STUDIA INFORMATICA 2011
Volume 32 Number 2A (96)

Natalia OGIEGŁO, Marek SIKORA
Politechnika Śląska, Instytut Informatyki

WYBRANE ALGORYTMY INDUKCJI REGUŁ TEMPORALNYCH

Streszczenie. W niniejszym artykule zostały zaprezentowane wybrane algorytmy indukcji reguł temporalnych. Szczegółowo opisano algorytm ARMADA oraz jego modyfikację, wykorzystującą okna przesuwne. Przedstawiono również algorytm ARED, pozwalający wyodrębniać reguły akcji bezpośrednio z tablicy decyzyjnej. Przedstawiono kierunki dalszych prac, artykuł ma charakter przeglądowy.

Słowa kluczowe: reguły temporalne, miary ocen jakości reguł temporalnych

SELECTED ALGORITHMS OF TEMPORAL RULES INDUCTION

Summary. In this paper, selected algorithms for temporal rules mining were presented. The ARMADA algorithm was described in details such as its modification utilizing sliding windows technique. Furthermore the ARED algorithm was introduced which allows to discover action rules directly from decision table.

Future work directions were proposed. This article has the character of a review.

Keywords: temporal rules, quality measures of temporal rules

1. Wstęp

Reprezentacje regułowe znajdują bardzo szerokie zastosowanie w dziedzinie maszynowego uczenia oraz odkrywania wiedzy w bazach danych [22]. Najczęściej spotykamy się z zastosowaniami reguł klasyfikacyjnych [26] lub asocjacyjnych [25]. Rzadziej reprezentacje regułowe wykorzystuje się do rozwiązywania zadań regresji [27]. Popularność reprezentacji regułowych wynika z cechy, jakiej nie mają tzw. subsymboliczne [21] metody uczenia, mianowicie możliwości interpretacji uzyskanego regułowego modelu danych. Interpretacja modelu umożliwia podjęcie próby zrozumienia zależności, jakie udało się odkryć w danych. Bogata literatura dziedzinowa i setki, a nawet tysiące badań porównawczych pokazują, że nie sposób określić jednego, najlepszego algorytmu indukcji reguł, który uzyskiwałby każdora-

zowo najlepsze wyniki zarówno w aspekcie klasyfikacji, jak i opisu. Stąd poza rozwojem uniwersalnych algorytmów indukcji [17, 18, 19, 20] spotyka się wiele algorytmów dopasowanych do konkretnego zastosowania [28, 29]. Standardowo stosowane algorytmy nie uwzględniają zależności czasowych pomiędzy analizowanymi przykładami, tzn. kolejność występowania przykładów w analizowanym zbiorze danych jest nieistotna. W ostatnich kilkunastu latach rozpoczęto prace nad tzw. regułami temporalnymi, które uwzględniają aspekt zależności czasowej pomiędzy prezentowanymi przykładami.

Poza standardowymi, znanymi z zastosowań reguł klasyfikacyjnych i asocjacyjnych, dziedzinami zastosowań reguł temporalnych mogą być obszary dotychczas niedostępne (lub trudno dostępne – np. wymagające dodatkowego przekształcania danych) dla „zwykłych" reguł. Mamy tutaj na myśli analizę przebiegu terapii medycznych, analizę przebiegu procesu sterowania, analizę zachowań klientów czy też analizę zjawisk fizycznych.

Celem niniejszego artykułu jest przedstawienie podstawowych definicji (rozdział 2) i algorytmów związanych z indukcją reguł temporalnych (rozdział 3), a także zaprezentowanie możliwych kierunków rozwoju systemów indukcji reguł temporalnych (podsumowanie).

2. Podstawowe definicje

Mówiąc o danych mających wymiar czasowy należy przede wszystkim określić sposób reprezentacji danych. Podstawową jednostką danych jest zdarzenie [9] czy też epizod (*ang. event*). Każde zdarzenie określane jest poprzez czas wystąpienia oraz rodzaj zdarzenia.

Definicja 1

Jeżeli *TS* jest zbiorem pojedynczych punktów w czasie, wówczas zdarzeniem nazywamy parę $e = (E, t)$, w której E określa rodzaj zdarzenia, natomiast $t \in TS$ określa czas wystąpienia zdarzenia e. Zbiór wszystkich zdarzeń oznaczany jest jako Ω.

W niektórych przypadkach dane temporalne lepiej poddają się analizie, jeśli są traktowane nie jako punkty w czasie, lecz jako sekwencje zdarzeń jednego typu, pojawiające się z określonym interwałem czasowym [11].

Definicja 2

Sekwencją zdarzeń *ES związaną* z pewnym ustalonym obiektem c oraz z określonym zdarzeniem typu E nazywamy uporządkowany ciąg $ES = \langle e_1, e_2, \ldots, e_n \rangle$, w którym $e = (E, t_i)$, $t_i \in TS$ oraz prawdziwa jest nierówność $t_i \leq t_{i+1}$ dla każdego $i = 1, 2, \ldots, n - 1$.

