STUDIA INFORMATICA 2010
Volume 31 Number 2A (89)

Krzysztof CZAJKOWSKI
Politechnika Krakowska, Katedra Teleinformatyki,
Wydział Fizyki, Matematyki i Informatyki Stosowanej

REGUŁY DECYZYJNE I BAZY DANYCH W KLASYFIKACJI STRON INTERNETOWYCH

Streszczenie. Artykuł dotyczy zastosowania teorii zbiorów przybliżonych w klasyfikacji stron internetowych. W pracy zaproponowano podejście integrujące elementy teorii zbiorów przybliżonych z bazami danych, którego celem jest zwiększenie wydajności oraz przetwarzanie danych bezpośrednio w miejscu ich przechowywania. Artykuł dotyczy implementacji w środowisku bazodanowym algorytmów selekcji atrybutów nieusuwalnych i reduktów względnych oraz wyznaczania reguł decyzyjnych. Celem jest klasyfikacja stron internetowych w oparciu o reguły decyzyjne oraz zbiór cech opisujących poszczególne dokumenty.

Słowa kluczowe: zbiory przybliżone, redukcja atrybutów, redukty względne, reguły decyzyjne, klasyfikacja stron internetowych

DECISION RULES AND DATABASES IN WEB PAGES CLASSIFICATION

Summary. This paper concerns applying of rough sets theory to web pages classification. In this work the approach integrating elements of rough sets theory with databases was proposed, which the aim is improving of the efficiency as well as processing of data in the place of them store. The paper describes implementations of algorithms of core attributes and reducts selection as well as decision rules determining. The aim is web pages classifications on the basis of decision rules and the set of features describing individual web pages.

Keywords: rough sets, attributes reduction, reducts, decision rules, web pages classification.

1. Wstęp

Nieustanny wzrost ilości treści dostępnych w sieci Internet powoduje, że coraz trudniejsze staje się ich przeszukiwanie. Znalezienie interesujących materiałów wymaga przeglądania ogromnych ilości informacji, których wciąż przybywa. Klasyfikacja i katalogowanie stron nabiera coraz większej wagi. Proces ten musi być sprawnie i skutecznie realizowany tak, aby nadążyć za stałym wzrostem liczby stron. W wielu przypadkach użytkownicy poszukujący konkretnych treści otrzymują jako rezultat poszukiwania tysiące stron i zmuszeni są poświęcić wiele czasu na ich przeglądanie. W dużych repozytoriach występuje stały wzrost liczby dokumentów, które powinny podlegać klasyfikacji, a realizacja tego procesu w sposób manualny jest niepraktyczna. Odpowiedzią na te potrzeby mogą być rozwiązania dostarczające możliwości automatycznego klasyfikowania dokumentów, wspierając w ten sposób proces skutecznego ich wyszukiwania. W procesie klasyfikacji systemy automatyczne bazują na modelach określonych na etapie uczenia bazującego na przykładowych zbiorach.

Artykuł dotyczy automatycznej klasyfikacji stron internetowych wykorzystującej redukcję atrybutów opisujących takie dokumenty. Redukcja atrybutów pozwala wyeliminować te atrybuty, które z punktu widzenia klasyfikacji są nieistotne. Zmniejszenie liczby właściwości opisujących dokumenty, zbieranych w celu ich prawidłowej klasyfikacji, pozwala uprościć proces pozyskiwania tych właściwości, zredukować pamięć potrzebną do ich przechowywania oraz zmniejszyć złożoność całego procesu klasyfikacji. Do redukcji atrybutów wykorzystano elementy teorii zbiorów przybliżonych w celu wyznaczenia zbioru atrybutów nieusuwalnych (rdzenia), reduktów względnych (zbiorów atrybutów o takich samych zdolnościach klasyfikacyjnych jak pełny zbiór atrybutów warunkowych) oraz reguł decyzyjnych.

Algorytmy wyznaczające atrybuty nieusuwalne i redukty cechują się zazwyczaj dużą złożonością obliczeniową. W większości rozwiązań wykorzystujących zbiory przybliżone operacje te są wykonywane z wykorzystaniem zwykłych plików „płaskich” i nie korzystają, z o wiele bardzie wydajnych w tym przypadku, systemów bazodanowych. Dodatkową wadą podejść tradycyjnych (wykorzystujących dedykowane aplikacje, tworzone w celu przeprowadzania operacji związanych z teorią zbiorów przybliżonych) jest konieczność przenoszenia danych (często dużej ich ilości) z miejsca ich przechowywania (często są to bazy danych) do osobnych aplikacji. Najkorzystniejszym rozwiązaniem wydaje się być wykonywanie operacji redukujących atrybuty wewnątrz systemu bazodanowego.

