ИНЖИНИРИНГ ОНТОЛОГИЙ

УДК 004.89, 004.832 *Научная статья* DOI: 10.18287/2223-9537-2023-13-3-392-404

Поиск зависимостей в данных на основе методов удовлетворения табличных ограничений

© 2023, А.А. Зуенко ✉, О.Н. Зуенко

Институт информатики и математического моделирования им. В.А. Путилова, ФИЦ «Кольский научный центр Российской академии наук», Апатиты, Россия

Аннотация

Работа посвящена поиску в данных особого типа закономерностей, называемых частыми паттернами. Под частым паттерном понимается некоторая совокупность признаков, которая характеризует большое количество объектов обучающей выборки. Существующие методы выявления паттернов, как правило, не позволяют гибко учитывать необходимые требования к их виду. Изменение условий, которым должны удовлетворять искомые закономерности, приводит к трудоёмкой модификации используемых алгоритмов и снижению производительности вычислений. В статье предлагается подход на основе парадигмы программирования в ограничениях, свободный от перечисленных недостатков. Подход основан на оригинальном способе представления обучающей выборки с помощью специализированных табличных ограничений – сжатых таблиц *D*-типа, на авторском методе поиска с возвратами, а также на специализированных правилах редукции для табличных ограничений. Особое внимание уделяется решению задачи поиска замкнутых паттернов, которая входит как часть в решение рассматриваемых в работе задач машинного обучения и включает учёт дополнительных требований к виду искомых паттернов. В качестве дополнительных требований к виду паттерна рассматриваются ограничения на частоту встречаемости замкнутого паттерна, а также условия на вхождение некоторого элемента (признака) в паттерн. К основным правилам редукции сжатых таблиц *D*-типа добавляются правила, учитывающие интересующие особенности анализируемых паттернов. Преимуществом подхода является то, что учёт и анализ новых ограничений позволяет на ранних стадиях процесса поиска исключать из рассмотрения заведомо неперспективные кандидаты в паттерны, что способствует сокращению количества этапов вычислений (узлов дерева поиска) и позволяет снизить расход оперативной памяти, требуемой для реализации этих этапов.

Ключевые слова: *программирование в ограничениях, извлечение паттернов, правила редукции, машинное обучение, табличные ограничения.*

Цитирование: *Зуенко А.А., Зуенко О.Н.* Поиск зависимостей в данных на основе методов удовлетворения табличных ограничений // Онтология проектирования. 2023. Т.13, №3(49). С.392-404. DOI:10.18287/2223-9537-2023-13-3-392-404.

Благодарности: работа выполнена в рамках НИР «Разработка теоретических и организационно-технических основ информационной поддержки управления жизнеспособностью региональных критических инфраструктур Арктической зоны Российской Федерации» (регистрационный номер 122022800547-3).

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Введение

В статье для решения задач машинного обучения (МО), связанных с извлечением паттернов, предлагается новый подход на основе парадигмы программирования в ограничениях. Данную парадигму предлагается применять для генерации кандидатов в искомые паттерны.

Любой метод удовлетворения ограничений содержит две обязательные компоненты: одна из них реализует рассуждения на ограничениях, а другая служит для разбиения пространства поиска на меньшие подпространства для их дальнейшего исследования [1]. Рассуждения на ограничениях носят дедуктивный характер и сводятся к последовательному усечению исходных областей определения переменных путём исключения из них «лишних» значений, т.е. значений, которые не способны образовывать ни одного решения.

Статья посвящена разработке авторских методов поиска в данных особого типа закономерностей (причинно-следственных отношений), называемых частыми паттернами. Под частым паттерном понимается совокупность признаков, которая характеризует большое количество объектов обучающей выборки. Существующие методы выявления паттернов не позволяют гибко учитывать дополнительные требования к их виду [2]. Учёт новых условий, которым должны удовлетворять искомые закономерности, приводит к трудоёмкой модификации ранее используемых алгоритмов. Анализ подобных дополнительных условий влечёт снижение производительности вычислений.

В отличие от исследований, посвящённых использованию технологии программирования в ограничениях для выявления зависимостей в данных [3–6], предлагаемый подход основан на оригинальном способе представления обучающей выборки с помощью специализированных табличных ограничений – сжатых таблиц *D*-типа [7], а не на основе представления в виде объектно-признаковой таблицы (бинарной матрицы). Особое внимание уделяется решению задачи поиска замкнутых паттернов, которые являются своеобразным базисом в пространстве рассматриваемых закономерностей. Решение задачи поиска замкнутых паттернов входит как часть в решение рассматриваемых в работе задач МО, включающих учёт и анализ дополнительных требований к виду искомых паттернов. В качестве дополнительных требований к виду паттерна рассмотрены ограничения на частоту встречаемости замкнутого паттерна, а также условия на вхождение некоторого элемента (признака) в паттерн. В работе применяется авторский метод поиска с возвратами, а также специализированные правила редукции для табличных ограничений. Предлагаемый подход позволяет легко адаптироваться к дополнительным требованиям, накладываемым на паттерны. Для этого к основным правилам редукции сжатых таблиц *D*-типа, используемым при поиске замкнутых паттернов, добавляются новые правила, учитывающие интересующие особенности анализируемых закономерностей.

1 Задача удовлетворения табличных ограничений

При использовании парадигмы программирования в ограничениях решаемая задача представляется, как задача удовлетворения ограничений [8]. Задача удовлетворения ограничений заключается в поиске решений для сети ограничений [9]. Сеть ограничений задаётся тремя компонентами $<X,D,C>$: X – множество переменных $\{X_1, X_2, \ldots, X_n\}$, D – множество доменов переменных $\{D_1, D_2, \ldots, D_n\}$, C – множество ограничений $\{C_1, C_2, \ldots C_m\}$, которые регламентируют допустимые сочетания значений переменных. Каждый домен D_i описывает множество допустимых значений $\{v_1,\ldots,v_k\}$ для переменной X_i.