Definicja 3

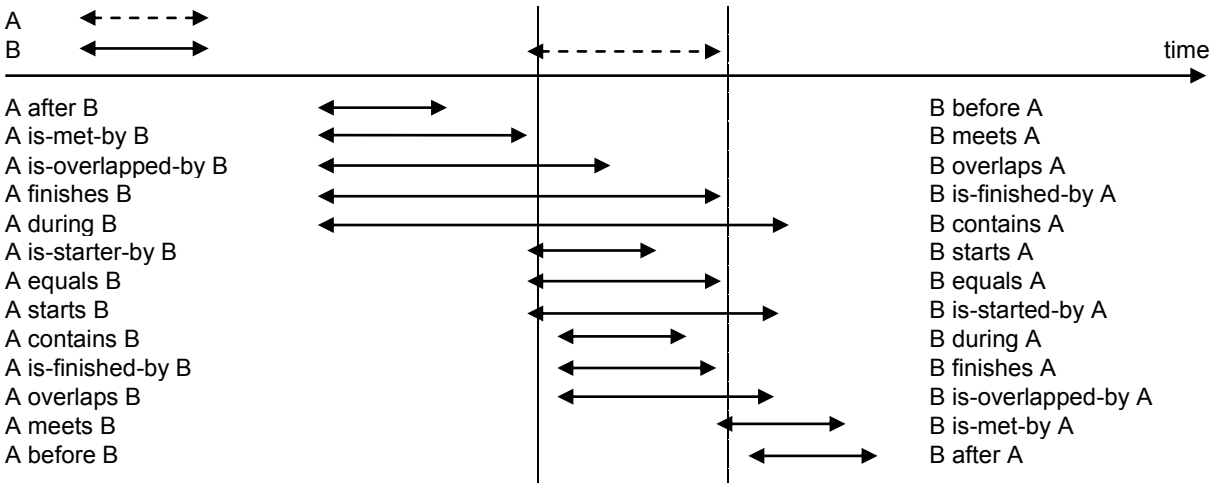
Sekwencją zdarzeń typu E z interwałem czasowym jest ciąg $ES' = <e_1, e_2, \ldots, e_n>$, gdzie $e_j=(E,[vs_j, ve_j]), vs_j \leq ve_j \leq vs_{j+1} \leq ve_{j+1}$ dla każdego $j = 1, 2, \ldots, n$, natomiast E jest typem zdarzenia e_j.

Mając zdefiniowane podstawowe pojęcia dotyczące zdarzeń i ciągu zdarzeń można określić relacje, jakie zachodzą pomiędzy poszczególnymi zdarzeniami. Po raz pierwszy relacje między zdarzeniami zostały zdefiniowane w pracy [12]. Relacja temporalna określona jest w następujący sposób:

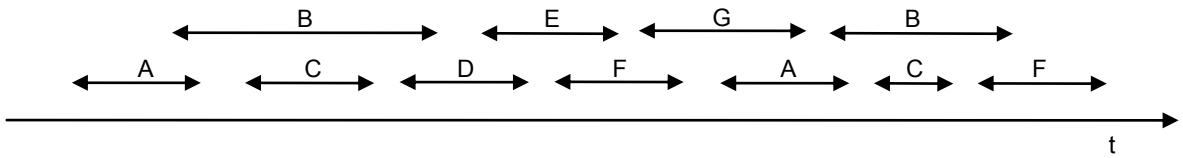
Definicja 4

Dla każdej pary zdarzeń można określić relację R, taką że $R(e_i,e_j) = \{P(e_i,e_j) : e_i,e_j \in \Omega, P \in IO\}$, gdzie: Ω jest zbiorem zdarzeń, natomiast IO jest zbiorem operatorów relacji.

Na rys. 1 zaprezentowane zostały rodzaje operatorów relacji, wprowadzone przez Allena [12].



Rys. 1. Operatory relacji Allena
Fig. 1. Allen's relation operators



Rys. 2. Kolejność zdarzeń
Fig. 2. Events' sequence

Określenie relacji pomiędzy poszczególnymi zdarzeniami pozwala wprowadzić nowe pojęcie, jakim jest wzorzec sekwencyjny [13]. Zależności, które zachodzą pomiędzy zdarzeniami można przedstawić w postaci macierzy kwadratowej o wymiarze k, który jednocześnie określa rozmiar wzorca. Załóżmy, iż w bazie danych zdarzenia występują w kolejności analogicznej do przedstawionej na rys. 2.

Na podstawie powyższego rysunku można określić, iż zdarzenie C jest zawsze poprzedzone wystąpieniem zdarzenia A. Również nietrudno zauważyć, że zawsze występuje zdarzenie B zawierające zdarzenie C. Zależności pomiędzy tymi trzema zdarzeniami zostały zaprezentowane w macierzach relacji.

	A	**C**
A	=	b
C	a	=

	A	**C**	**B**
A	=	b	io
C	a	=	d
B	o	c	=

Rys. 3. Macierze relacji pomiędzy zdarzeniami
Fig. 3. Matrixes of relations between events

W tabelach przedstawionych na rys. 3 przyjęto następujące oznaczenia dla operatorów relacji:

a – after,

b – before,

o – overlaps,

io – is-overlapped-by,

d – during,

c – contains.

***Definicja 5* (*Wzorzec temporalny* [1])**

Załóżmy, że dostępnych jest n zdarzeń $ES = <e_1, e_2, \ldots, e_n>$, wówczas wzorcem temporalnym rozmiaru $1 < m \leq n$ nazywamy parę $p = (ES, M)$, w której M jest macierzą kwadratową o rozmiarze $m \times m$ zawierającą elementy $M[i, j]$, oznaczające relacje pomiędzy stanami e_i i e_j.

Liczba elementów wzorca temporalnego p oznaczana jest jako $\dim(p)$. Jeśli $\dim(p) = k$, wówczas wzorzec nazywany jest k – wzorcem.

Mając zdefiniowane pojęcie wzorca temporalnego można przejść do definicji reguły temporalnej [1].

Definicja 6

Regułą temporalną nazywamy wyrażenie postaci $X \rightarrow Y$, w którym X i Y są wzorcami temporalnymi takimi, że X jest wzorcem składowym wzorca Y.

