# Поиск частых элементов произведения частичных порядков и ассоциативные правила

**И.Е. Генрихов[1], Е.В. Дюкова[2]**

[1]ООО «Мобайл парк ИТ», Панфилова 21/1, Химки, Россия, 141407
[2]ФИЦ «Информатика и управление» РАН, Вавилова 44/2, Москва, Россия, 119333

**Аннотация.** Один из известных способов нахождения частых наборов атрибутов при поиске ассоциативных правил в бинарной базе данных основан на построении FP-дерева (Frequent Pattern Growth Tree). В случае небинарных данных, как правило, осуществляется бинаризация значений каждого небинарного атрибута по некоторому числовому порогу и задача сводится к построению классического FP-дерева. Результат существенно зависит от выбора указанных порогов. В работе предлагается модифицировать конструкцию классического FP-дерева путём введения для каждого небинарного атрибута дополнительной вершины, названной полной и содержащей информацию о возможных вариантах бинаризации значений этого атрибута. Новая модель FP-дерева названа полным FP-деревом. Рассматриваются вопросы применения полного FP-дерева для анализа небинарных данных при условии, что на значениях атрибутов заданы частичные порядки. Приводятся иллюстративные модельные примеры.

## 1. Введение

Задача поиска ассоциативных правил в данных является одной из центральных задач интеллектуального анализа информации и актуальна для многих прикладных областей. Эта задача впервые поставлена в [1] и первоначально формулировалась как задача анализа потребительской корзины. Приведём её стандартную постановку в случае бинарных данных.

Дано некоторое множество $P$, элементы которого называются атрибутами. Дана база данных $D$, содержащая некоторые подмножества множества $P$, не обязательно различные. Подмножества множества $P$ называются наборами атрибутов, а те из них, которые содержатся в $D$, называются транзакциями. *Ассоциативное правило* это пара непересекающихся наборов атрибутов $X$ и $Y$, которые одновременно содержатся минимум в одной транзакции. Ассоциативное правило, порождаемое $X$ и $Y$, обычно обозначается через $X \Rightarrow Y$. *Поддержкой* (*support*) правила $X \Rightarrow Y$ называется отношение числа транзакций, содержащих $X \cup Y$, к числу всех транзакций. *Достоверностью* (*confidence*) правила $X \Rightarrow Y$ называется отношение числа транзакций, содержащих $X \cup Y$, к числу транзакций, содержащих $X$. Требуется найти ассоциативные правила с поддержкой не менее $s$, $s \in [0,1]$, и с достоверностью не менее $c$, $c \in [0,1]$.

Поиск ассоциативных правил обычно осуществляется в два этапа. Сначала находятся все так называемые $s$-частые наборы атрибутов. Набор атрибутов $Z$ называется $s$-частым, если отношение числа транзакций, содержащих $Z$, к числу всех транзакций не менее $s$. Затем для

каждого найденного $s$-частого набора $Z$ путем разбиения $Z$ на два непересекающихся подмножества $X$ и $Y$ строятся ассоциативные правила вида $X \Rightarrow Y$ с достоверностью не менее $c$.

В более общей постановке каждый атрибут имеет некоторое множество числовых значений и вместо наборов атрибутов рассматриваются наборы их значений. Как правило, поиск ассоциативных правил сводится к наиболее простому бинарному случаю путем задания для каждого небинарного атрибута некоторого числа (порога), позволяющего перекодировать исходные данные в бинарные [2, 3]. Результат существенно зависит от выбора варианта бинаризации. Однако перебор по всем возможным вариантам бинаризации данных требует больших временных затрат.

В настоящей работе рассмотрены вопросы нахождения частых наборов значений атрибутов и ассоциативных правил при условии, что на значениях атрибутов заданы частичные порядки. База данных представлена в виде некоторой совокупности элементов множества $P = P_1 \times \ldots \times P_n$, где $P_1, \ldots, P_n$ – конечные частично упорядоченные числовые множества (далее атрибуты), и элемент $y = (y_1, \ldots, y_n) \in P$ следует за элементом $x = (x_1, \ldots, x_n) \in P$ ($x \circ y$), если $y_i$ следует за $x_i$ при $i = 1, 2, \ldots, n$. В случае бинарных данных множество $P$ – это $n$-мерный булев куб, в котором установлен стандартный частичный порядок.

Предложена схема бинаризации исходных данных согласно которой для каждого атрибута $P_i$, $i \in \{1, 2, \ldots, n\}$, строится множество «значимых порогов» $Q_i \phi P_i$. Набор порогов $\{p_1, \ldots, p_n\}$, в котором $p_i \in Q_i$ при $i = 1, 2, \ldots, n$, порождает один из возможных вариантов бинарной перекодировки элементов множества $P$. Ассоциативные правила, найденные с использованием данной схемы бинаризации, названы пороговыми. Вводится понятие оптимального значимого набора порогов, позволяющего находить наибольшее число пороговых ассоциативных правил.

