УДК 004.032.34

АЛГОРИТМ ПОИСКА ЧАСТЫХ ПОДПОСЛЕДОВАТЕЛЬНОСТЕЙ В WF-МОДЕЛЯХ «F-ПОИСК»

Н.Р. Ляпин

ГОУ ВПО «Тамбовский государственный технический университет», г. Тамбов

Рецензент В.Г. Матвейкин

Ключевые слова и фразы: алгоритм «Apriori»; журнал выполнения бизнес-процесса; поиск частых подпоследовательностей; свойство антимонотонности.

Аннотация: Представлен эффективный алгоритм интеллектуального анализа (поиска частых подпоследовательностей) выполнения бизнес-процессов в системе электронного документооборота, превосходящий по производительности известные алгоритмы.

Задача поиска частых подпоследовательностей встречается во многих предметных областях: анализ потребительской корзины в супермаркетах, биоинформатика, анализ распространения пожаров по территории [1] и т.п. При анализе выполнения бизнес-процесса в системе электронного документооборота также возникают подобные задачи [2], эффективный алгоритм решения которых представлен в этой работе.

Определим необходимые в дальнейшем понятия. Пусть имеется WF-модель [3] P и множество экземпляров $F = \{I_1,...,I_n\}$. Граф $p = \langle A_p, E_p \rangle \subseteq P$ называется F-шаблоном если существует $I = \langle A_I, E_i \rangle \in F$ такое, что $A_p \subseteq A_I$ и p подграф I, включенный посредством узлов в A_p.

Пусть $\text{freq}(p) = |\{I \in F \mid I| = p\}| / |F|$ поддержка F-шаблона [3] p, тогда отсюда вытекает следующее определение.

Определение 1. FCPD(σ): задача поиска всех связанных шаблонов, поддержка которых больше, чем σ.

Ляпин Н.Р. – аспирант кафедры «Информационные процессы и управление» ТамбГТУ, г. Тамбов

При прямолинейном подходе к решению задачи напрашивается алгоритм поиска шаблонов на основе простой генерации прямо связанных подграфов, с дальнейшим тестированием в полиномиальном времени. Является ли он экземпляром *P*? Другое предложение основано на идее ослабления количества шаблонов, подлежащих генерации. Мы можем рассматривать только «закрытые» графы (с учетом глобальных и локальных ограничений), то есть таких, что $p\big| = c$ для всех $c \in C_L \cup C_G$. Будем называть такие графы *ослабленными шаблонами* или просто *w*-шаблоны. Формализуем приведенные рассуждения.

Определение 2. Дана WF-модель $p = \langle A_p, E_p \rangle$, *детерменестическое закрытие p* (*ws_закрытие* (*p*)) определяется, как граф $p' = \langle A_{p'}, E_{p'} \rangle$ такой, что: (1) $A_p \subseteq A_{p'}$, и $E_p \subseteq E_{p'}$, (2) $a \in A_{p'}$ - *and-join* подразумевает, что для каждого $(b,a) \in E$, $(b,a) \in E_{p'}$ и $b \in A_{p'}$, (3) $a \in A_{p'}$ – *deterministic fork* подразумевает, что для каждого $(a,b) \in E$ с *b - or-join*, $(a,b) \in E_{p'}$ и $b \in A_{p'}$. Более того граф *p* такой, что *p* = *ws_закрытие* (*p*) называется *p ws_закрытым*.

Определение 3. *Ослабленный шаблон*, или просто *w*-шаблон, это *ws_закрытый* граф *p*, такой что для каждого узла *a*, $\left|\{(a,b)|(a,b) \in E_p\}\right| \le OUT_{\max}(a)$.

Определение 4. Пусть дана WF-модель $WS = \langle A, E \rangle$, тогда для каждого $a \in A$ граф *p* = *ws_закрытие*(<{*a*},{}>) называется элементарным *w*-паттерном.