Pojawiły się propozycje integracji zbiorów przybliżonych z relacyjnym systemem baz danych, takie jak Rough Set Data Miner [3]. Stosuje ono wbudowane zapytania SQL w celu wykorzystania zalet technologii bazodanowych. Warte uwagi jest podejście zaproponowane w [4], oparte na zastosowaniu algebry relacyjnej do wyznaczania rdzenia i reduktów. Wykorzystując operacje typowe dla baz danych, takie jak zliczanie oraz projekcja, poprawia ono

wydajność całego procesu. Dodatkowo podejście to umożliwia wykorzystanie efektywnych rozwiązań bazodanowych, takich jak na przykład indeksowanie. Coraz doskonalsze optymalizatory zapytań stosowane w bazach danych pozwalają zredukować koszt dostępu do pamięci dyskowej i dobrze sobie radzą w przypadku wielkich ilości danych.

W artykule przedstawiono metodę redukcji atrybutów oraz generowania reguł decyzyjnych w teorii zbiorów przybliżonych, z wykorzystaniem bazy danych i algebry relacyjnej. Zaprezentowano implementację wykazującą zalety takiego podejścia w klasyfikacji stron internetowych.

2. Zbiory przybliżone w ujęciu bazodanowym

W teorii zbiorów przybliżonych dane można zaprezentować w postaci tablicy decyzyjnej, w której wiersze odpowiadają obiektom, a kolumny atrybutom tych obiektów. Formalnie tablicą decyzyjną nazywamy uporządkowaną piątkę [10]:

$$DT=(U,C,D,V,f), \tag{1}$$

gdzie: $C, D \subset A; C \neq \emptyset, D \neq \emptyset; C \cup D = A; C \cap D = \emptyset$. U jest niepustym skończonym zbiorem zwanym uniwersum tablicy decyzyjnej, elementy tego zbiory nazywamy obiektami. C to zbiór atrybutów warunkowych, a D – zbiór atrybutów decyzyjnych. f nazywamy funkcją decyzyjną. $V = \bigcup_{a \in A} V_a$, przy czym V_a nazywamy dziedziną atrybutu $a \in A$.

Zakładając, że $B \subseteq C$, zbiór atrybutów Q ($\subseteq B$) nazywamy reduktem zbioru atrybutów B względem atrybutu decyzyjnego d (przy założeniu, że zbiór atrybutów decyzyjnych jest jednoelementowy), gdy zbiór atrybutów Q jest niezależny oraz $IND(B,d)=IND(Q,d)$. $IND(B,d)$ to relacja nierozróżnialności względem decyzji d, generowana przez zbiór atrybutów B. Redukt jest najmniejszym zbiorem atrybutów, przy którym zostaje zachowana dotychczasowa klasyfikacja (rozróżnialność) obiektów. W tabeli decyzyjnej może występować więcej niż jeden redukt.

Rdzeniem, oznaczanym jako $CORE(B,d)$, nazywamy zbiór wszystkich atrybutów niezbędnych (nieusuwalnych) w zbiorze B ze względu na atrybut decyzyjny d. Rdzeń stanowi część wspólna wszystkich reduktów.

W prezentowanym podejściu podstawowe pojęcia z teorii zbiorów przybliżonych zostały zdefiniowane z wykorzystaniem pojęć z teorii baz danych. Celem takiego rozwiązania jest zastosowanie mechanizmów istniejących w bazach danych dla uzyskania większej wydajności w wyznaczaniu atrybutów nieusuwalnych oraz reduktów względnych.

Wszystkie atrybuty nieusuwalne są niezbędnymi elementami każdego reduktu [1], tak więc kluczowym problemem jest możliwość efektywnego wyszukiwania takich atrybutów.