В работе [7] выполнен обзор табличных ограничений, к которым, в частности, относятся сжатые таблицы (компактные таблицы) и смарт-таблицы [10–13].

В настоящей статье используются два типа табличных ограничений для компактного представления многоместных отношений. Первый тип - это сжатые таблицы *C*-типа, а второй тип - это сжатые таблицы *D*-типа.

Пример 1. Пусть дано отношение в виде бинарной матрицы (см. таблицу 1). В рамках МО данная бинарная матрица может рассматриваться как обучающая выборка. Строки мат-

рицы соответствуют объектам, а столбцы – атрибутам. Можно предположить, например, что объекты - это какие-то транзакции покупок, а атрибуты - определённые товары.

Данная бинарная матрица может быть записана как сжатая таблица *C*-типа:

$$R[XY] = \begin{bmatrix} \{1,2,3\} & \{a\} \\ \{1,2,4\} & \{b\} \\ \{2,4,5\} & \{c\} \\ \{2,3\} & \{d\} \end{bmatrix}.$$

Таблица 1 – Пример бинарной матрицы

	a	*b*	*c*	*d*
1	1	1		
2	1	1	1	1
3	1			1
4		1	1	
5			1	

Домен переменной X - это множество объектов {1, 2, 3, 4, 5}, домен переменной Y - это множество атрибутов $\{a, b, c, d\}$. Каждый кортеж данной сжатой таблицы *C*-типа компактно описывает множество ячеек бинарной матрицы, где стоят единицы (т.е. определённое множество элементарных кортежей). Например, первый кортеж компактно описывает множество ячеек первого столбца бинарной матрицы: (1, *a*), (2, *a*), (3, *a*). Вся таблица *C*-типа интерпретируется как объединение множеств элементарных кортежей, описываемых её отдельными строками. Обучающая выборка также может быть представлена в виде сжатой таблицы *D*-типа:

$$K[XY] = \begin{bmatrix} \{1,2,3\} & \{b,c,d\} \\ \{1,2,4\} & \{a,c,d\} \\ \{2,4,5\} & \{a,b,d\} \\ \{2,3\} & \{a,b,c\} \end{bmatrix}.$$

Данный тип табличных ограничений заключён в обратные скобки.

Каждая строка таблицы $K[XY]$ соответствует столбцу исходной бинарной матрицы. Например, первая строка $K[XY]$ соответствует первому столбцу бинарной матрицы и компактно описывает следующее логическое выражение:

$$(Y \in \{a\}) \rightarrow (X \in \{1, 2, 3\}).$$

Эта формула после замены операции импликации может быть записана следующим образом:

$$(Y \notin \{a\}) \vee (X \in \{1, 2, 3\}).$$

Последнее выражение эквивалентно следующей записи:

$$(Y \in \{b, c, d\}) \vee (X \in \{1, 2, 3\}).$$

Сжатая таблица *D*-типа целиком компактно описывает множество истинности конъюнкции импликаций, соответствующих каждой её строке.

При определении рассматриваемых табличных ограничений могут использоваться два типа фиктивных компонент: пустая компонента (“∅”), которая не содержит значений, и полная компонента (“∗”), которая содержит все значения из соответствующего домена.

Стоит отметить, что исходное отношение (бинарная матрица) может быть представлено в виде сжатых таблиц *C*- и *D*-типа не единственным способом. Например, каждый кортеж сжатой таблицы может отображаться не в столбец, а в строку исходной бинарной матрицы.

Как и к обычным таблицам, к сжатым таблицам также можно применять операции реляционной алгебры. Эти операции проводятся с использованием специализированных теорем без разложения сжатых таблиц на обычные таблицы [14].

2 Постановки задачи выявления замкнутых паттернов

Задача поиска замкнутых паттернов состоит в том, чтобы на основе исходного представления обучающей выборки в виде бинарной матрицы (объектно-признаковой таблицы), найти все замкнутые паттерны.

Пусть $I = \{1, ..., n\}$ – множество идентификаторов элементов или атрибутов, а $T = \{1, ..., m\}$ – множество идентификаторов транзакций или объектов. Паттерн p (элементный набор) – это любое подмножество множества I. Транзакционная база данных - это множество $D \subseteq I \times T$. Множество элементов, соответствующих транзакции t, обозначается $D[t]=\{i \mid (i, t) \in D\}$. Транзакция t содержит паттерн p тогда и только тогда, когда множество $D[t]$ содержит p (т.е. $p \subseteq D[t]$). Покрытие паттерна p, обозначается $T_D(p)$, - это множество транзакций, содержащих p (т.е. $T_D(p)=\{t \in T \mid p \subseteq D[t]\}$). Пусть дано подмножество транзакций $S \subseteq T$, тогда $I_D(S)=\cap_{t \in S} D[t]$ описывает множество общих элементов S. Замыкание паттерна p в D, обозначается $Close(p)=I_D(T_D(p))$, - это множество общих элементов его покрытия $T_D(p)$. Паттерн p является замкнутым тогда и только тогда, когда $Close(p)=p$.

Для решения задач, связанных с выявлением паттернов, используются различные алгоритмы, например алгоритм *Apriori* [15]. Выявление замкнутых паттернов — операция, требующая много вычислительных ресурсов и времени, особенно если кроме свойства замкнутости паттерна приходится анализировать дополнительные требования к виду искомых закономерностей. Наиболее популярным требованием к виду паттерна является ограничение на частоту его встречаемости в обучающей выборке.

Абсолютная частота паттерна (абсолютная поддержка паттерна) p – это мощность его покрытия, т.е. $freq_D(p) = |T_D(p)|$. Пусть дана константа $\theta \in N$, называемая минимальной абсолютной поддержкой паттерна (порогом абсолютной поддержки паттерна), тогда паттерн p является частым, если $freq_D(p) \geq \theta$. Задача нахождения частых замкнутых паттернов заключается в нахождении всех паттернов p, таких что $freq_D(p) \geq \theta$ и $Close(p) = p$. Большинство алгоритмов, в том числе модификации алгоритма *Apriori*, используют при поиске паттернов с ограничением на частоту встречаемости свойство антимонотонности: с ростом мощности множества элементов паттерна его покрытие уменьшается либо остаётся тем же. Из данной формулировки следует, что любое k-элементное множество будет часто встречающимся тогда и только тогда, когда все его $(k-1)$-элементные подмножества будут часто встречающимися.