Для поиска решения задачи используется алгоритм, который инкрементально строит частые *w*-шаблоны, стартуя с «*элементарных*» *w*-шаблонов (описанных ниже) и расширяя каждый частый шаблон посредством использования двух базовых операций: добавление частой дуги и слияние с другим элементарным *w*-шаблоном.

Элементарные *w*-шаблоны, с которых начинается построение частых шаблонов, получаются как детерменестическое закрытие отдельных узлов. Шаблон является элементарным *w*-шаблоном (обозначим через *ew*-шаблон) для узла *a*, если это минимальный *w*-шаблон, содержащий *a*. Множество всех *ew*-шаблонов обозначается как *EW*. Более того, пусть *p w*-шаблон, тогда EW_p обозначает множество элементарных *w*-шаблонов, содержащихся в *p*. Заметим, что если дан *ew*-шаблон *e*, EW_e не обязательно содержит только один элемент. Кроме того, данное множество $E' \subseteq EW$, $\text{Compl}(E') = EW - \bigcup_{e \in E'} EW_e$ содержит все элементарные шаблоны, которые не содержатся ни в E, ни в одном элементе E'.

При работе алгоритма (рис. 1) инкрементально строятся частые *w*-шаблоны с использование двух базовых операций: добавление частой дуги и слияние с другим частым *w*-шаблоном.

Вход: WF – *модель* WS, *множество экземпляров* $F = \{I_1, ..., I_N\}$ *для модели* WS

Выход: Множество *частых* F – *шаблонов*

Метод:

$L_0 := \{e \mid e \in EW, e$ *является частым шаблоном с учетом* $F\}$;

$k := 0; R := L_0$;

$FrequentArcs := \{(a,b), (a,b) \in E^{\subset}, < \{a,b\}, \{(a,b)\}$ *является частым с учетом* $F >\}$

$E_f^{\subset} := E^{\subset} \cap FrequentArcs$;

повтор

$U := 0$;

forall $p \in L_k$ *do begin*

$U := U \cup addFrequentArc(p)$;

forall $e \in Compl(EW_p) \cap L_0$ *do*

$U := U \cup addFrequentEWPattern(p, e)$;

end

$L_{k+1} := \{p \mid p \in U, p -$ *частый с учетом* $F\}$;

$R := R \cup L_{k+1}$

пока $L_{k+1} = 0$

return R;

Function $addFrequentEWPattern(p =< A_p, E_p >, e =< A_e, E_e >) : w -$ *шаблон*;

$p' =< A_p \cup A_e, E_p \cup E_e >$;

if p' *связанный then return* p' *else return* $addFrequentConnection(p', p, e)$

Function $addFrequentConnection(p' =< A_{p'}, E_{p'} >, p =< A_p, E_p >, e =< A_e, E_e >)$

$S := 0$

forall $frequent(a,b) \in E_f^{\subset} - E_p \; (a \in A_p, b \in A_e) \vee (a \in A_e, b \in A_p)$ *do begin*

$q =< A_{p'}, E_{p'} \cup (a,b) >$;

if $WS \models q$ *then* $S := S \cup \{q\}$;

end

return S;

Function $addFrequentArc(p =< A_p, E_p >) :$ *шаблон*

$S := 0$

forall $frequent(a,b) \in E_f^{\subset} - E_p \; a \in A_p, b \in A_p$ *do begin*

$p' =< A_p, E_p \cup (a,b) >$

if $WS \models p'$ *then* $S := S \cup \{p'\}$;

end

return S

Рис. 1. Алгоритм поиска частых подпоследовательностей «F-поиск»

Элементарные шаблоны, с которых начинается алгоритм, получаются как *ws_закрытие*(*p*) для каждого из узлов. Далее идет вычисление в главном цикле алгоритма, где каждое новое значение для L_{k+1} получается путем расширения любого шаблона *p*, сгенерированного на предыдущем шаге