W podejściu tradycyjnym popularną metodą jest konstruowanie macierzy decyzyjnej, a następnie przeszukiwanie wszystkich pozycji w takiej macierzy w celu znalezienia pozycji, które mają tylko jeden atrybut [6]. Metoda ta cechuje się niesatysfakcjonującą wydajnością, szczególnie w przypadku systemów gromadzących bardzo duże ilości danych. Z kolei inne metody, niewykorzystujące macierzy decyzyjnej, np. [8], mają złożoność obliczeniową *O(mnlogn)* (gdzie *n* to liczba wierszy, a *m* jest liczbą atrybutów).

Prezentowane podejście ma złożoność *O(mn)* [4] i jest realizowane bez potrzeby wyznaczania dolnego i górnego przybliżenia.

Oznaczając operację relacyjną zliczania (*Count*) poprzez *Card*, operację projekcji (*Projection*) jako $\prod$, a zbiór atrybutów decyzyjnych jako *D*, możemy zapisać, że $C_j \in C$ jest atrybutem nieusuwalnym, jeżeli zachodzi (2).

$$Card\left(\Pi\left(C - C_j + D\right)\right) \neq Card\left(\Pi\left(C - C_j\right)\right) \tag{2}$$

Jeżeli liczba wierszy, które można rozróżnić za pomocą aktualnych wartości atrybutów warunkowych (z pominięciem danego atrybutu) oraz atrybutu decyzyjnego, jest różna od liczby rozróżnialnych wierszy za pomocą tego samego podzbioru atrybutów warunkowych, ale bez udziału atrybutu decyzyjnego, oznacza to, że usunięcie spośród atrybutów warunkowych tego konkretnego atrybutu powoduje częściową utratę możliwości identyfikowania obiektów – atrybut ten jest nieusuwalny.

Wynika z tego, że można ustalić, czy atrybut jest atrybutem nieusuwalnym za pomocą podstawowych operacji języka SQL. Niezbędne jest wykonanie tylko dwóch projekcji: jednej na atrybutach C-C_j+D, a drugiej na atrybutach C-C_j. Jeżeli liczba wierszy w obu projekcjach jest różna, to dany atrybut jest atrybutem nieusuwalnym, w przeciwnym przypadku (3) jest on zbędny (nie ma utraty informacji, po usunięciu atrybutu – każdy obiekt może być sklasyfikowany w ten sam sposób, bez względu na to czy atrybut jest obecny, czy też nie).

$$Card\left(\Pi\left(C - C_j + D\right)\right) = Card\left(\Pi\left(C - C_j\right)\right) \tag{3}$$

W celu wyznaczenia reduktów zbioru atrybutów warunkowych można zdefiniować stopień zależności pomiędzy reduktem a zbiorem atrybutów decyzyjnych – jak w (4).

$$K(REDU, D) = \frac{Card\left(\Pi\left(REDU + D\right)\right)}{Card\left(\Pi\left(C + D\right)\right)} \tag{4}$$

Podzbiór zbioru atrybutów warunkowych $REDU(\subseteq C)$ jest reduktem zbioru atrybutów warunkowych *C* ze względu na zbiór atrybutów decyzyjnych *D*, jeżeli jest minimalnym zbiorem atrybutów, które mają taką samą zdolność klasyfikowania, jak cały zbiór atrybutów warunkowych (5) i (6).

$$K(REDU, D) = K(C, D) \tag{5}$$

oraz

$$K(REDU, D) \neq K(R', D), \forall R' \subset REDU \tag{6}$$

Jeżeli współczynnik *K(REDU,D)* jest równy 1, to zbiór atrybutów decyzyjnych *D* całkowicie zależy od zbioru atrybutów warunkowych *C*. Przy *K(REDU,D)* < 1 mówimy o zależności częściowej (o stopniu *K(REDU,D)*) [2].

Idea redukcji atrybutów może być uogólniona przez wprowadzenie pojęcia istotności atrybutu. Nie jest stosowana dwuwartościowa skala: znaczący i nieznaczący. Zamiast tego konkretnym atrybutom przypisane są wartości z przedziału [0,1], wyrażające istotność tych atrybutów dla tablicy decyzyjnej. Istotność atrybutu może zostać określona przez efekt wywołany jego usunięciem z tablicy decyzyjnej [2].