Могут быть сформулированы и другие дополнительные ограничения на вид искомого паттерна, которые можно классифицировать следующим образом:

- ограничения на наличие/отсутствие элемента в паттерне;
- ограничения на подпаттерн и суперпаттерн;
- ограничения на количество элементов в паттерне;
- агрегатные ограничения, представляющее собой количественные ограничения на совокупности элементов в паттерне, где агрегатной функцией может быть сумма, среднее значение, максимум, минимум и т.д.

Например, если в качестве элементов транзакций выступают покупки в некотором магазине, то в качестве агрегатного ограничения может рассматриваться сумма счёта (по чеку), превышающая какую-либо сумму (например, 2000 рублей).

Для демонстрации возможностей предлагаемого подхода в работе рассматривается два типа дополнительных ограничений: ограничения на наличие/отсутствие элемента в паттерне и ограничения на частоту встречаемости паттерна в обучающей выборке.

3 Метод решения задачи выявления замкнутых паттернов

Разработанный метод состоит из двух этапов: на первом этапе генерируются кандидаты в замкнутые паттерны, на втором выполняется проверка, удовлетворяют ли кандидаты требованиям к замкнутым паттернам. В терминах табличных ограничений (см. раздел 1) обучающая выборка представляется как сжатая таблица *D*-типа, а каждый кандидат в замкнутые паттерны и искомый паттерн представляется, как сжатая таблица *C*-типа [*A*, *B*], которая состоит из одной строки, где $A \subseteq T$, $B \subseteq I$. *B* – это собственно паттерн, а *A* – его покрытие.

В примере 1 показано, как кортеж сжатой таблицы *D*-типа может быть соотнесён с каждым столбцом исходной бинарной матрицы, которая моделирует обучающую выборку. Для этого примера задача нахождения замкнутых паттернов может быть интерпретирована как нахождение значимых комбинаций одновременно приобретаемых товаров. Замкнутые паттерны обеспечивают минимальное представление всех паттернов, т.е можно получить все паттерны из замкнутых.

Схему вычислений, проводимых в рамках предложенного метода нахождения замкнутых паттернов, можно представить в следующем виде.

Первый этап. На этом этапе важно исключить как можно больше явно неперспективных случаев. Для этого необходимо: 1) представить обучающую выборку в виде сжатой таблицы *D*-типа; 2) преобразовать сжатую таблицу *D*-типа в эквивалентную сжатую таблицу *C*-типа с помощью методов ветвления и отсечения неперспективных ветвей дерева поиска. Каждый из полученных кортежей является кандидатом в паттерны.

Разработанный метод нахождения замкнутых паттернов включает систематическое исследование пространства поиска и построение дерева поиска следующим образом. Каждый уровень дерева поиска отображается в одну из строк сжатой таблицы *D*-типа, а узел отображается в определённую компоненту этой строки. Каждое решение задачи удовлетворения ограничений формируется путём выбора одной компоненты из каждой строки сжатой таблицы *D*-типа. Фрагмент дерева поиска для примера 1, построенного на основе сжатой таблицы *K*[*XY*], показан на рисунке 1.

Второй этап. Для каждого кандидата в замкнутые паттерны, описанного сжатым кортежем [*A*, *B*], выполняется следующая проверка.

1) для объектов *A* вычисляется $I_D(A)$. Это вычисление выполняется путём перемножения булевых векторов, соответствующих объектам из множества *A*.
2) $I_D(A)$ сравнивается с множеством *B*, в случае несовпадения - вывод отрицательный. Иначе проверка прошла успешно.

На рисунке 1 серым фоном обозначены кандидаты в паттерны, не прошедшие проверку.

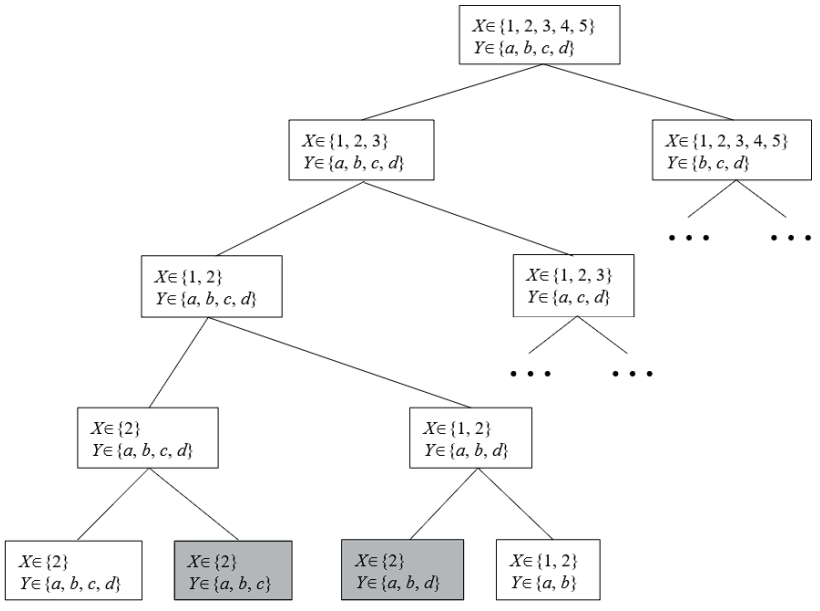

Рисунок 1 – Фрагмент дерева поиска для задачи нахождения замкнутых паттернов

Все замкнутые паттерны, найденные в примере 1 и прошедшие проверку, - это сжатые кортежи, которые можно объединить в одну сжатую таблицу *C*-типа:

$$\begin{bmatrix} \{1,2,3\} & \{a\} \\ \{1,2,4\} & \{b\} \\ \{2,4,5\} & \{c\} \\ \{1,2\} & \{a,b\} \\ \{2,4\} & \{b,c\} \\ \{2,3\} & \{a,d\} \\ \{2\} & \{a,b,c,d\} \end{bmatrix}.$$

Общая алгоритмическая сложность двух стадий предложенного метода может быть вычислена как: $O((|T||I|+2*|I|+2*|T|-min(|T|,|I|))*2^{min(|T|,|I|)})$. Более подробно описанные этапы изложены в [16].