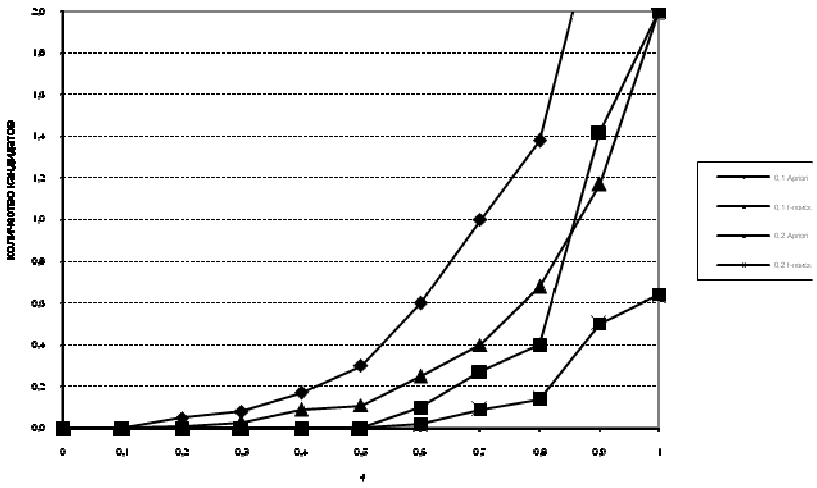

Рис. 2. Сравнение алгоритмов «Apriori» и «F-поиск»: количество кандидатов при различных значения коэффициента F и порога минимальной поддержки

($p \in L_k$) двумя способами: 1) добавлением частой дуги из $E^{\subseteq}$ (посредством функции *addFrequentArc*), 2) добавлением элементарного *w*-шаблона (функция *addFrequentEWPattern*). Каждый шаблон p', генерируемый функциями, указанными выше, – допустимый подграф для *WS*, то есть для каждой $a \in A_{p'}$, $OutDegree_{p'}(a) \le OUT_{\max}(a)$.

На рис. 2 приводится сравнение производительности алгоритмов «F-поиск» и «Apriori» [1]. Из рисунка видно, что количество кандидатов, генерируемых алгоритмом «F-поиск», меньше чему у алгоритма «Apriori» при прочих равных параметрах. В связи с чем можно говорить о более высокой производительности «F-поиск», учитывая необходимость прохода записей базы данных для каждого кандидата из множества.

Предложенный алгоритм частых подпоследовательносей «F-поиск» является адаптацией уже известных [1] алгоритмов интеллектуального анализа, превосходя их по производительности при решении задач данной предметной области (поиска частых подпоследовательностей в журнале выполнения бизнес-процесса при наличии WF-модели).

Список литературы

1. Agrawal, R. Fast algorithms for mining association rules / R. Agrawal, R. Srikant // Proc. Of the 20th Int'l Conference on Very Large Databases. – 1994. – Pp. 487–499.

2. Ляпин, Н.Р. Автоматизация делопроизводства как инструмент повышения качества управления производством / Н.Р. Ляпин, Б.С. Дмитриевский // Компьютерные технологии в науке, производстве, социальных и экономических процессах : Материалы IV Междунар. науч.-практ. конф.,

г. Новочеркасск, 14 ноября 2003 г. : В 4 ч. / Юж.-Рос. гос. техн. ун-т (НПИ). – Новочеркасск, 2003. – Ч. 4. – С. 33–34.

3. Van der Aalst W.M.P., Weijters A.J.M.M., Maruster L. Workflow Mining: Discovering Process Models from Event Logs. QUT Technical report, FIT-TR-2003-03, Queensland University of Technology, Brisbane, 2003.

Algorithm of Search for Frequent Subsequences in WF-Models "F-Search"

N.R. Lyapin

Tambov State Technical University, Tambov

Key words and phrases: algorithm "Apriori"; workflow log; frequent subsequences search; antimonotone property.

Abstract: Effective algorithm of intellectual analysis (search for frequent subsequences) of business processes accomplishment within electronic document management system is presented. The given algorithm is more efficient than other existing ones.