Przykładowo dla zależności przedstawionych na rys. 2 można powiedzieć – patrząc na obie macierze zależności pomiędzy zdarzeniami – iż wystąpienie zdarzenia A implikuje wystąpienie zdarzenia C. W ten sposób sformułowany wniosek staje się regułą temporalną. Przedstawiony powyżej sposób wnioskowania został wykorzystany do stworzenia wielu algorytmów generujących reguły temporalne.

Równocześnie z rozwojem technik odkrywania zależności pomiędzy sekwencjami zdarzeń oraz reguł temporalnych pojawiła się konieczność określenia kryteriów istotności badanych reguł. Przy ogromnej ilości danych wejściowych dość trudnym zadaniem okazało się

rozpoznanie znaczących powiązań temporalnych, wyrażanych za pomocą reguł temporalnych. Najczęstszym efektem działania algorytmów indukcji reguł temporalnych (podobnie jak w przypadku reguł asocjacyjnych) jest generowanie bardzo dużej liczby reguł. Taka sytuacja powoduje trudności w interpretacji uzyskanego modelu danych oraz przysparza problemów na samym etapie analizy, wydłużając czas obliczeń.

W przypadku reguł temporalnych – podobnie jak w indukcji reguł asocjacyjnych i klasyfikacyjnych – wprowadzono pojęcia wsparcia (ang. *support*) i pewności (ang. *confidence*) reguły [3].

Definicja 7

Transakcją nazywamy zbiór zdarzeń typu $e = (C_{id}, E, t)$, gdzie C_{id} jest identyfikatorem obiektu, dla którego wystąpiło zdarzenie typu E w punkcie czasu t.

Definicja 8

Obiektem (klientem) C nazywamy sekwencję transakcji T taką, że $C = <T_1, T_2, \ldots, T_n>$, gdzie $ts(T_i) < ts(T_j)$, jeśli $i < j$. Wyrażenie $ts(T_i)$ oznacza punkt w czasie, w którym transakcja T_i wystąpiła.

Definicja 9

Wsparciem supp zdarzenia $e = (E, t)$ nazywamy stosunek liczby obiektów C_e, dla których wystąpiło zdarzenie e do liczby wszystkich rozpatrywanych obiektów C.

Definicja 10

Wsparciem minimalnym $supp_{min}$ zdarzenia e nazywamy minimalną liczbę obiektów C_e, dla których powinno wystąpić zdarzenie e, aby zdarzenie e było określane, jako zdarzenie częste.

Sekwencję zdarzeń ES (a także sekwencję zdarzeń z interwałem czasowym ES') nazywamy częstą sekwencją zdarzeń, jeśli spełniony jest dla niej warunek $supp(ES) \geq supp_{min}$ ($supp(ES') \geq supp_{min}$). Również każdy wzorzec temporalny, dla którego $supp(p) \geq supp_{min}$ nazywany jest częstym wzorcem temporalnym.

Parametr wsparcia minimalnego bardzo często wykorzystywany jest do ograniczenia liczby generowanych reguł. Jeżeli wartość $supp_{min}$ zostanie ustawiona na wysoką wartość, wówczas do generowania reguł temporalnych będą kwalifikowane tylko zdarzenia występujące bardzo często, a same reguły pozwolą opisać zjawisko na dość ogólnym poziomie. Natomiast ustawienie progu $supp_{min}$ na wartość zbyt małą będzie skutkowało generowaniem dużej liczby często nieistotnych reguł temporalnych oraz wydłużeniem czasu wykonania algorytmu.

Definicja 11

Ufnością reguły temporalnej $X \rightarrow Y$ nazywamy liczbę zdefiniowaną następująco:

$$\mathrm{Conf}(X \rightarrow Y) = \frac{\mathrm{supp}(X)}{\mathrm{supp}(Y)} \tag{1}$$

Ufność reguły temporalnej określa, z jakim prawdopodobieństwem zachodzi opisana regułą zależność. Przykładowo, jeśli 2 z 6 wystąpień wzorca temporalnego p_1 jest poprzedzonych wystąpieniem wzorca p_2, wówczas ufność reguły określonej jako $p_2 \rightarrow p_1$ wynosi 33%.

Poniżej przedstawiono wartości parametrów supp i conf dla poszczególnych wzorców temporalnych, utworzonych dla przykładowych danych z tabeli 1.

X = (A), supp = 3, conf = 75%
X = (B), supp = 3, conf = 75%
X = (C), supp = 4, conf = 100%
X = (D), supp = 4, conf = 100%
X = (E), supp = 3, conf = 75%
Y = (A after E), supp = 1, conf = 25%
Y = (B after E), supp = 2, conf = 75%
Y = (B before D), supp = 2, conf = 50%
Y = (D during C), supp = 2, conf = 50%
Y = (E before B), supp = 3, conf = 75%

Korzystając z obliczonych wartości supp i conf dla 1-elementowych wzorców temporalnych obliczono wartości supp dla wybranych utworzonych reguł temporalnych.

Conf(E → E before B) = 100%
Conf(B → B after E) = 100%

Oprócz przedstawionych powyżej kryteriów podczas wieloletnich badań powstały również inne. Przykładowo niektóre algorytmy definiują czasy minimalny i maksymalny odstępu pomiędzy dwoma zdarzeniami, należącymi do wzorca czy też maksymalną długość generowanych wzorców temporalnych [1]. Ponadto, istnieje pojęcie interesującego wzorca, które oznacza, iż jest on wart odkrycia i wnosi nową informację [7]. Jednak definicje interesujących wzorców zasadniczo różnią się między sobą w zależności od autora i z uwagi na przeglądowy charakter tego artykułu nie zostaną przybliżone.