Ускорение базового метода поиска. Известны три способа ускорения базового метода поиска: 1) использование отношений частичного порядка на множестве объектов и признаков; 2) сужение пространства поиска за счёт распространения ограничений; 3) использование дополнительных ограничений. Первый способ разобран в [16]. В данной работе рассматриваются 2-й и 3-й способы.

В статье реализуется идея, согласно которой распространение ограничений проводится на каждом шаге процедуры поиска с возвратами.

Утверждение 1 (**У1**). Если в сжатой таблице *D*-типа имеется пустой столбец (столбец, содержащий все пустые компоненты), то этот столбец удаляется из таблицы.

Утверждение 2 (**У2**). Если в сжатой таблице *D*-типа имеется пустая строка (строка, содержащая все пустые компоненты), то таблица пуста (задача удовлетворения ограничений не имеет решения).

Утверждение 3 (**У3**). Если в сжатой таблице *D*-типа имеется строка, содержащая только одну непустую компоненту, то элементы, не принадлежащие этой компоненте, удаляются из соответствующего домена.

Утверждение 4 (**У4**). Если в сжатой таблице *D*-типа имеется строка, содержащая хотя бы одну полную компоненту, то эта строка удаляется из сжатой таблицы *D*-типа.

Утверждение 5 (**У5**). Если компонента сжатой таблицы *D*-типа содержит элемент, не принадлежащий соответствующему домену, то этот элемент удаляется из компоненты.

Пример 2. Пусть имеется обучающая выборка (таблица 2). Эта бинарная матрица моделируется следующей сжатой таблицей *D*-типа:

$$\begin{array}{c} \\ \{1,2,3,4\} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{cc} X & Y \\ \{a,b,c,d\} & \\ \left[\{1,2\}\right. & \{b,c,d\} \\ \{1,2\} & \{a,c,d\} \\ \{3,4\} & \{a,b,d\} \\ \left.\{3,4\}\right] & \{a,b,c\} \end{array}.$$

Таблица 2 – Пример применения правил редукции

	a	*b*	*c*	*d*
1	1	1		
2	1	1		
3			1	1
4			1	1

Здесь в заголовках столбцов перечислены имена соответствующих переменных и доменов этих переменных.

Пусть на первом шаге поиска выбрана первая компонента первой строки (компонента {1, 2}), т.е. домен переменной *X* равен множеству {1, 2}. Тогда согласно утверждению **У4**,

первая и вторая строки удаляются из сжатой таблицы. Первые компоненты третьей и четвёртой строк становятся равны пустому множеству на основании утверждения **У5**. Тогда:

$$\begin{array}{cc} X & Y \\ \{1,2\} & \{a,b,c\} \\ 3 \left[\begin{array}{cc} \varnothing & \{a,b,d\} \end{array}\right. \\ 4 \left[\begin{array}{cc} \varnothing & \{a,b,c\} \end{array}\right. \end{array}.$$

Согласно утверждению **У3**, домен переменной Y сокращается до множества $\{a, b\}$, а оставшиеся строки удаляются на основании **У4**. Таким образом, все строки удаляются без формирования пустых строк, что означает нахождение решения. Получившееся решение описывается следующим сжатым кортежем в пространстве X, Y: $[\{1,2\}, \{a, b\}]$. Это решение является одним из замкнутых паттернов.

4 Ограничения на частоту встречаемости замкнутых паттернов

При нахождении частых паттернов сжатые таблицы сокращаются более активно по сравнения с нахождением просто замкнутых паттернов. Это происходит вследствие того, что к базовым правилам редукции **У1-У5** добавляются два дополнительных правила.

Утверждение 6 (**У6**). Компоненты сжатой таблицы D-типа, соответствующие переменной X и имеющие мощность меньше определённой границы θ, заменяются пустыми компонентами.

Утверждение 7 (**У7**). Если мощность домена переменной X меньше определённой границы θ, то задача удовлетворения ограничений не имеет решения.

Пример дерева поиска для задачи нахождения частых замкнутых паттернов показан на рисунке 2.

Условия задачи формулируются на основе исходных данных из примера 1, но налагается дополнительное ограничение $\theta = 2$ на частоту встречаемости паттерна. Во время вывода используются приведённые правила редукции. Данное дерево содержит существенно меньше узлов, нежели дерево на рисунке 1, которое не удаётся полностью поместить на рисунок.

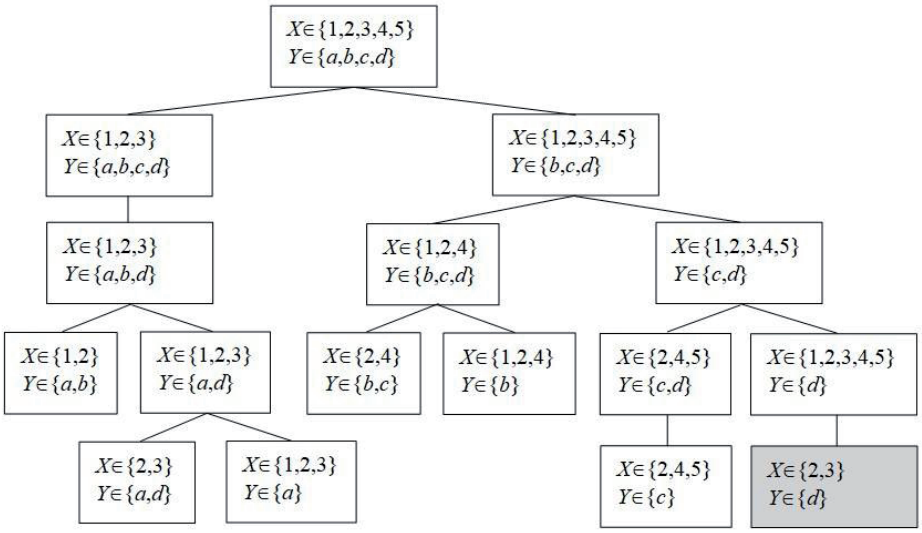

Рисунок 2 – Пример дерева поиска для задачи нахождения частых замкнутых паттернов