Współczynnik istotności dla danego atrybutu C_j ze zbioru atrybutów *C* definiowano jako:

$$Merit(C_j, C, D) = 1 - \frac{Card(\Pi(C - C_j + D))}{Card(\Pi(C + D))} \tag{7}$$

Wartość tego współczynnika reprezentuje wkład wnoszony przez atrybut C_j do zależności pomiędzy *C* (atrybutami warunkowymi) a *D* (atrybutami decyzyjnymi). Większa wartość tego współczynnika oznacza, że jest on istotniejszy w rozpatrywanej tablicy decyzyjnej.

3. Wyszukiwanie atrybutów nieusuwalnych i reduktów względnych

W prezentowanym podejściu zakładamy, że w tabeli decyzyjnej nie występują krotki sprzeczne (niespójne), to jest takie, które posiadając identyczne wartości atrybutów warunkowych, mają inne wartości atrybutów decyzyjnych (należą do innej klasy). Przedstawiany algorytm [4] opiera się na operacjach istniejących w systemie bazodanowym, bez konieczności obliczania dolnego i górnego przybliżenia i cechuje się większą wydajnością od podejścia tradycyjnego.

```
Algorytm 1: Core Attribute Algorithm
Wejście: tablica decyzyjna T (C,D)
Wyjście: Core - zbiór atrybutów nieusuwalnych (rdzeń) dla tablicy T

   1. Core = ∅
   2. For each A∈C {
      If Card(Π(C − A + D)) ≠ Card(Π(C − A))
      Then Core = Core ∪ A
   }
```

Można zapisać algorytm 1 jako operację: $Card(\Pi(X))$, gdzie *X* może być *C*, *C-A*, *C-A+D*. Wykorzystując język SQL operacja ta może zostać wyrażona za pomocą polecenia SELECT:

```
SELECT COUNT(*) FROM (SELECT DISTINCT X FROM T);
```

Jak udowodniono w [4], algorytm 1 może być zaimplementowany przy złożoności obliczeniowej O(mn), gdzie m to liczba atrybutów, natomiast n jest liczbą wierszy (zakładając wykorzystanie indeksów). Prawidłowość działania algorytmu potwierdzają wyniki uzyskane w systemach Rosetta [12] i RSES [13].

W powyższych rozważaniach zakładamy niewystępowanie danych niespójnych. W rzeczywistości takie niespójności (zaszumienia) pojawiają się i powinny zostać wyeliminowane. W celu wykrycia, czy w zbiorze danych istnieją niespójności wystarczy sprawdzić, czy zachodzi równość $Card(\Pi(C)) = Card(\Pi(C + D))$. Jeżeli równość nie jest spełniona, oznacza to, że istnieją dane niespójne. Do ustalenia, które wiersze są niespójne, wystarczy zapytanie:

```
SELECT * FROM T U WHERE EXISTS (SELECT * FROM T V WHERE (U.C=V.C) AND
(U.D<>V.D));
```

W ten sposób niespójności w danych mogą być wyeliminowane w czasie *O(n)*, przy *n* równym ilości wierszy w tabeli.

W tabeli decyzyjnej występować mogą dwa rodzaje atrybutów zbędnych z punktu widzenia ich znaczenia dla opisu obiektów. Pierwszym z nich są atrybuty nieistotne, takie jak różnego typu identyfikatory. Drugi rodzaj stanowią atrybuty nadmiarowe w stosunku do pozostałych atrybutów. Niektóre atrybuty są niezbędne dla poprawnej klasyfikacji, ale niekoniecznie wszystkie jednocześnie. W celu zredukowania liczby atrybutów do minimum należy przeprowadzić selekcję atrybutów, polegającą na wybraniu podzbioru zbioru atrybutów, który składać się będzie tylko z tych atrybutów, które są istotne, przy jednoczesnym wyeliminowaniu atrybutów nieznaczących. Pozwala to zminimalizować czas w procesie dalszego przetwarzania informacji.