Для эффективного перечисления пороговых ассоциативных правил, порождаемых всеми возможными вариантами бинарной перекодировки, предложена модель полного (full) FP-дерева (FFP-дерева). Конструкция FFP-дерева является модификацией классической конструкции FP-дерева [4, 5]. В FFP-дереве для каждого атрибута $P_i$, не являющегося бинарным, строится дополнительная вершина, названная полной. Этой вершине ставится в соответствие множество значимых порогов $Q_i$. Фактически каждый порог из $Q_i$ порождает поддерево FFP-дерева, которое является бинарным FP-деревом. Обоснование эффективности предложенного подхода к построению пороговых ассоциативных правил проведено на модельных примерах.

Представленные в работе результаты частично анонсированы в [6].

## 2. Основные понятия

Пусть $P = P_1 \times \ldots \times P_n$, где $P_1, \ldots, P_n$ – конечные частично упорядоченные числовые множества и элемент $y = (y_1, \ldots, y_n) \in P$ следует за элементом $x = (x_1, \ldots, x_n) \in P$ ($x \circ y$), если $y_i$ следует за $x_i$ при $i = 1, 2, \ldots, n$. В случае $x \circ y$, $x \neq y$ пишут $x \prec y$.

Элементы $x, y \in P$ называются *сравнимыми*, если либо $x \circ y$, либо $y \circ x$. В противном случае $x$ и $y$ называются *несравнимыми*.

Предполагается, что каждое множество $P_i$ имеет *наименьший элемент*, т.е. такой элемент $l_i$, для которого выполнено $l_i \circ x_i$ для любого $x_i \in P_i$. Элемент $x_i \in P_i$ называется *значением* элемента $x = (x_1, \ldots, x_i, \ldots, x_n) \in P$ и называется *существенным значением*, если $x_i \neq l_i$.

Предполагается также, что база данных $D$ представлена в виде некоторой совокупности элементов множества $P$ и не содержит транзакцию $l = (l_1, \ldots, l_n)$.

В случае бинарных данных считается, что $P_i = \{0, 1\}$, $i = 1, 2, \ldots, n$, и в каждом $P_i$ установлен порядок $0 \prec 1$. Здесь $l_i = 0$, каждое $P_i$ имеет всего один существенный элемент, равный 1, и база данных не содержит транзакцию (0,…,0).

Пусть $x \in P$. Число транзакций $y$ в $D$ таких, что $x \circ y$ обозначается через $S_D(x)$. Величина $S_D(x) / |D|$, где $|D|$ – число всех транзакций в $D$, называется *поддержкой* элемента $x$ в базе данных $D$. Элемент $x$ называется $s$-*частым*, если его поддержка не менее $s$. Элемент $x$ называется максимальным $s$-частым элементом, если для любого другого $s$-частого элемента $y \in P$ не верно $x \prec y$.

Пара несравнимых элементов $x = (x_1, \ldots, x_n)$ и $y = (y_1, \ldots, y_n)$ множества $P$ называется *непересекающейся*, если для любого $i \in \{1, 2, \ldots, n\}$ хотя бы один из элементов $x_i$ и $y_i$ равен $l_i$, т.е. хотя бы один из этих элементов не является существенным. Из определения следует, что если $x$ и $y$ – непересекающиеся элементы множества $P$, то $x \neq l$, $y \neq l$.

Если данные бинарные, то согласно сказанному выше пара элементов $x$ и $y$ множества $P$ называется *непересекающейся*, если $x \neq (0, \ldots, 0)$, $y \neq (0, \ldots, 0)$ и для каждого $i \in \{1, 2, \ldots, n\}$ выполнено одно из условий: 1) $x_i = y_i = 0$; 2) $x_i = 0$, $y_i = 1$; 3) $x_i = 1$, $y_i = 0$.

Пусть $x, y \in P$, $x = (x_1, \ldots, x_n)$, $y = (y_1, \ldots, y_n)$, – непересекающаяся пара элементы. Через $x \sqcup y = (u_1, \ldots, u_n)$ обозначается элемент множества $P$, в котором $u_i = l_i$, если $x_i = y_i = l_i$, иначе либо $u_i = x_i$, если $x_i$ – существенное значение для $x$, либо $u_i = y_i$, если $y_i$ – существенное значение для $y$.

*Ассоциативным* $(s, c)$*-правилом,* $s \in [0,1]$, $c \in [0,1]$, называется пара непересекающихся элементов $x$ и $y$ множества $P$, таких что $x \sqcup y$ – $s$-частый элемент и $S_D(x \sqcup y) / S_D(x) \geq c$.