5 Ограничения на наличие/отсутствие элемента в замкнутом паттерне

5.1 Ограничение на наличие элемента в паттерне

Требуется найти все паттерны, содержащие некоторый заданный элемент y из домена переменной Y, т.е. требуется найти такие p, что $y \in p, y \in I$.

Пусть обучающая выборка представлена с помощью сжатой таблицы D-типа. В этом случае в процессе вывода, помимо базовых утверждений **У1-У5**, применяются следующие утверждения.

Утверждение 8 (**У8**). В кортеже, где в компоненте Y отсутствует значение y, которое должно содержаться в искомых паттернах, данная компонента заменяется на пустую.

Утверждение 9 (**У9**). Если в домене атрибута Y отсутствует значение y, которое должно содержаться в искомом паттерне, то задача удовлетворения ограничений не имеет решения.

Пусть в качестве обучающей выборки выступает бинарная матрица из примера 1, представленная в виде сжатой таблицы D-типа $K[XY]$. Необходимо, чтобы искомые паттерны содержали элемент "c". При этом ограничение на частоту встречаемости искомых паттернов имеет вид $freq_D(p) \geq 2$.

Третий кортеж исходной сжатой таблицы D-типа $K[XY]$ отличается от других кортежей таблицы тем, что его компонента Y не содержит значения "c". Эту компоненту следует заменить на пустую, поскольку при генерации искомых паттернов данная компонента никогда не может быть выбрана, а следовательно всегда выбирается компонента X рассматриваемого кортежа. Тогда текущие домены переменных станут: $X \in \{2, 4, 5\}$, $Y \in \{a, b, c, d\}$, а исходная таблица $K[XY]$ с учётом новых доменов преобразуется на основе утверждений **У1-У5** к следующему виду:

$$\begin{array}{c} X \quad\quad Y \\ \{2,4,5\} \quad \{a,b,c,d\} \\ \begin{array}{c} 1 \\ 2 \\ 4 \end{array} \left\lceil \begin{array}{cc} \{2\} & \{b,c,d\} \\ \{2,4\} & \{a,c,d\} \\ \{2\} & \{a,b,c\} \end{array} \right\rfloor \end{array}.$$

Из анализа требования к частоте встречаемости паттерна получается:

$$\begin{array}{c} X \quad\quad Y \\ \{2,4,5\} \quad \{a,b,c,d\} \\ \begin{array}{c} 1 \\ 2 \\ 4 \end{array} \left\lceil \begin{array}{cc} \varnothing & \{b,c,d\} \\ \{2,4\} & \{a,c,d\} \\ \varnothing & \{a,b,c\} \end{array} \right\rfloor \end{array}.$$

По сравнению с предыдущей сжатой таблицей D-типа в кортежах все компоненты X, мощность которых меньше заданного порога встречаемости ($\theta = 2$), заменяются пустыми компонентами (согласно **У6**). Тогда, применяя утверждения **У1-У5**, получаются следующие новые домены переменных: $X \in \{2, 4, 5\}$, $Y \in \{b, c\}$, а «остаток» сжатой таблицы D-типа будет иметь вид:

$$\begin{array}{c} X \quad\quad Y \\ \{2,4,5\} \quad \{b,c\} \\ \left]\{2,4\} \quad\quad \{c\}\right[\end{array}.$$

Дерево поиска для рассмотренного примера представлено на рисунке 3. Таким образом, в качестве решения имеются следующие паттерны: [{2,4} {*b*,*c*}] и [{2,4,5} {*c*}]. Видно, что они удовлетворяют как требованию на вхождение в паттерн элемента "*c*", так и требованию к частоте встречаемости искомых паттернов. Анализируя рисунок 3, можно сделать вывод, что других замкнутых паттернов с заданным набором ограничений не существует.

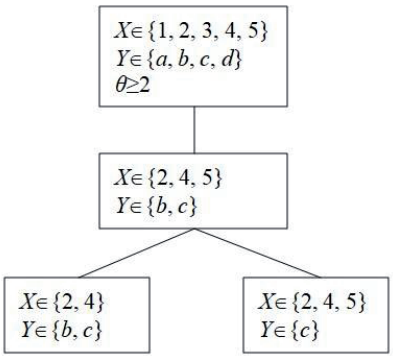

Рисунок 3 – Дерево поиска для задачи нахождения паттернов с заданным элементом (пример)

5.2 Ограничение на отсутствие элемента в паттерне

Пусть требуется найти все паттерны, не содержащие некоторый заданный элемент *y* из домена переменной *Y*, т.е. требуется найти такие *p*, что $y \notin p, y \in I$. В этом случае к базовым утверждениям **У1-У5** добавляются следующие утверждения.

Утверждение 10 (**У10**). В кортеже, где в компоненте *Y* отсутствует значение *y*, которое не должно содержаться в искомых паттернах, компонента *X* заменяется на пустую.

Утверждение 11 (**У11**). Если в сжатой таблице отсутствует кортеж, компонента *Y* которого не содержит значение *y*, то задача удовлетворения ограничений не имеет решения.

Пример 3. Пусть имеется обучающая выборка из примера 1 и требуется найти замкнутые паттерны, которые не должны содержать элемент "*a*" (т.е. $a \notin p$), при этом паттерны должны удовлетворять требованию $freq_D(p) \geq 2$.