W tabeli decyzyjnej może występować więcej niż jeden redukt. Znalezienie wszystkich reduktów w tabeli decyzyjnej jest zadaniem *NP*-trudnym [9]. Istnieją jednak zastosowania, w których nie występuje konieczność znalezienia wszystkich reduktów, wystarczy znalezienie jednego z nich. Częścią wspólną wszystkich reduktów jest zbiór atrybutów nieusuwalnych (rdzeń).

$$CORE(C) = \bigcap REDU(C) \tag{8}$$

Każdy element rdzenia stanowi część każdego reduktu, dlatego też zbiór atrybutów nieusuwalnych jest najważniejszym podzbiorem zbioru atrybutów warunkowych [1]. Żaden element z tego zbioru nie może zostać usunięty bez skutków w postaci mniejszych zdolności klasyfikacyjnych zbioru atrybutów warunkowych. Dalej zamieszczony jest algorytm (zachłanny) wyznaczania reduktów w procesie selekcji i eliminacji.

```
Algorytm 2: Wyznaczanie minimalnego podzbioru atrybutów (reduktu)
Wejście: Tabela decyzyjna T (C,D)
Wyjście: Minimalny podzbiór zbioru atrybutów (REDU)
```

- Uruchom Algorytm 1 w celu wyznaczenia atrybutów kluczowych CORE
- REDU = CORE
- AR = C – CORE
  {Forward selection}
- WHILE K(REDU,D) ≠ K(C,D)
  - For each A∈AR oblicz współczynnik istotności (merit)
  - Posortuj malejąco atrybuty w AR w oparciu o wartość merit
  - Wybierz atrybut C_j z największą wartość merit (gdy kilka atrybutów ma taką samą wartość merit, wybierz ten z mniejszą liczbę kombinacji z atrybutami w REDU)
  - REDU = REDU ∪ C_j, AR = AR – C_j
- ENDWILE
  {Backward elimination}
- N = ||REDU||
- FOR j=0 to N-1 DO
  - IF $a_j \notin$ CORE THEN oblicz K(REDU-a_j,D)
  - IF K(REDU-a_j,D) = K(REDU,D) THEN REDU = REDU - a_j
- ENDFOR

Pierwszym etapem działania algorytmu 2 jest wykonanie operacji algorytmu 1. Ustalenie zbioru CORE i usunięcie jego elementów z dalszego przetwarzania jest korzystne z tego względu, iż wyznaczanie jego elementów jest operacją mniej złożoną, niż operacje służące ustaleniu atrybutów REDU.

Prezentowany algorytm cechuje się większą wydajnością niż inne podejścia, ponieważ przetwarza zbiory dyskowe obsługiwane przez system bazodanowy i wszystkie działania służące ustalaniu niezbędności elementu w zbiorze atrybutów decyzyjnych (z punktu widzenia możliwości klasyfikacyjnych) oraz wyliczanie współczynników istotności dla poszczególnych atrybutów odbywa się za pomocą operacji pod kontrolą systemu zarządzania relacyjnymi bazami danych. Dodatkowo wszystkie operacje wykonywane są w miejscu przechowywania danych, bez konieczności przenoszenia danych do innych środowisk.

4. Wyznaczanie reguł decyzyjnych

Koncepcji klasyfikatorów w przypadku teorii zbiorów przybliżonych odpowiada koncepcja reduktów względnych. Na podstawie reduktu względnego możliwe jest wygenerowanie tabeli (*reduct table*). Następnie na podstawie tej tabeli możliwe jest zbudowanie klasyfikatora składającego się z reguł decyzyjnych. Reguła decyzyjna jest kombinacją wartości wybranych atrybutów warunkowych, takich, dla których zbiór wszystkich przypadków (obiektów) posiadających takie wartości należy do tej samej klasy (ma taką samą wartość atrybutu decyzyjnego). Możemy zapisać regułę jako implikację:

$$r : G_{i1} = V_{i1} \bigcap G_{i2} = V_{i2} \bigcap \ldots G_{ik} = V_{ik} \rightarrow D = d_i \tag{9}$$

Celem procesu generowania reguł jest uzyskanie reguł możliwie najbardziej ogólnych. W tym celu należy z danej reguły usunąć tak wiele atrybutów warunkowych, jak to możliwe

bez utraty poprawności reguły. W ten sposób uzyskujemy reguły, które reprezentują najbardziej ogólne wzorce występujące w zbiorze danych.