Ассоциативное $(s, c)$-правило $x \Rightarrow y$ указывает на определённую зависимость между набором существенных значений элемента $x$ и набором существенных значений элемента $y$.

Рассмотрим случай небинарных данных. Задачу поиска ассоциативных правил сведём к бинарному случаю путем задания для каждого $P_i$ некоторого числа (порога) $p_i$, $p_i \in P_i$, $p_i \neq l_i$. В элементе $x = (x_1, \ldots, x_n) \in P$ число $x_i$ заменим на 1, если $p_i \circ x_i$ и заменим на 0 в противном случае. В результате такой кодировки для каждого заданного набора порогов $H = (p_1, \ldots, p_n)$ получаем вместо множества $P$ множество $P_H = P_1^H \times \ldots \times P_n^H$, где $P_i^H = \{0, 1\}$, $i = 1, 2, \ldots, n$, и в каждом $P_i^H$ установлен порядок $0 \prec 1$. Вместо базы данных $D$ получаем базу данных $D_H$. Ассоциативные правила, найденные по базе данных $D_H$, назовём *пороговыми*. Порогу $p_i$, $p_i \in P_i$, поставим в соответствие элемент $\varphi(p_i) = (x_1, \ldots, x_n) \in P$, в котором $x_i = p_i$ и $x_j = l_i$ при $j \neq i$. Тогда порог $p_i$ называется *значимым*, если $S_D(\varphi(p_i)) \geq s$, т.е. элемент $\varphi(p_i)$ является $s$-частым в $P$. Предполагается, что каждое $P_i$ имеет хотя бы один значимый порог. Набор порогов $H = (p_1, \ldots, p_n)$ называется *значимым*, если для любого $i \in \{1, 2, \ldots, n\}$ порог $p_i$ – значимый. Для бинарной перекодировки будем выбирать только значимые наборы порогов. Набор порогов $\{p_1, \ldots, p_n\}$, в котором для каждого $p_i$, $i \in \{1, 2, \ldots, n\}$, выполняется условие $S_D(\varphi(p_i)) = \max_{p \in P_i} S_D(\varphi(p))$, называется о*птимальным значимым.* Оптимальный значимый набор порогов позволяет находить наибольшее число частых элементов по сравнению с другими значимыми наборами порогов и в результате строить наибольшее число пороговых ассоциативных правил.

Ставится задача нахождения всех таких максимальных частых элементов и таких пороговых ассоциативных правил, которые порождаются значимыми наборами порогов. Рассматриваются два способа решения задачи. Первый способ основан на последовательном нахождении значимых наборов порогов и построении для каждого такого набора классического бинарного FP-дерева. Второй способ основан на построении FFP-дерева. Результаты экспериментального сравнения по скорости счёта двух указанных подходов к перечислению искомых ассоциативных правил приводятся в разделе 5. Кроме того, в разделе 5 приведены результаты экспериментов по сравнению числа максимальных частых элементов и числа пороговых ассоциативных правил, порождаемых оптимальным значимым набором порогов, соответственно с числом максимальных частых элементов и числом пороговых ассоциативных правил, порождаемых всеми значимыми наборами порогов.

*Замечание. Введённые в данном разделе понятия являются оригинальными, за исключением понятия $s$-частого элемента произведения частичных порядков, которое ранее было дано в [7] в связи с задачей поиска в данных, представленных в виде произведения частичных порядков, ассоциативных правил специального вида, названных в [7] неприводимыми. Определение неприводимого ассоциативного правила использует исключительно понятия максимального $s$-частого элемента и минимального $s$-нечастого элемента произведения частичных порядков.*

**3. FP-дерево и поиск частых элементов произведения частичных бинарных порядков**

Дадим описание классического FP-дерева в предположении, что данные представлены в виде произведения частичных бинарных порядков, а именно $P = P_1 \times \ldots \times P_n$, где $P_i = \{0,1\}$, $i \in \{1,2,\ldots,n\}$, и в каждом множестве (атрибуте) $P_i$ установлен порядок $0 \prec 1$. Фактически FP-дерево это структурированное представление исходной базы данных.

Отметим, что в классическом варианте FP-дерево используется для перечисления всех $s$-частых элементов множества $P$. В силу большого числа $s$-частых элементов имеет смысл искать и хранить только те из них, которые являются максимальными.

Введем обозначения: $D(P_i)$ – база, полученная удалением атрибута $P_i$ из базы $D$ и удалением транзакций из $D$, в которых значение атрибута $P_i$ равно 1; $S_D(P_i)$ – число транзакций в $D$, в которых значение атрибута $P_i$ равно 1; $\max(D) = \max(S_D(P_i))$, $i = 1,2,\ldots,n$.