В качестве исходной берётся сжатая таблица *D*-типа *K*[*XY*] из примера 1. Поскольку получаемые паттерны не должны содержать элемент "*a*", то всегда должна выбираться компонента *Y*, не содержащая данный элемент, – это компонента *Y* первого кортежа таблицы *K*[*XY*]. Это означает, что компоненту *X* первого кортежа таблицы *K*[*XY*], следует заменить на пустую компоненту. В результате, согласно **У1-У5**, получается:

$$\begin{array}{cc} X & Y \\ \{1,2,3,4,5\} & \{b,c,d\} \\ 2\left[\{1,2,4\}\right. & \left.\{c,d\}\right[\\ 3\left[\{2,4,5\}\right. & \left.\{b,d\}\right] \\ 4\left[\{2,3\}\right. & \left.\{b,c\}\right] \end{array}.$$

Дерево поиска представлено на рисунке 4. Пусть в полученной сжатой таблице выбирается компонента *X* первой строки. Тогда текущие домены переменных описываются следующим образом: $X \in \{1, 2, 4\}$, $Y \in \{b, c, d\}$. После «настройки» упрощённой сжатой таблицы на новый домен переменной *X*, получается:

$$\begin{array}{cc} X & Y \\ \{1,2,4\} & \{b,c,d\} \\ 3\left[\{2,4\}\right. & \left.\{b,d\}\right[\\ 4\left[\{2\}\right. & \left.\{b,c\}\right] \end{array}.$$

В процессе вывода используется тот факт, что искомый паттерн должен иметь порог абсолютной частоты встречаемости θ=2. Следовательно, к полученной сжатой таблице

можно применить утверждение **У6**. Тогда новые домены переменных: $X \in \{1, 2, 4\}$, $Y \in \{b, c\}$. Применяя **У1-У5**, получается сжатая таблица *D*-типа, состоящая из одной строки:

$$\begin{array}{cc} X & Y \\ \{1,2,4\} & \{b,c\} \\ ]\{2,4\} & \{b\}[\end{array}.$$

В результате прохода по данной ветке листовые вершины дерева поиска получаются путём выбора одной из компонент сжатой таблицы (см. рисунок 4). Всего в данном дереве поиска получаются три паттерна с заданной частотой встречаемости и не содержащие элемент "*a*": $[\{2,4\} \quad \{b,c\}]$, $[\{1,2,4\} \quad \{b\}]$, $[\{2,4,5\} \quad \{c\}]$.

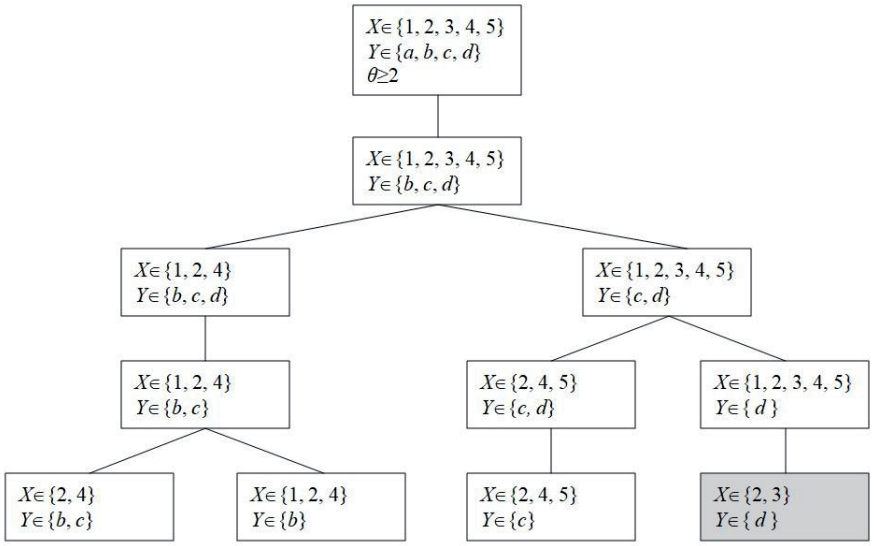

Рисунок 4 – Дерево поиска для задачи нахождения паттернов, не содержащих заданный элемент (пример)

Заключение

В работе представлен оригинальный подход для решения задач, связанных с нахождением замкнутых паттернов при наличии дополнительных требований к их виду. Рассмотренные задачи решаются как задачи удовлетворения табличных ограничений с применением авторской процедуры систематического поиска и методов распространения ограничений.

Новизной исследований является то, что обучающую выборку предложено представлять в виде специального типа табличных ограничений - сжатых таблиц *D*-типа. Такое представление информации и предложенный способ построения дерева поиска позволяют ускорить процесс вычислений: для некоторых типов входных данных приведённая оценка вычислительной сложности авторского метода нахождения замкнутых паттернов оказывается лучше, чем оценка большинства методов-прототипов [16]. Дополнительное сокращение количества узлов дерева поиска достигается за счёт применения разработанных процедур распространения табличных ограничений, основанных на эквивалентных преобразованиях сжатых таблиц *D*-типа с использованием специализированных правил редукции.

Предлагаемый подход позволяет легко адаптироваться к дополнительным требованиям, накладываемым на паттерны. Для этого к основным правилам редукции сжатых таблиц *D*-типа, используемым при поиске замкнутых паттернов, добавляются правила, учитывающие интересующие особенности анализируемых закономерностей.

Список источников

[1] ***Russel S., Norvig P.*** Artificial Intelligence: A Modern Approach. 3rd ed. Prentice Hall, 2010. 1132 p.

[2] ***Bisaria J., Shivastava N., Pardasani K.R.*** A rough set model for constraint driven mining of sequentioal patterns. Int. J. of Computer and Network Sequrity. 2009; 1(1).

[3] ***Gan W., Lin J.C.W., Fournier-Viger P., Chao H.C., Tseng V.S., Yu P.S.*** A survey of utility-oriented pattern mining. Transactions on Knowledge and Data Engineering. 2021. Vol. 33(4). P.1306–1327. DOI: 10.1109/TKDE.2019.2942594.