W generalizacji reguł decyzyjnych istotne są dwa pojęcia: nadmiarowość oraz niespójność. Nadmiarowość ma miejsce w sytuacji, gdy dla dwóch reguł decyzyjnych wartość atrybutów decyzyjnych jest taka sama, a zbiór atrybutów warunkowych jednej z nich jest podzbiorem zbioru atrybutów warunkowych drugiej reguły (a wartości pokrywających się atrybutów są takie same). Wówczas druga reguła zawiera się w regule pierwszej. Dla dokonania prawidłowej klasyfikacji wystarczy, aby w konkretnym rozpatrywanym przypadku spełnione były warunki określone w regule pierwszej, reguła druga jest więc zbędna. Do niespójności natomiast dochodzi w sytuacji, gdy dla dwóch reguł decyzyjnych zbiór atrybutów warunkowych jednej z nich jest podzbiorem zbioru atrybutów warunkowych drugiej reguły lub są to takie same zbiory (i wartości pokrywających się atrybutów są takie same), a wartości atrybutów decyzyjnych są różne. W takie sytuacji dwie reguły, bazując na tych samych przesłankach, dają sprzeczne decyzje.

Cały proces generowania reguł decyzyjnych ma więc na celu znalezienie zestawu takich reguł, które są możliwie ogólne, nie zawierają reguł nadmiarowych ani reguł sprzecznych. Po wyznaczeniu reduktu możliwe jest utworzenie tablicy reduktu, która składa się tylko z wartości atrybutów tworzących redukt. Każdy wiersz w takiej tablicy jest regułą decyzyjną, którą należy poddać procesowi generalizacji tak, aby dla każdej z nich usunąć wszystkie zbędne wartości. Należy również usuwać reguły nadmiarowe i nie dopuścić do powstawania reguł sprzecznych. W ten sposób możliwe jest uzyskanie skutecznego klasyfikatora.

Kolejność rozpatrywania poszczególnych atrybutów warunkowych w danej regule decyzyjnej (pod kątem ich ewentualnego usunięcia) determinuje możliwość uzyskania możliwie najbardziej ogólnej reguły. Dla każdej reguły, posiadającej n atrybutów warunkowych, liczba możliwych podzbiorów takiego zbioru wynosi 2^n, tak więc rozpatrywanie wszystkich możliwych kombinacji jest niepraktyczne. Dla uproszczenia algorytmu możliwe jest określenie dodatkowego parametru SIG dla każdego atrybutu. Parametr ten będzie określał znaczenie danego atrybutu [5]. Proces usuwania atrybutów będzie przeprowadzany w kolejności rosnących wartości tego parametru (na początku przeprowadzana będzie próba usunięcia atrybuty o znaczeniu najmniejszym, a na końcu o znaczeniu największym). Parametr SIG dla danej wartości atrybutu warunkowego $C_{ik}=V_{ik}$ można zapisać w postaci:

$$SIG(C_{ik}=V_{ik})=P(C_{ik}=V_{ik})*(P(D=d_i|C_{ik}=V_{ik})-P(D=d_i)),$$

gdzie $P(C_{ik}=V_{ik})$ to prawdopodobieństwo wystąpienia wartości V_{ik} dla atrybutu C_{ik}, a $P(D=d_i|C_{ik}=V_{ik})$ to prawdopodobieństwo warunkowe wystąpienia klasy $D=d_i$ – pod warunkiem zaistnienia wartości atrybutu warunkowego $C_{ik}=V_{ik}$.

W celu uniknięcia generowania zbyt wyspecyfikowanych reguł możliwe jest wykorzystanie testu Laplace'a. Umożliwia to eliminację wielu reguł pokrywających niewielką liczbę przypadków. Preferowane są bardziej ogólne reguły.

$$Laplace = (n_c + 1)/(n_{total} + k), \tag{10}$$

gdzie k to liczba klas, n_c to liczba krotek przewidywanej klasy c pokrywanych przez tę regułę, n_{total} to całkowita liczba krotek pokrywanych przez tę regułę.

Poniżej zaprezentowany jest algorytm generujący zestaw możliwie najbardziej ogólnych reguł na podstawie tablicy decyzyjnej oraz reduktu (wyznaczonego algorytmem 2) [5].

```
Algorytm 3: Wyznaczanie zestawu najbardziej ogólnych reguł
Wejście: Tablica decyzyjna T(C,D), redukt REDU
Wyjście: Zestaw najbardziej ogólnych reguł MG_Rules
```