Для ветвления из корневой вершины дерева строится набор атрибутов $A_D = \{P_{j_1},\ldots,P_{j_t}\}$ такой, что $S_D(P_{j_1}) = \max(D)$. Процедура выбора остальных атрибутов набора $A_D$ заключается в следующем.

Положим $D_1 = D$, $D_2 = D_1(P_{j_1})$. Тогда $P_{j_2}$ – атрибут, для которого $S_{D_2}(P_{j_2}) = \max(D_2) \neq 0$. Далее строится база $D_3 = D_2(P_{j_1})$ и находится атрибут $P_{j_3}$, для которого $S_{D_3}(P_{j_3}) = \max(D_3) \neq 0$. Процесс построения набора $A_D$ продолжается до тех пор, пока указанный выбор атрибутов возможен.

Из корневой вершины дерева выходит дерева $t$ ребер. Ребро с меткой $j_i$, $i \in 1,2,\ldots,t$, входит в вершину $(P_{j_i}, S_{D_i}(P_{j_i}))$. Таким образом, каждая вершина первого яруса дерева содержит некоторый атрибут из $A_D$. Вершине $(P_{j_i}, S_{D_i}(P_{j_i}))$ приписывается текущая база данных $D'$, полученная из $D_i$ удалением атрибута $P_{j_i}$ и удалением транзакций, в которых значение атрибута $P_{j_i}$ равно 0. Ветвление из внутренней вершины дерева с текущей базой $D'$ происходит аналогичным образом (база $D$ заменяется на базу $D'$). Процесс синтеза дерева

продолжается до тех пор, пока текущая база данных содержит не менее одного атрибута и не менее одной транзакции.

Для формирования $s$-частых элементов, в которых значение атрибута $P_i$ равно 1, в построенном FP-дереве выделяют ветви дерева, в которых встречается вершина вида $(P_i, S_{D'}(P_i))$. Каждая такая ветвь порождает определённый набор атрибутов $A$. Атрибут $P_j$ принадлежит $A$, если либо $P_j = P_i$, либо вершина $(P_i, S_{D'}(P_i))$ является потомком вершины, содержащей $P_j$. Набору $A$ приписывается число $S(A) = S_{D'}(P_i)$. Вершина $(P_i, S_{D'}(P_i))$ называется *крайней* для набора атрибутов $A$.

Пусть $R(P_i) = \{A_1, \ldots, A_q\}$ – множество таких наборов атрибутов, для каждого из которых вершина $(P_i, S_{D'}(P_i))$ является крайней. Для каждого атрибута $Q$ из $\bigcup_{j=1}^{q} A_j$ вычисляется значение величины $T(Q) = \sum_{j=1,\ldots,q \mid Q \in A_j} S(A_j)$. Если $T(Q)/|D| < s$, то в каждом наборе $A$ из $R$ удаляется атрибут $Q$. Получается новое множество наборов атрибутов $R(P_i)$. Далее наборы атрибутов в множестве $R(P_i)$ сортируются в порядке уменьшения их мощности, чтобы избежать ситуации когда будет найден $s$-частый элемент, который следует за уже найденным. После этого для каждого $A$ из $R(P_i)$ строятся те максимальные $s$-частые элементы множества $P$, в которых значение каждого атрибута из $A$ равно 1. Затем для каждого найденного максимального $s$-частого элемента путем применения некоторой рекурсивной процедуры получаются ассоциативные правила с требуемой достоверностью [5, 8].

Рассмотрим пример синтеза FP-дерева для бинарной базы данных $D$, представленной в таблице 1.

**Таблица 1.** Пример бинарной базы данных.

| № | $P_1$ | $P_2$ | $P_3$ | $P_4$ |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 |
| 2 | 1 | 1 | 0 | 0 |
| 3 | 0 | 1 | 1 | 1 |
| 4 | 1 | 1 | 1 | 1 |
| 5 | 1 | 1 | 0 | 0 |
| 6 | 1 | 0 | 1 | 1 |
| 7 | 0 | 1 | 1 | 1 |