3. Przegląd algorytmów

W niniejszym rozdziale zostaną przedstawione dwa najważniejsze algorytmy indukcji reguł temporalnych. Ukazano tu także algorytm ARED [10], który nie jest dedykowany dla

odkrywania reguł temporalnych, ale zdaniem autorów niniejszego artykułu reguły generowane przez ten algorytm można uznać jako specjalny rodzaj reguł temporalnych.

3.1. Algorytm ARMADA

Tabela 1

Postać danych wejściowych dla algorytmu ARMADA

Dzień	Czynnik	Początek	Koniec	Zależność z innymi zdarzeniami
1	A	10	15	
1	E	3	5	
1	B	6	10	
1	D	13	15	
1	C	12	18	
2	B	7	12	
2	D	4	8	
2	C	13	18	
2	E	2	5	
3	A	10	18	
3	C	1	7	
3	D	2	6	
3	E	9	17	
4	B	7	16	
4	E	2	7	
4	C	3	6	
4	D	14	18	

Algorytm ARMADA został zaproponowany przez Winiarko i Roddicka [1]. Danymi wejściowymi dla algorytmu jest uporządkowany zbiór zdarzeń opatrzonych datą wystąpienia oraz identyfikatorem. Dane wejściowe przeważnie mają postać bazy danych. W tabeli 1 przedstawiono przykładowy zbiór danych wejściowych. Poszczególne wiersze tabeli prezentują różne czynniki atmosferyczne wpływające na pogodę. Każdy czynnik pogodotwórczy został przyporządkowany do dnia, w którym wystąpił z zaznaczeniem czasu wystąpienia. Poszczególne czynniki zostały oznaczone jako A, B, C, D lub E.

Algorytm ARMADA działa dwuetapowo. W pierwszym kroku, będącym rozszerzeniem algorytmu MEMISP [14], wejściowy zbiór danych wczytywany jest do pamięci. W trakcie procesu pobierania danych obliczana jest wartość parametru wsparcia dla każdego z występujących stanów i na podstawie kryterium $supp_{min}$ tworzony jest zbiór częstych stanów. Następnie każdy z takich stanów jest łączony kolejno z pozostałymi częstymi stanami, które występują po nim w danych wejściowych. Dla każdego nowo utworzonego wzorca obliczana jest wartość wsparcia supp. Jeśli dany stan spełnia podane kryterium $supp_{min}$, wówczas rekursywnie dołączane są kolejne stany z analizowanej bazy danych i tworzone są coraz dłuższe wzor-

ce. Każdorazowo sprawdzane jest kryterium wsparcia minimalnego, wzorce niespełniające tego kryterium są odrzucane.

Rezultatem działania algorytmu jest zbiór częstych wzorców temporalnych. Przykładowo, dla zaprezentowanych danych otrzymamy między innymi następujące wzorce temporalne:

X =

	B	**E**
B	=	a
E	*	=

Y =

	B	**E**	**C**
B	=	a	b
E	*	=	b
C	*	*	=

Rys. 4. Otrzymane wzorce temporalne
Fig. 4. Resulting temporal patterns

Następnie na podstawie otrzymanych wzorców temporalnych konstruowane są asocjacyjne reguły temporalne. Zgodnie z definicją 6 dla dwóch wzorców temporalnych, przedstawionych na rys. 2, możemy utworzyć regułę temporalną X → Y, której pewność wynosi 100%. Interpretacja otrzymanej reguły byłaby następująca: „jeśli pojawi się zdarzenie E, a po nim zdarzenie B, wówczas jest wysoce prawdopodobne, że wystąpi również zdarzenie C".

Ponieważ zbiór reguł generowanych przez algorytm ARMADA może być bardzo liczny, wprowadza się ograniczenia umożliwiające wyodrębnienie tylko najbardziej istotnych reguł. Pierwszym jest wartość będąca jednocześnie parametrem wejściowym algorytmu, określająca minimalne wsparcie $supp_{min}$. Innym ograniczeniem może być parametr maxgap zdefiniowany, jako maksymalny czas odstępu pomiędzy zdarzeniami wchodzącymi w skład wzorca. Również wartość określająca ufność może być wykorzystana do zawężenia zbioru wynikowego stworzonych reguł temporalnych.

Przykładem zastosowania algorytmu ARMADA może być próba określenia zależności pomiędzy symptomami chorobowymi, występującymi u pacjentów cierpiących na pewną chorobę. Traktując wystąpienie objawu choroby jako zdarzenie można w łatwy sposób wyodrębnić wzorce temporalne dla analizowanych danych. Na podstawie otrzymanych wzorców temporalnych możliwe jest wygenerowanie reguł temporalnych. Każda z wygenerowanych reguł będzie zawierała informację postaci: „jeśli wystąpi objaw chorobowy A, a po nim w czasie T wystąpi objaw chorobowy B, to z prawdopodobieństwem p wystąpi następnie objaw chorobowy C".