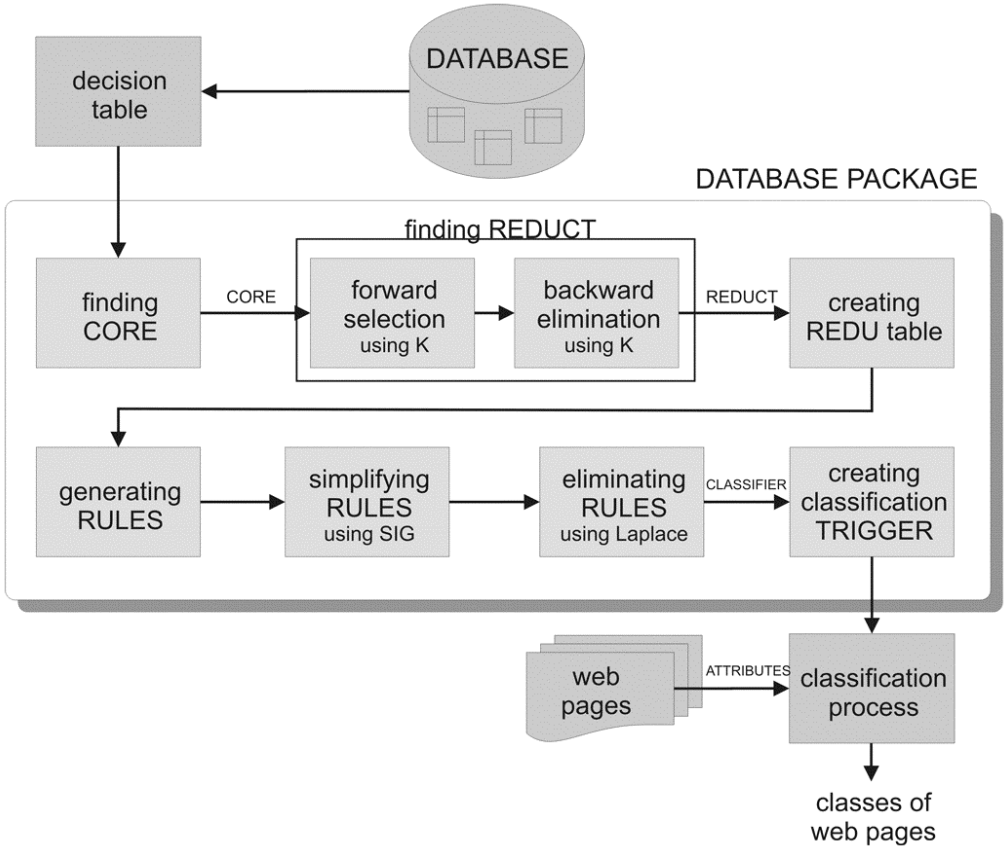
- Wygeneruj tabelę reduktu REULES za pomocą projekcji reduktu REDU oraz atrybutu decyzyjnego D na tablicy decyzyjnej T(C,D)
- Zbiór reguł MG_Rules = ∅
- For each $C_{ik}=V_{ik} \in$ RULES oblicz wartość parametru znaczenia SIG
- For each $r_i \in$ RULES uprość każdą krotkę:
  - Posortuj zbiór atrybutów warunkowych w regule r_i bazując na SIG
  - For each $C_{ik}=V$
    - Usuń wartość z r_i
    - Jeżeli r_i jest niespójne z dowolną inną regułą wówczas przywróć usuniętą wartość
  - Usuń wszystkie reguły z MG_Rules, które logicznie zawierają się w regule r_i
  - MG_Rules = MG_Rules $\cup$ r_i
- For each $r_i \in$MG_Rules oblicz wartość Laplace(r_i)
- For each $r_i \in$MG_Rules If Laplace(r_i)<threshold Then MG_Rules=MG_Rules-r_i

5. Implementacja

Jako środowisko do zaimplementowania omawianych rozwiązań wybrany został system Oracle. Możliwe jest oczywiście wykorzystanie innych systemów bazodanowych, w tym również typu Open Source. Wybór środowiska Oracle podyktowany był kilkoma przyczynami. Po pierwsze system Oracle jest bardzo wydajnym środowiskiem, dysponującym zaawansowanymi mechanizmami optymalizacji operacji. Po drugie, w tym środowisku dostępne są bardzo wygodne i elastyczne rozwiązania usprawniające proces tworzenia tego typu rozwiązań. Po trzecie i bardzo istotne, jedną z zalet proponowanego systemu jest analizowanie danych w miejscu ich przechowywania, co powoduje, że wybór środowiska do implementacji powinien być podyktowany powszechnością wykorzystania danego systemu – stąd też wybór systemu Oracle.