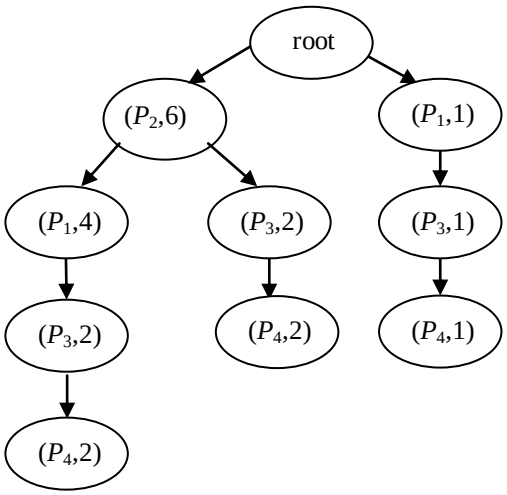


**Рисунок 1.** FP-дерево для базы данных, представленной в таблице 1.

В данном случае в качестве $A_D$ может выступать каждый из трёх наборов: $\{P_2, P_1\}$, $\{P_2, P_3\}$, $\{P_2, P_4\}$. На рисунке 1 изображено дерево, построенное при условии $A_D = \{P_2, P_1\}$. Из корневой вершины дерева выходит ровно два ребра. Первое ребро входит во внутреннюю вершину с атрибутом $P_2$, так как $S_D(P_2) = \max(D)$. Второе ребро входит во внутреннюю вершину с атрибутом $P_1$. В каждой вершине дерева указан некоторый атрибут $Q \in \{P_1, P_2, P_3, P_4\}$ и после

запятой приведено значение величины $S_{D'}(Q)$. Если сложить значения величин $S_{D'}(Q)$ для всех внутренних вершин, содержащих $Q$, то получим $S_D(Q)$.

Пусть $s = 0,4$. Используя построенное FP-дерево, найдём все максимальные $s$-частые элементы, в которых значение атрибута $P_4$ равно 1. Нетрудно видеть, что имеется всего три ветви, в которых вершина, содержащая $P_4$, является крайней. Эти ветви порождают наборы атрибутов $A_1 = (P_2, P_1, P_3, P_4)$, $A_2 = (P_2, P_3, P_4)$, $A_3 = (P_1, P_3, P_4)$. Далее, определив $S(A_1) = 2$, $S(A_2) = 2$, $S(A_3) = 1$, получим $T(P_i)/|D| > s$, $i = 1,2,3,4$. Следовательно, $R(P_4) = R(P_4)$. Рассмотрим набор $A_1$. Поскольку $S(A_1)/|D| < s$, то разобьём $A_1$ на тройки атрибутов, содержащие атрибут $P_4$. Получим $A_{11} = (P_2, P_3, P_4)$, $A_{12} = (P_2, P_1, P_4)$, $A_{13} = (P_1, P_3, P_4)$. Вычислим по FP-дереву $S(A_{11}) = 4$, $S(A_{12}) = 2$, $S(A_{13}) = 3$. Получим, что наборы $A_{11}$ и $A_{13}$ порождают $s$-частые элементы $(0,1,1,1)$ и $(1,0,1,1)$, а набор $A_{12}$ не порождает ни одного такого элемента. Набор $A_{12}$ разобьём на пары атрибутов, содержащие $P_4$. Получим $A_{121} = (P_2, P_4)$, $A_{122} = (P_1, P_4)$. Наборы $A_{121}$ и $A_{122}$ порождают частые $s$-элементы $(0,1,0,1)$ и $(1,0,0,1)$, при этом $(0,1,0,1) \prec (0,1,1,1)$ и $(1,0,0,1) \prec (1,0,1,1)$. Тем самым набор $A_1$ порождает два максимальных $s$-частых элемента для атрибута $P_4$, а именно, $(0,1,1,1)$ и $(1,0,1,1)$. Рассматривать наборы $A_2$ и $A_3$ не имеет смысла, так как $A_2 = A_{11}$ и $A_3 = A_{13}$.

**4. FFP-дерево и поиск частых элементов произведения частичных небинарных порядков**

Предложенная нами конструкция полного FP-дерева (FFP-дерева) является модификацией классического FP-дерева и позволяет просматривать все возможные варианты бинаризации исходных данных.

На рисунке 2 приведен пример FFP-дерева для некоторой модельной задачи с двумя небинарными атрибутами $P_1$ и $P_2$ и одним бинарным атрибутом $P_3$. Пусть для атрибута $P_1$ найдено два значимых порога 1,7 и 1,5, а для атрибута $P_2$ только один значимый порог 1,5. Для атрибутов $P_1$ и $P_2$ строятся полные вершины. Каждой полной вершине соответствует пара $(Q, H)$, где $Q \in \{P_1, P_2\}$, $H$ – набор значимых порогов атрибута $Q$, упорядоченных по убыванию значимости. Из полной вершины для атрибута $P_1$ выходят два ребра. При спуске по каждому из этих ребер осуществляется бинарная перекодировка значений атрибута $P_1$ и формируется новая база данных, в которой только один небинарный атрибут $P_2$. При спуске из двух полных вершин $(P_2, \{1,5\})$ осуществляется бинарная перекодировка значений атрибута $P_2$ и строятся корневые вершины двух FP-деревьев (на этом шаге все атрибуты бинарные). На рисунке 2 каждой обычной (внутренней) вершине FFP-дерева соответствует пара $(Q, S_{D'}(Q))$, где $Q \in \{P_1, P_2, P_3\}$ и $D'$ текущая бинарная база данных. Для наглядности метки внутренних вершин в FP-деревьях для атрибутов $P_1$ и $P_2$ указаны с соответствующим порогом перекодировки.