3.2. Algorytm indukcji reguł temporalnych z wykorzystaniem okien przesuwnych

Kolejny algorytm indukcji reguł temporalnych przedstawiono w pracy [9]. Przykład danych wejściowych dla algorytmu został zaprezentowany na rys. 5a, jako modyfikacja przykładu z rozdziału 3.1.

a)

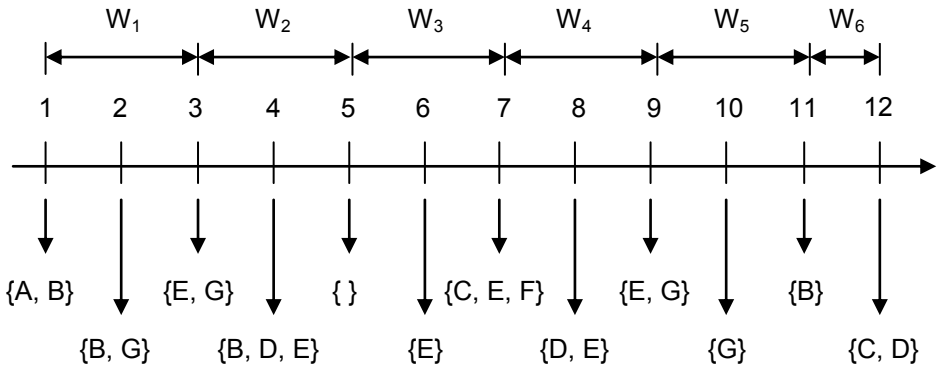
Dzień	Czas wystąpienia	Typ zdarzenia
1	1	B
1	3	G
1	4	B, D, E
1	7	C, E, F
1	9	E
2	1	A, B
2	2	B
2	6	E
2	7	F
2	10	G
3	4	B
3	8	D, E
3	9	E, G
4	2	G
4	3	E
4	11	B
4	12	C, D

b)

Typ zdarzenia	Wsparcie
A	1
B	4
C	2
D	3
E	4
F	2
G	4

Rys. 5. Przykładowe dane wejściowe: a) baza danych, b) częste zdarzenia
Fig. 5. An example of input data: a) database, b) frequent events

Reprezentacja danych wejściowych przedstawiona na rys. 5 jest zbliżona do reprezentacji danych dla algorytmu ARMADA. Jednak konstrukcja algorytmu różni się wykorzystaniem okna przesuwnego jako dodatkowego ograniczenia dla tworzonych wzorców, co zostało opisane w dalszej części tego rozdziału.



Rys. 6. Przykładowe sekwencje zdarzeń
Fig. 6. An example of events' sequences

W pierwszym kroku na podstawie danych wejściowych wybierane są częste zdarzenia, czyli takie, które spełniają warunek wsparcia minimalnego. Przykładowy zestaw zdarzeń został przedstawiony na rys. 5b. Na podstawie wyselekcjonowanych zdarzeń konstruowane są sekwencje zdarzeń, z których każda musi spełniać warunek maksymalnych odstępów czasowych pomiędzy poszczególnymi zdarzeniami składowymi. Na rys. 6 przestawiono schema-

tycznie sekwencję zdarzeń. Szerokość okna W, określająca maksymalną długość sekwencji, jest jednym z parametrów wejściowych algorytmu.

Kolejnym krokiem jest wyodrębnienie tylko tych zdarzeń, które odpowiednio często się powtarzają. Zakładając minimalną częstotliwość na poziomie 33% otrzymamy zbiór zdarzeń, zebrany w tabeli na rys. 7.

Typ zdarzenia	Częstotliwość
B	3
C	2
D	3
E	4
G	2

Rys. 7. Zestawienie częstych zdarzeń
Fig. 7. Summary of frequent events

Następnie korzystając z definicji relacji temporalnych Allena tworzone są relacje temporalne pomiędzy otrzymanymi zdarzeniami. Wynikowy zbiór relacji zawężany jest do tak zwanych relacji częstych, czyli takich, których liczba wystąpień spełnia warunek wsparcia minimalnego.

Przykładem zastosowania przedstawionego algorytmu jest analiza danych, zawierających informacje na temat sprzedaży w księgarni wysyłkowej dokonanej przez poszczególnych klientów. Określenie zbioru interwałowych relacji temporalnych wraz z ich wsparciem pozwoliłoby na przewidywanie sekwencji zdarzeń typu: „75% klientów, którzy zakupili produkt A kupiło również produkt B, a nie więcej niż tydzień później zakupiło jeszcze produkt C”. Ograniczenie czasowe w formie „nie więcej niż tydzień później” dla tej relacji temporalnej wynika bezpośrednio z faktu zastosowania okna przesuwnego o określonym rozmiarze, czyli czasie trwania sekwencji tworzących pojedynczy wzorzec temporalny. Należy również zauważyć, że ten rodzaj trendów nie mógłby być odkryty za pomocą istniejących technik, takich jak odkrywanie wzorców sekwencyjnych (*ang. sequential pattern mining*). Informacje tego rodzaju mogłyby znaleźć szerokie wykorzystanie w dziedzinie marketingu czy e-commerce.

Opisane w niniejszym rozdziale algorytmy, pomimo że definiują skrajnie różne pojęcia, tak naprawdę odnoszą się do tego samego typu danych. Nie ma większego znaczenia czy dane temporalne traktować będziemy jako zdarzenia, punkty w czasie czy też stany, dalej będzie to ten sam rodzaj danych. Również z uwagi na terminologię wykorzystywaną w przytaczanych pracach można odnieść wrażenie, iż opisywane są zgoła odmienne problemy. Cechą wspólną algorytmów indukcji reguł temporalnych jest to, że bazują one na relacjach, zdefiniowanych przez Allena oraz na algorytmie wyznaczania zbiorów częstych, zaproponowanym dla reguł asocjacyjnych przez Agrawala i Srikanta [3].

3.3. Algorytm ARED

Algorytm ARED nie jest typowym algorytmem umożliwiającym generowanie reguł temporalnych. W swoim założeniu działa on na zbiorze danych, pomiędzy którymi może nie występować żadna zależność temporalna. Jednak w szczególnym przypadku możliwe jest potraktowanie reguł generowanych przez ten algorytm, jako reguł temporalnych. Algorytm działa na „płaskim" zbiorze danych, który można przedstawić w postaci systemu informacyjnego lub tablicy decyzyjnej [23].

