right)\left(Z^{c_1} \vee Z^{c_3}\right)$; *Support = 4*.

Строим правила:

$\left(X^{a_1} \vee X^{a_3}\right)\left(Y^{b_1} \vee Y^{b_2}\right) \to \left(Z^{c_1} \vee Z^{c_3}\right)$ *Confidence = 4/6 =2/3*;

$\left(X^{a_1} \vee X^{a_3}\right)\left(Z^{c_1} \vee Z^{c_3}\right) \to \left(Y^{b_1} \vee Y^{b_2}\right)$ *Confidence = 4/4 = 1*;

$\left(Y^{b_1} \vee Y^{b_2}\right)\left(Z^{c_1} \vee Z^{c_3}\right) \to \left(X^{a_1} \vee X^{a_3}\right)$ *Confidence = 4/4 = 1*;

$\left(X^{a_1} \vee X^{a_3}\right) \to \left(Y^{b_1} \vee Y^{b_2}\right)\left(Z^{c_1} \vee Z^{c_3}\right)$ *Confidence = 4/6 = 2/3*.

Отметим еще одну особенность подсчета уровня доверия обобщенных правил.

Теорема. Уровень доверия (Confidence) правила вида $S \to Z^{c_1} \vee Z^{c_2} \vee \ldots \vee Z^{c_k}$ равен сумме уровней доверия правил, полученных из него путем последовательного отбрасывания из правой части различных k–1 дизъюнктивных членов.

Доказательство. Уровень доверия правила $S_1 \to S_2$ определяется как $Conf\left(S_1 \to S_2\right)=\dfrac{Sup\left(S_1 \cup S_2\right)}{Sup\left(S_1\right)}$. Следовательно,

$$Conf\left(S \to Z^{c_1} \vee Z^{c_2} \vee \ldots \vee Z^{c_k}\right)=\frac{Sup\left(S \cup \left(Z^{c_1} \vee Z^{c_2} \vee \ldots \vee Z^{c_k}\right)\right)}{Sup\left(S\right)}=$$
$$=\frac{Sup\left(\left(S \cup Z^{c_1}\right) \vee \left(S \cup Z^{c_2}\right) \vee \ldots \vee \left(S \cup Z^{c_k}\right)\right)}{Sup\left(S\right)}.$$

Но так как $Z^{c_1}, Z^{c_2}, \ldots, Z^{c_k}$ описывают взаимно непересекающиеся множества объектов, то

$$Sup\left(\left(S \cup Z^{c_1}\right) \vee \left(S \cup Z^{c_2}\right) \vee \ldots \vee \left(S \cup Z^{c_k}\right)\right)=$$
$$=Sup\left(S \cup Z^{c_1}\right)+Sup\left(S \cup Z^{c_2}\right)+\ldots+Sup\left(S \cup Z^{c_k}\right).$$

Следовательно,

$$\frac{Sup\left(S \cup Z^{c_1}\right) \vee \left(S \cup Z^{c_2}\right) \vee \ldots \vee \left(S \cup Z^{c_k}\right)}{Sup(S)} =$$

$$= \frac{Sup\left(S \cup Z^{c_1}\right) + Sup\left(S \cup Z^{c_2}\right) + \ldots + \left(S \cup Z^{c_k}\right)}{Sup(S)} = \frac{Sup\left(S \cup Z^{c_1}\right)}{Sup(S)} +$$

$$+ \frac{Sup\left(S \cup Z^{c_2}\right)}{Sup(S)} + \ldots\ldots + \frac{\left(S \cup Z^{c_k}\right)}{Sup(S)} =$$

$$= Conf\left(S \to Z^{c_1}\right) + Conf\left(S \to Z^{c_2}\right) + \ldots + Conf\left(S \to Z^{c_k}\right)$$

Таким образом, например, Confidence правила $X^{a_1} Z^{c_1} \to \left(Y^{b_1} \vee Y^{b_2}\right)$ можно получить как

$$Conf\left(X^{a_1} Z^{c_1} \to Y^{b_1}\right) + Conf\left(X^{a_1} Z^{c_1} \to Y^{b_2}\right) = \frac{2}{3} + \frac{1}{3} = 1.$$

При этом обобщенное правило дает более высокий уровень доверия, чем составляющие его простые правила. Например, при заданном *minConf*=0,8 ни правило $X^{a_1} Z^{c_1} \to Y^{b_1}$, ни правило $X^{a_1} Z^{c_1} \to Y^{b_2}$ не обладают достаточным уровнем доверия, в то время как обобщенное правило отвечает поставленным требованиям.

Выводы. В статье предложен алгоритм построения ассоциативных правил для случая, когда признаки объектов являются категорированными величинами. Этот алгоритм является расширением и усовершенствованием алгоритма, предложенного в [1]. Кроме того, предлагаемый алгоритм позволяет строить обобщенные правила, т.е. правила, когда признаки объектов принимают значения из множества возможных.

ЛИТЕРАТУРА

1. *Amir A., Feldman R., Kashi R. A new and versatile method for association generation // Information Systems. – 1997. – Vol. 22, No 6/7. – P. 333 – 347.*
2. *Шабанов-Кушнаренко Ю. П. Теория интеллекта. Математические средства. – Х.: Вища шк., 1984. – 177 с.*

Поступила 23.10.2002

***СИТНИКОВ Дмитрий Эдуардович**, канд. техн. наук, доцент, зав. кафедрой информационно-документных систем ХГАК. В 1988 году окончил Харьковский институт радиоэлектроники. Область научных интересов – применение алгебры конечных предикатов для обработки информации в БД.*

***ТИТОВА Елена Витольдиевна**, младший научный сотрудник НЦ Войск ПВО. В 1988 году окончила Харьковский институт радиоэлектроники. Область научных интересов – применение алгебры конечных предикатов для обработки информации в БД.*