Пусть $H_1 = \{p_{1,\dots,}p_n\}$ и $H_2 = \{q_{1,\dots,}q_n\}$, – два значимых набора порогов таких, что $p_t \neq q_t$ для некоторого $t \in 1,2,\dots,n$ и $p_i = q_i$ при всех $i \neq t$, $i \in \{1,2,\dots,n\}$.

Пусть $W(H_1)$ – множество $s$-частых элементов из $P_{H_1}$, $W(H_2)$ – множество $s$-частых элементов из $P_{H_2}$. Имеет место

**Утверждение 1.** Если $S_D(\varphi(p_t)) \geq S_D(\varphi(q_t))$, то $W(H_2) \subseteq W(H_1)$.

Пусть $W_t(H_1)$ – множество $s$-частых элементов из $P_{H_1}$, в которых значение атрибута с индексом $t$ равно нулю, $W_t(H_2)$ – множество $s$-частых элементов из $P_{H_2}$ в которых значение атрибута с индексом $t$ равно нулю . Имеет место

**Утверждение 2.** Если $S_D(\varphi(p_t)) \geq S_D(\varphi(q_t))$, то $W_t(H_1) = W_t(H_2)$.

Утверждения 1 и 2 позволяют существенно сократить время поиска $s$-частых элементов для каждого значимого набора порогов.

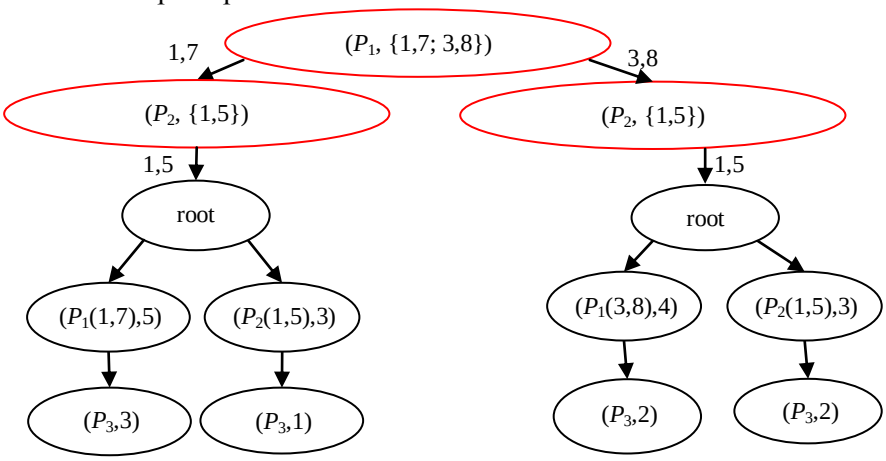


**Рисунок 2.** Пример FFP-дерева для задачи с двумя небинарными атрибутами ($P_1$ и $P_2$) и одним бинарным ($P_3$).

## 5. Результаты экспериментов на модельных данных

Проведены эксперименты на случайных модельных данных, содержащих бинарные и небинарные целочисленные атрибуты. Число бинарных атрибутов составляло 90% от общего числа атрибутов. Значение бинарного атрибута в транзакции выбиралось следующим образом. Это значение полагалось равным 1 с вероятностью $q$, $q \in [0{,}1; 0{,}5]$, и равным 0 с вероятностью $1 - q$. Число $q$ выбиралось с использованием датчика случайных чисел. Значения небинарного атрибута брались из интервала [0; 9] с равной вероятностью. Число атрибутов $n$ варьировалось от 20 до 40, число транзакций $m$ варьировалось от 20 до 300. Искались частые элементы с поддержкой не менее $0{,}3$.

На рисунках 3 и 4 приведены результаты сравнения времени поиска всех максимальных частых элементов в случае использования FFP-дерева и в случае последовательного построения классических бинарных FP-деревьев для всех значимых наборов порогов. В каждом из этих случаев для фиксированных $m$ и $n$ указано среднее время поиска при обработке 10 случайных баз данных. На рисунке 3 изображены результаты поиска при $m = 40$ и $n = 20, 25, 30, 35, 40$. На рисунке 4 изображены результаты поиска при $n = 40$ и $m = 20, 40, 80, 120, 150, 300$ .