Definicja 12

Systemem informacyjnym S nazywamy $S = (X, A)$, gdzie X jest niepustym skończonym zbiorem obiektów, A jest niepustym skończonym zbiorem atrybutów, każdy atrybut $a \in A$ można potraktować, jako funkcję przypisującą dowolnemu obiektowi ze zbioru X wartość należącą do dziedziny V_a atrybutu a (a: $X \rightarrow V$a). Jeżeli w zbiorze atrybutów A zostanie określony atrybut $\{d\}$ zwany atrybutem decyzyjnym, wówczas system informacyjny S nazywany jest tablicą decyzyjną.

Za pomocą algorytmu ARED można określić zbiór tzw. reguł akcji [10] odzwierciedlających, w jaki sposób zmiana wartości jednego (lub kilku atrybutów) wpływa na zmianę wartości innego (innych) atrybutów (w szczególności może to być atrybut decyzyjny). Można przyjąć założenie, że warunki dotyczące zmiany wartości atrybutów i ich wpływu na inne wartości mogą być rozpatrywane w aspekcie temporalnym.

Reguły akcji można wyznaczać na dwa sposoby:

- korzystając z wyznaczonych wcześniej, na podstawie tablicy decyzyjnej, reguł decyzyjnych (klasyfikacyjnych) [24],
- korzystając bezpośrednio z tablicy decyzyjnej [10].

Pierwszy sposób polega na analizie otrzymanych reguł klasyfikacyjnych pod kątem możliwości zmiany wartości jednego z atrybutów warunkowych, w celu uzyskania nowej klasyfikacji. Drugi analizuje wprost zawartość tablicy decyzyjnej określając zmiany atrybutów warunkowych, które wpłyną na zmianę atrybutu decyzyjnego.

Definicja 13

Klasa decyzyjna jest to zbiór X_v zdefiniowany jako $X_v = \{ x \in X: d(x) = v \}$, gdzie $d(x)$ jest wartością atrybutu decyzyjnego obiektu x.

Definicja 14

Regułą akcji nazywamy regułę określającą możliwość zmiany przyporządkowania obiektu do klasy decyzyjnej poprzez zmianę jednej (lub więcej) wartości atrybutów warunkowych.

Sposób działania algorytmu ARED zostanie przybliżony na przykładowych danych wejściowych w postaci tablicy decyzyjnej, przedstawionej w tabeli 2. Pierwszym krokiem jest

utworzenie zbioru zawierającego tzw. ziarna wartości atrybutów. Określenie to – wprowadzone przez autorów algorytmu – oznacza obiekty z tablicy decyzyjnej, dla których występują poszczególne wartości atrybutów.

Tabela 2

Tablica decyzyjna

	a	b	c	d
x_1	a_2	b_1	c_1	d_1
x_2	a_2	b_2	c_3	d_1
x_3	a_1	b_1	c_1	d_2
x_4	a_1	b_1	c_2	d_1
x_5	a_2	b_2	c_2	d_3

Przykładowo ziarnami dla wszystkich wartości atrybutów a, b, c są zbiory:

$a_1^* = \{ x_3, x_4 \}$

$a_2^* = \{ x_1, x_2, x_5 \}$

$b_1^* = \{ x_1, x_3, x_4 \}$

$b_2^* = \{ x_2, x_5 \}$

$c_1^* = \{ x_1, x_3 \}$

$c_2^* = \{ x_4, x_5 \}$

$c_3^* = \{ x_2 \}$

Natomiast dla atrybutu decyzyjnego d otrzymano zbiory:

$d_1^* = \{ x_1, x_2, x_4 \}$

$d_2^* = \{ x_3 \}$

$d_3^* = \{ x_5 \}$

Następnym krokiem algorytmu jest określenie dwóch zbiorów τ i δ, aby sprawdzić możliwe przejścia pomiędzy wartościami atrybutów. Niech zbiór T będzie zbiorem możliwych połączeń pomiędzy powyższymi zbiorami atrybutów warunkowych i decyzyjnych. Jako poprawne połączenie definiujemy 2 zbiory, które mają przynajmniej jeden element wspólny. Przykładowo poprawnym połączeniem będzie $(a_2 \cdot d_1)$, z uwagi na wspólny element $\{ x_1 \}$. Wówczas zbiory τ i δ będą zdefiniowane zgodnie ze wzorem 2.

$$\tau = T \cdot d_1, d_1 \in V_d, (\forall p_1 \in T \cdot d_1)(\sup(p_1) \geq \lambda_1) \qquad (2)$$

$$\delta = T \cdot d_2, d_2 \in V_d, (\forall p_2 \in T \cdot d_2)(\sup(p_2) \geq \lambda_2)$$

Dla przedstawionej tablicy decyzyjnej zbiory τ i δ mają postać:

Tabela 3

Zbiory τ i δ

τ	δ
($a_1 \cdot d_1$)	($a_1 \cdot d_1$)
($a_1 \cdot d_2$)	($a_1 \cdot d_2$)
($a_2 \cdot d_1$)	($a_2 \cdot d_1$)
($a_2 \cdot d_3$)	($a_2 \cdot d_3$)
($b_1 \cdot d_1$)	($b_1 \cdot d_1$)
($b_1 \cdot d_2$)	($b_1 \cdot d_2$)
($b_2 \cdot d_1$)	($b_2 \cdot d_1$)
($b_2 \cdot d_3$)	($b_2 \cdot d_3$)
($c_1 \cdot d_1$)	($c_1 \cdot d_1$)
($c_1 \cdot d_2$)	($c_1 \cdot d_2$)
($c_2 \cdot d_1$)	($c_2 \cdot d_1$)
($c_2 \cdot d_3$)	($c_2 \cdot d_3$)
($c_3 \cdot d_1$)	($c_3 \cdot d_1$)