[4] ***Hien A., Loudni S., Aribi N., Lebbah Y., Laghzaoui M., Ouali A., Zimmermann A.*** A Relaxation-Based Approach for Mining Diverse Closed Patterns. In: book Machine Learning and Knowledge Discovery in Databases. February 2021. P.36-54. DOI:10.1007/978-3-030-67658-2_3.

[5] ***Belaid M., Bessi`ere C., Lazaar N.***: Constraint Programming for Association Rules. In: Proc. Of the Int. Conf. on Data Mining. May 2019. P.127–135. DOI:10.1137/1.9781611975673.15.

[6] ***Jabbour S., Mazouri F., Sais L.*** Mining Negatives Association Rules Using Constraints. Procedia Computer Science. 2018. Vol.127. P.481–488. DOI: 10.1016/j.procs.2018.01.146.

[7] ***Zuenko A.*** Representation and processing of qualitative constraints using a new type of smart tables. In: Proc. of the 4th Int. Conf. on Computer Science and Application Engineering. October 2020. P.1–7. DOI:10.1145/3424978.3425023.

[8] ***Mackworth A.*** Consistency in networks of relations // Artificial Intelligence. 8(1), 1977. P.99–118.

[9] ***Кузнецов С.О.*** Автоматическое обучение на основе формальных понятий // Автоматика и телемеханика, 2001. № 12, С.3–27. DOI: 10.1023/A:1012435612567.

[10] ***Yap R., Wang W.*** Generalized Arc Consistency algorithms for table constraints: a summary of algorithmic ideas. In: Proc. of the AAAI Conf. of Artificial Intelligence. April 2020. P.13590–13597. DOI:10.1609/aaai.v34i09.7086.

[11] ***Ingmar L., Schulte C.*** Making compact-table compact. In: Proc. of the Int. Conf. of Principle and Practice of Constraint Programming. Cham. 2015 August 23. P.271–287. DOI: 10.1007/978-3-319-98334-9_14.

[12] ***Mairy J.-B., Deville Y., Lecoutre C.*** The smart table constraint. In: Proc. of the Int. Conf. of Integration of Constraint Programming, Artificial Intelligence, and Operations Research. Cham. 2015 April 16. P.271-287. DOI:10.1007/978-3-319-18008-3_19.

[13] ***Bennai S., Amroun K., Loudni S.*** Exploiting Data Mining Techniques for Compressing Table Constraints. In: Proc. of the 31st Int. Conf. on Tools with Artificial Intelligence. November 2019. P.42–49. DOI:10.1109/ICTAI.2019.00015.

[14] ***Кулик Б.А., Зуенко А.А., Фридман А.Я.*** Алгебраический подход к интеллектуальной обработке данных и знаний. Санкт-Петербург: Изд-во Политехнического ун-та, 2010. 235 с.

[15] ***Agrawal R., Imieli'nsky T., Swami A.*** Mining association rules between sets of items in large databases. In: Proc. of the 1993 ACM SIGMOD Int. Conf. on Management of Data. Washington, United States, January 1993.

[16] ***Зуенко А.А.*** Метод машинного обучения для выявления замкнутых множеств общих признаков объектов с применением технологии программирования в ограничениях // Автоматика и телемеханика, 2022. №12. С.156-168. DOI: 10.31857/S000523102212011X.

Сведения об авторах

Зуенко Александр Анатольевич, 1983 г.р., к.т.н., ведущий научный сотрудник Института информатики и математического моделирования – обособленного подразделения ФИЦ «Кольский научный центр Российской академии наук». Области научных интересов: программирование в ограничениях; моделирование слабо формализованных предметных областей. ORCID: 0000-0001-5431-7538; Author ID (RSCI): 528493; Author ID (Scopus): 26536974000; Researcher ID (WoS): E-7944-2017. *zuenko@iimm.ru*. ✉

Зуенко Ольга Николаевна, 1980 г.р., младший научный сотрудник Института информатики и математического моделирования – обособленного подразделения ФИЦ «Кольский научный центр Российской академии наук». Область научных интересов - машинное обучение. ORCID: 0000-0002-7165-6651; Author ID (RSA): 1069604; Author ID (Scopus): 57222359556; Researcher ID (WoS): HKN-6360-2023. *ozuenko@iimm.ru*.

Поступила в редакцию 01.07.2023, после рецензирования 25.08.2023. Принята к публикации 29.08.2023.

Scientific article DOI: 10.18287/2223-9537-2023-13-3-392-404

Finding dependencies in data based on methods of satisfying table constraints

© 2023, A.A. Zuenko✉, O.N. Zuenko

Putilov Institute for Informatics and Mathematical Modeling, Apatity, Russia
Subdivision of the Federal Research Centre «Kola Science Centre of the Russian Academy of Sciences»,

Abstract

The work deals with the search for a special type of regularities in data, called frequent patterns. A frequent pattern is understood as a certain set of attributes that characterizes a sufficiently large number of objects of the training sample. There are many methods for pattern discovery, but they usually do not allow flexible consideration of necessary requirements for their type. Taking into account the new conditions that the desired patterns must meet leads in practice to a time-consuming modification of used algorithms and a decrease in computing performance. This article proposes a new approach based on the constraint programming paradigm, which is free from the listed disadvantages. The approach is based on the original way of presenting the training sample using specialized table constraints – compressed D-type tables, on the author's method of backtracking, as well as on specialized reduction rules for table constraints. Particular attention is paid to solving the closed patterns discovery problem, which is included as part of the solution of all machine learning problems considered in the work, which means taking into account additional requirements for the type of patterns. As additional requirements to the type of pattern, constraints on the frequency of occurrence of a closed pattern, as well as conditions for the occurrence of some element (attribute) into the pattern, are considered. To the basic rules for the reduction of compressed D-type tables, rules are added that take into account the interesting attributes of the analyzed patterns. The advantage of the approach is that the taking into account and analyzing new constraints makes it possible to speed up the calculation process.