Из представленных на рисунке 3 результатов счёта следует, что при $m = 40$ с ростом $n$ время поиска максимальных частых элементов в случае использования FFP-дерева растёт существенно медленнее, чем при последовательном синтезе классических FP-деревьев. При $n = 40$ указанное время в первом случае примерно в 8 раз меньше, чем во втором случае. Более сильный разрыв во времени счёта наблюдается и с ростом $m$ при фиксированном $n = 40$ (см. рисунок 4).

В таблицах 2 и 3 для каждой пары чисел $m$ и $n$ через точку с запятой приведено число максимальных частых элементов и число ассоциативных правил, найденных в модельных базах данных в случае использования FFP-дерева и классического FP-дерева. При поиске ассоциативных правил значение достоверности полагалось равным 0,9. Классическое FP-дерево

строилось для оптимального значимого набора порогов, т.е. такого набора порогов $\{p_1,\dots,p_n\}$, в котором для каждого $p_i$, $i \in \{1,2,\dots,n\}$, выполняется условие $S_D(\varphi(p_i)) = \max_{p \in P_i} S_D(\varphi(p))$.

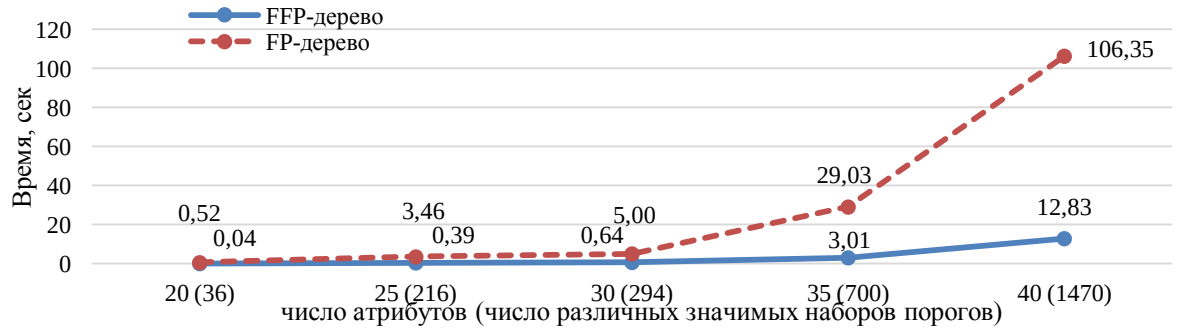


**Рисунок 3.** Зависимость времени поиска максимальных частых элементов от числа атрибутов при $m = 40$.

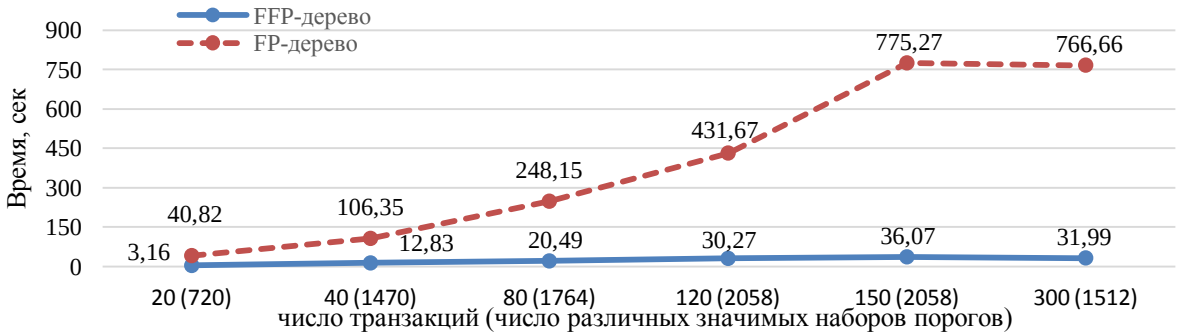


**Рисунок 4.** Зависимость времени поиска максимальных частых элементов от числа транзакций при $n = 40$.

Из представленных в таблицах 2 и 3 результатов следует, что по сравнению с классическим FP-деревом FFP-дерево дает существенный прирост по числу максимальных частых элементов и по числу ассоциативных правил.

**Таблица 2**. Число максимальных частых элементов и ассоциативных правил в FFP-дереве в зависимости от размерности базы данных.

| *m* | *n=20* | *n=25* | *n*=30 | *n*=35 | *n*=40 |
|---|---|---|---|---|---|
| 20 | (249; 92) | (1897; 1608) | (2953; 2097) | (12344; 5557) | (12139; 12142) |
| 40 | (102; 66) | (1360; 1267) | (1374; 465) | (7220; 4803) | (20322; 8954) |
| 80 | (116; 41) | (1909; 1404) | (2163; 786) | (22512; 14065) | (23255; 18664) |
| 120 | (105; 43) | (1828; 735) | (2172; 1415) | (37402; 27805) | (27144; 12707) |
| 150 | (211; 34) | (2007; 480) | (2552; 777) | (21192; 13797) | (34607; 26983) |
| 300 | (210; 92) | (1477; 688) | (2551; 1079) | (29153; 25661) | (22799; 14774) |