Kolejnym krokiem algorytmu jest określenie możliwych przejść pomiędzy poszczególnymi ziarnami. Poprawną tranzycją jest taka zmiana atrybutu warunkowego, która da określoną zmianę atrybutu decyzyjnego i występuje w tablicy decyzyjnej. Dla każdego możliwego przejścia obliczane są wartości wsparcia oraz ufności. Wszelkie tranzycje, które nie występują w tablicy decyzyjnej (są niepoprawne) bądź nie spełniają kryterium wsparcia minimalnego są odrzucane. Natomiast przejścia uznane za poprawne są wykorzystywane do tworzenia kolejnych reguł biorących pod uwagę coraz większą liczbę atrybutów. Przykład możliwych przejść został przedstawiony w tabeli 4.

Tabela 4

Dwuelementowe reguły akcji

$\tau \rightarrow \delta$	**supp**	**conf**
($a_1 \cdot d_2$) → ($a_2 \cdot d_1$)	1	0.5
($a_2 \cdot d_1$) → ($a_1 \cdot d_2$)	1	1
($c_1 \cdot d_2$) → ($c_2 \cdot d_1$)	1	1
($c_2 \cdot d_1$) → ($c_1 \cdot d_2$)	1	1
($c_2 \cdot d_3$) → ($c_3 \cdot d_1$)	1	1
($c_3 \cdot d_1$) → ($c_2 \cdot d_3$)	1	1

Jakość wyznaczonych reguł akcji zdefiniowana jest za pomocą pojęć ufności oraz wsparcia. Wsparcie reguły akcji p określonej, jako poprawne przejście pomiędzy $\tau \rightarrow \delta$ zdefiniowane jest jako liczba obiektów z tablicy decyzyjnej, mających wartości atrybutów, które występują w regule p. Natomiast ufność reguły akcji określa się następująco:

$$\mathrm{Conf} = \frac{\mathrm{supp}(p)}{\mathrm{supp}(\tau)} \tag{3}$$

Efektem działania algorytmu jest zbiór reguł akcji postaci:

$$a_2 \rightarrow a_1 \mapsto d_2 \rightarrow d_1$$

Znaczenie powyższej reguły jest następujące – jeśli wartość atrybutu warunkowego zmieni się z a_2 na a_1, wówczas decyzja powinna zmienić się z d_2 na d_1.

Przykładem zastosowania reguł akcji może być zapisana w postaci tablicy decyzyjnej analiza aplikacji do pracy, wyszczególniająca między innymi takie atrybuty warunkowe, jak: wykształcenie, oczekiwania finansowe, doświadczenie zawodowe oraz umiejętności interpersonalne. Atrybut decyzyjny przyjmowałby jedną z wartości, określającą ocenę kandydata w czterostopniowej skali: „kandydat niespełniający wymagań", „zadowalający kandydat", „pożądany kandydat", „idealny kandydat". Zastosowanie algorytmu ARED do analizy tak skonstruowanych danych pozwoliłoby na przykład na odkrycie reguły akcji mówiącej, iż jeśli wykształcenie kandydata zmieni się z „wyższego" na „wyższe z ukończonymi dodatkowymi kursami", wówczas decyzja o przyjęciu do pracy zmieni się z „zadowalający kandydat" na „idealny kandydat" [10].

4. Podsumowanie

Celem niniejszego artykułu było przybliżenie tematyki tworzenia reguł dla danych, które mają wymiar czasowy. Przedstawione zostały podstawowe pojęcia oraz algorytmy z zakresu odkrywania reguł temporalnych.

Obecne badania w zakresie odkrywania reguł temporalnych koncentrują się wokół dwóch tematów. Pierwszy kierunek to optymalizacja już istniejących algorytmów, poprzez zmianę wykorzystywanych struktur danych bądź modyfikację samych algorytmów pod kątem wydajności. W tym temacie głównym problemem staje się ograniczenie liczby odczytów z bazy danych bądź z dysku do minimum, co w przypadku dużych baz danych staje się zadaniem bardzo skomplikowanym.

Drugim kierunkiem badań są próby definiowania kryteriów, które pozwoliłyby ograniczać zbiory generowanych reguł tylko do tych najbardziej interesujących. Klasyczne kryteria, takie jak minimalne wsparcie czy pewność generowanych reguł dla niektórych zastosowań okazują się niewystarczające. Pojęcie „być regułą interesującą" rozpatrywane jest tutaj w takim samym kontekście, jak w przypadku reguł asocjacyjnych [30], jednakże przeniesienie tzw. miar stopnia, w jakim dana reguła jest interesująca (ang. *interestingness measures*), stosowanych na granicy reguł asocjacyjnych, do reguł temporalnych wymaga dodatkowych badań. Ponadto, również dla reguł asocjacyjnych problem doboru odpowiedniej miary dla danego zbioru danych i zadania jest w dalszym ciągu problemem otwartym [31].

Wydaje się, że zupełnie nierozpoznanym problemem jest problem poszukiwania ciągów (łańcuchów) reguł temporalnych. Ciągiem reguł temporalnych nazywać będziemy taki ciąg, w którym konkluzje reguły poprzedzającej mają wpływ na przesłankę reguły następnej (lub

definiują ją w sposób bezpośredni). Spośród wszystkich możliwych ciągów reguł przy ustalonym stanie początkowym szczególnie interesujące będą te ciągi, które prowadzić będą do stanów pożądanych przez użytkownika.