Key words: *constraint programming, pattern extraction, reduction rules, machine learning, table constraints.*

For citation: *Zuenko AA, Zuenko ON.* Finding dependencies in data basied on methods of satisfying table constraints [In Russian]. *Ontology of designing*. 2023; 13(3): 392-404. DOI: 10.18287/2223-9537-2023-13-3-392-404.

Acknowledgment: The work was carried out within the framework of the current research topic «Development of theoretical and organizational and technical foundations of information support for managing the viability of regional critical infrastructures of the Arctic zone of the Russian Federation» (registration number 122022800547-3).

Conflict of interest: The authors declare no conflict of interest.

List of figures and tables

References

[1] ***Russel S, Norvig P***. Artificial Intelligence: A Modern Approach. 3rd ed. Prentice Hall, 2010. 1132 p.
[2] ***Bisaria J, Shivastava N, Pardasani KR.*** A rough set model for constraint driven mining of sequentioal patterns. Int. J. pf Computer and Network Sequrity. 2009; 1(1).
[3] ***Gan W, Lin JCW, Fournier-Viger P, Chao HC, Tseng VS, Yu PS.*** A survey of utility-oriented pattern mining. Transactions on Knowledge and Data Engineering. 2021; 33(4): 1306–1327. DOI: 10.1109/TKDE.2019.2942594.

[4] ***Hien A, Loudni S, Aribi N, Lebbah Y, Laghzaoui M, Ouali A, Zimmermann A.*** A Relaxation-Based Approach for Mining Diverse Closed Patterns. In: book Machine Learning and Knowledge Discovery in Databases. 2021 February. P.36-54. DOI:10.1007/978-3-030-67658-2_3.

[5] ***Belaid M, Bessi`ere C, Lazaar N.*** Constraint Programming for Association Rules. In: Proc. Of the Int. Conf. on Data Mining. 2019 May. P.127–135. DOI:10.1137/1.9781611975673.15.

[6] ***Jabbour S, Mazouri F, Sais L.*** Mining Negatives Association Rules Using Constraints. Procedia Computer Science. 2018; 127: 481–488. DOI: 10.1016/j.procs.2018.01.146.

[7] ***Zuenko A.*** Representation and processing of qualitative constraints using a new type of smart tables. In: Proc. of the 4th Int. Conf. on Computer Science and Application Engineering. October 2020. P.1–7. DOI:10.1145/3424978.3425023.

[8] ***Mackworth A.*** Consistency in networks of relations // Artificial Intelligence. 8(1), 1977. P.99–118.

[9] ***Kuznetsov S.*** Automatic learning based on the Analysis of Formal Concepts [In Russian]. Automation and Telemechanics, 2001; 12: 3–27. DOI: 10.1023/A:1012435612567.

[10] ***Yap R, Wang W.*** Generalized Arc Consistency algorithms for table constraints: a summary of algorithmic ideas. In: Proc. of the AAAI Conf. of Artificial Intelligence. April 2020. P.13590–13597. DOI:10.1609/aaai.v34i09.7086.

[11] ***Ingmar L, Schulte C.*** Making compact-table compact. In: Proc. of the Int. Conf. of Principle and Practice of Constraint Programming. Cham, 2015 August 23. P.271–287. DOI: 10.1007/978-3-319-98334-9_14.

[12] ***Mairy J-B, Deville Y, Lecoutre C.*** The smart table constraint. In: Proc. of the Int. Conf. of Integration of Constraint Programming, Artificial Intelligence, and Operations Research. Cham, 2015 April 16. P.271-287. DOI:10.1007/978-3-319-18008-3_19.

[13] ***Bennai S, Amroun K, Loudni S.*** Exploiting Data Mining Techniques for Compressing Table Constraints. In: Proc. of the 31st Int. Conf. on Tools with Artificial Intelligence. 2019 November, P.42–49. DOI:10.1109/ICTAI.2019.00015.

[14] ***Kulik BA, Zuenko AA, Fridman AYa.***. Algebraic approach to the intellectual processing of data and knowledge [In Russian] Saint-Petersburg: Izd-vo Politekhn. un-ta; 2010. 235 p.

[15] ***Agrawal R, Imieli'nsky T, Swami A.*** Mining association rules between sets of items in large databases. In: Proc. of the 1993 ACM SIGMOD Int. Conf. on Management of Data. Washington, United States, January 1993.

[16] ***Zuenko AA.*** A machine learning method to reveal closed sets of common feature of objects using constraint programming [In Russian]. *Automation and Telemechanics*. 2022; 12: 156-168. DOI: 10.31857/S000523102212011X.

About the authors

Alexander Anatolyevich Zuenko (b. 1983). Graduated from the Petrozavodsk State University (Apatity, Russia) in 2005, PhD (2009), a senior researcher at the Institute for Informatics and Mathematical Modeling, a Subdivision of the Federal Research Centre «Kola Science Centre of the Russian Academy of Sciences (IIMM KSC RAS). He is a member of Russian Association of Artificial Intelligence. He is a co-author of more than 150 publications and monographies in the field of constraint programming and modeling in poorly formalized subject domains. ORCID: 0000-0001-5431-7538; Author ID (RSA): 528493; Author ID (Scopus): 26536974000; Researcher ID (WoS): E-7944-2017. *zuenko@iimm.ru* ✉.

Olga Nikolaevna Zuenko (b. 1980). Graduated from the Petrozavodsk State University (Apatity, Russia) in 2002, a junior researcher at the Institute for Informatics and Mathematical Modeling, a Subdivision of the Federal Research Centre «Kola Science Centre of the Russian Academy of Sciences (IIMM KSC RAS). She is a co-author of 7 publications in the field of machine learning. ORCID: 0000-0002-7165-6651; Author ID (RSA): 1069604; Author ID (Scopus): 57222359556; Researcher ID (WoS): HKN-6360-2023. *ozuenko@iimm.ru.*

Received July 2, 2023. Revised August 25, 2023. Accepted August 29, 2023.