**Таблица 3**. Число максимальных частых элементов и ассоциативных правил в классическом FP-дереве в зависимости от размерности базы данных.

| *m* | *n=20* | *n=25* | *n*=30 | *n*=35 | *n*=40 |
|---|---|---|---|---|---|
| 20 | (7; 9) | (9; 51) | (23; 49) | (27; 78) | (29; 114) |
| 40 | (6; 7) | (9; 50) | (10; 8) | (18; 67) | (33; 66) |
| 80 | (4; 7) | (10; 66) | (13; 42) | (32; 118) | (25; 147) |
| 120 | (4; 8) | (16; 19) | (19; 55) | (40; 202) | (40; 82) |
| 150 | (7; 6) | (17; 18) | (18; 29) | (28; 133) | (42; 243) |
| 300 | (8; 18) | (11; 36) | (16; 44) | (23; 238) | (39; 160) |

## 6. Заключение

В работе исследованы вопросы анализа небинарных данных, представленных в виде произведения частично упорядоченных множеств. Для указанного вида данных введено понятие ассоциативного $(s,c)$-правила, опирающегося на понятие $s$-частого элемента произведения частичных порядков, которое ранее было дано в [7]. Предложен подход к бинаризации исходных данных, основанный на построении значимого набора порогов. Рассмотрена задача поиска $s$-частых элементов и ассоциативных $(s,c)$-правил произведений частичных бинарных порядков, порождаемых всеми возможными значимыми наборами порогов, т.е. всеми возможными вариантами бинаризации исходных данных. С целью сокращения временных затрат при решении поставленной задачи усовершенствована конструкция классического FP-дерева. Новая конструкция дерева названа полным FP-деревом (FFP-деревом). Приведено теоретическое и экспериментальное обоснование подхода.

## 7. Благодарности

Работа выполнена при частичной финансовой поддержке РФФИ (проект № 19-01-00430-а).

## 8. Литература

[1] Agrawal, R. Mining association rules between sets of items in large databases / R. Agrawal, T. Imielinski, A. Swami // Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 1993. – P. 207-216.

[2] Imberman, S. Finding Association Rules from Quantitative Data Using Data Booleanization / S.P. Imberman, B. Domanski // AMCIS Proceedings, 2001. – P. 369-375.

[3] Angiulli, F. On the complexity of inducing categorical and quantitative association rules / F. Angiulli, G. Ianni, L. Palopoli // Theoretical Computer Science. – 2004. – Vol. 314(1). – P. 217-249.

[4] Aggarwal, C. Frequent Pattern Mining / C. Aggarwal, H. Jiawei – Springer, 2014. – 469 p.

[5] Han, J. Mining Frequent Patterns without Candidate Generation / J. Han, H. Pei, Y. Yin // Proc. Conf. on the Management of Data (SIGMOD'00, Dallas, TX), 2000. – P. 1-12.

[6] Генрихов, И.Е. О поиске ассоциативных правил в небинарных данных / И.Е. Генрихов, Е.В. Дюкова // 19-я Всероссийская конференция с международным участием «Математические методы распознавания образов» – М.: Российская академия наук, 2019. – С. 15-19.

[7] Elbassioni, K.M. On Finding Minimal Infrequent Elements in Multi-dimensional Data Defined over Partially Ordered Sets // arXiv:1411.2275, 2014. – 30 p.

[8] Borgelt, C. Frequent Item Set Mining // Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. – 2012. – Vol. 2(6). – P. 437-456.

# Finding frequent elements for a product of partial orders and association rules

**I.E. Genrikhov[1], E.V. Djukova[2]**

[1]LLC «Mobail Park IT», Panfilova street 21/1, Himki, Russia, 141407
[2]CC FRC CSC RAS, Vavilova street 44/2, Moscow, Russia, 119333

**Abstract.** One of the known ways to find frequent sets of attributes when searching for association rules in a binary database is based on the construction of a FP-tree (Frequent Pattern Growth Tree). In the case of binary data, as a rule, the values of each non-binary attribute are binarized by some numerical threshold and the problem is reduced to the construction of a classical FP-tree. The result depends significantly on the choice of the specified thresholds. The paper proposes to modify the construction of the classical FP-tree by introducing an additional vertex for each non-binary attribute, called a full vertex and containing information about possible variants for binarization of the values of this attribute. The new FP-tree model is named full FP-tree. We consider the application of the full FP-tree for nonbinary data analysis, provided that partial orders are specified on the attribute values. Illustrative model examples are given.