Prace nad definiowaniem miar oceny jakości reguł temporalnych, algorytmami filtracji reguł wykorzystującymi te miary oraz poszukiwaniem ciągów reguł temporalnych o pożądanych własnościach będą przedmiotem dalszych badań autorów nad algorytmami indukcji reguł temporalnych.

BIBLIOGRAFIA

1. Winarko E., Roddick J. F.: ARMADA – An algorithm for discovering richer relative temporal association rules from interval – based data. 2007.
2. Antunes C. M., Oliveira A. L.: Temporal Data Mining: an overview. 2001.
3. Srikant R., Agrawal R.: Mining generalized association rules. 1999.
4. Winarko E.: The Discovery and Retrieval Of Temporal Rules in Interval Sequence Data. 2007.
5. Zhu H., Huang W., Zheng H.: Method for Discovering Actionable Rule. 2007.
6. Srikant R., Agrawal R., Vu Q.: Mining Association Rules with Item Constraints. 1997.
7. Hilderman R. J., Hamilton H. J.: Knowledge Discovery and Interestingness Measures: A Survey. 2007.
8. Raś Z. W., Tzacheva A. A., Tsay L.: Discovery of Interesting Action Rules. 2005.
9. Lee Y. J, Lee J. W., Chai D. J., Hwang B. H., Ryu K. H.: Mining temporal interval relational rules from temporal data. 2009.
10. Im S., Raś Z. W.: Action Rule Extraction from A Decision Table: ARED. 2008.
11. Böhlen M. H., Busatto R. R., Jensen C. S.: Point-versus interval-based temporal data models. 1998.
12. Allen J.: Maintaining knowledge about temporal intervals. 1983.
13. Höppner F.: Learning Temporal Rules from State Sequences. 2001.
14. Lin M. Y., Lee S. Y.: Fast discovery of sequential patterns by memory indexing. 2002.
15. Srikant R., Agrawal R.: Fast Algorithms for Mining Association Rules. 2007.
16. Houstma M., Swami A.: Set-oriented mining of associations rules. 1993.
17. Cohen W.: Fast Effective Rule Induction. Proceedings of the 12th International Conference on Machine Learning, 1995.
18. Furnkranz J., Widmer G.: Incremental Reduced Error Pruning. Proceedings of the Eleventh International Conference of Machine Learning, New Brunswick, NJ 1994.

19. Grzymała-Busse J. W., Ziarko W.: Data mining based on rough sets, [in:] Wang J. (ed.): Data Mining Opportunities and Challenges. IGI Publishing, Hershey, PA, USA 2003, s. 142÷173.
20. Kaufman K. A., Michalski R. S.: Learning in Inconsistent World, Rule Selection in STAR/AQ18. Machine Learning and Inference Laboratory Report P99-2, 1999.
21. Michalski R. S., Carbonell J. G., Mitchel T. M.: Machine Learning, Vol. I, Los Altos: Morgan Kaufmann, 1983.
22. Kubat M., Bratko I. Michalski R. S. (eds.): Machine Learning and Data Mining: Methods and Applications. John Wiley and Sons, 1998.
23. Pawlak Z.: Rough sets: Theoretical aspects of reasoning about data. Kluwer, Dordrecht 1991.
24. Skowron A., Rauszer C.: The Discernibility Matrices and Functions in Information systems. Słowiński R. (ed.): Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory. Kluwer, Dordrecht 1992, s. 331÷362.
25. Agrawal R., Srikant R.: Mining Quantitative Association Rules in Large Relational Tables. Proceedings of SIGMOD Conference, Montreal 1996.
26. Furnkranz J.: Separate-and-conquer rule learning. Artificial Intelligence Review 13, 1999, s. 3÷54.
27. Quinlan J. R.: Learning with continuous classes. Proc. of the International Conference on Artificial Intelligence (AI`92, World Scientific), Singapore 1992.
28. Sikora M., Gruca A.: Quality improvement of rule-based gene group descriptions using information about GO terms importance occurring in premises of determined rules. Int. J. Appl. Math. Comput. Sci., 201, Vol. 20, No. 3, s. 555÷570.
29. Sikora M., Wróbel Ł.: Application of rule induction algorithms for analysis of data collected by seismic hazard monitoring systems in coal mines. Archives of Mining Sciences, Vol. 55, No.1, 2010, s. 91÷114.
30. Guillet F., Hamilton H.J.: Quality measures in data mining. Studies in computational intelligence. Vol. 43. Springer-Verlag, 2007.
31. Suzuki E.: Pitfalls for categorization of objective interestingness measures for rule discovery. Studies in Computational Intelligence 127, Springer-Verlag, 2008, s. 383÷395.

Recenzenci: Prof. dr hab. inż. Jerzy Klamka
Prof. dr hab. inż. Alicja Wakulicz-Deja

Wpłynęło do Redakcji 30 grudnia 2010 r.

Abstract

Nowadays research in the field of correlations in temporal data becomes increasingly important. There are multiple reasons for this. On the one hand there are corporations interested in collecting as much data as possible. Such large volumes of knowledge, after processing will serve great help for marketing department. On the other hand there is medical industry and the whole world of science concerned with improving our health. If these two subjects would be willing to sustain their interest in this field, we can be sure that there is a lot of good to come.

This paper shows basic definitions and algorithms of temporal rules induction (section 1 and 2). It also shows, how these algorithms might be used in real life and how to read discovered rules (section 3). In the last section (summary) the future work directions were discussed. The knowledge contained in this paper is a good starting-point to independent research.

Adresy

Natalia OGIEGŁO: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-100 Gliwice, Polska, natalia.ogieglo@gmail.com.
Marek SIKORA: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-100 Gliwice, Polska, marek.sikora@polsl.pl.