Tablica decyzyjna reprezentowana jest w bazie danych jako tabela (lub perspektywa, jeżeli dane pochodzą z kilku powiązanych ze sobą tabel), której kolumny reprezentują poszczególne atrybuty warunkowe i decyzyjne, natomiast kolejne obiekty reprezentowane są

w postaci rekordów. Operacje pozwalające określić istotność poszczególnych atrybutów warunkowych pod względem klasyfikowania obiektów realizowane są za pomocą wyrażeń języka SQL. Odpowiednie zapytania są generowane dynamicznie, stosownie do listy aktualnie rozpatrywanych, na danym etapie pracy algorytmu, atrybutów.



Rys. 1. Schemat działania pakietu
Fig. 1. Package working schema

Algorytmy zostały zapisane jako zestaw procedur i funkcji w języku PL/SQL. Wszystkie procedury i funkcje zebrane zostały w ramach jednego pakietu (rys. 1). Najważniejszymi funkcjami są: *findCORE*, *findREDU* oraz *findRULES*. Pierwsza z nich służy do określania zbioru atrybutów nieusuwalnych (rdzenia) ze zbioru atrybutów warunkowych, druga ma za zadanie wyznaczenie reduktu, ostatnia odpowiada za generowanie zbioru reguł decyzyjnych. Każda z funkcji jako argument otrzymuje nazwę tabeli przechowującej dane z tablicy decyzyjnej. Wszystkie pozostałe funkcje, mające charakter pomocniczy, zostały zadeklarowane w ciele pakietu i dostępne są tylko z poziomu jego wnętrza. Zarówno liczba jak i nazwy konkretnych kolumn są ustalane w trakcie działania programu na podstawie metadanych przechowywanych w bazie danych i dostępnych poprzez perspektywy systemowe. Polecenia języka SQL, realizujące poszczególne operacje, są wykonywane jako tzw. dynamiczny SQL. Dzięki takiemu rozwiązaniu nie jest konieczna znajomość składni w trakcie tworzenia funkcji. W ten sposób prezentowany pakiet może działać dla dowolnej tablicy decyzyjnej, której

struktura jest określana przez jednostki pakietu dopiero w chwili jej przetwarzania. Szersze omówienie implementacji dotyczącej znajdowania rdzenia oraz reduktów względnych zostało przedstawione w [15] i [16]. Po wyznaczeniu reduktu tworzona jest tabela zawierająca tylko wartości atrybutów wchodzących w jego skład. W oparciu o tę tabelę tworzone są reguły, które poddawane są procesowi upraszczania z wykorzystaniem wyznaczonego parametru znaczenia danego atrybutu (w ten sposób kolejność weryfikacji atrybutów zależy od ich znaczenia) i ewentualnie usuwane, jeżeli zawierają się w innych regułach. W kolejnym etapie usuwane są te reguły, dla których wyliczona wartość współczynnika Laplace'a jest niższa od zadanego progu (wartość progu dobierana jest eksperymentalnie). Ostatecznie uzyskiwany jest zestaw reguł tworzący klasyfikator.

Klasyfikator generowany jest w bazie danych w postaci wyzwalacza bazodanowego. Wyzwalacz taki jest budowany dynamicznie na podstawie tabeli, której dotyczyć ma klasyfikacja oraz wyznaczony zestaw reguł tworzących klasyfikator. Jedna z procedur generuje wyzwalacz w sposób automatyczny dla tabeli o dowolnej strukturze i klasyfikatora o dowolnej złożoności. W momencie wprowadzania informacji opisujących konkretną stronę internetową wyzwalacz dokona klasyfikacji i uzupełni wstawiane dane o wartość atrybutu określającego klasę dokumentu.

6. Charakterystyka dokumentu i proces klasyfikacji

W celu prawidłowej klasyfikacji stron internetowych konieczne jest zgromadzenie pewnych cech charakteryzujących poszczególne strony. W omawianym podejściu przyjęto wydobywanie możliwie dużej liczby właściwości opisujących strony, a następnie redukcję ich liczby z wykorzystaniem teorii zbiorów przybliżonych. Redukcja taka jest niezbędna ze względu na konieczność zapewnienia efektywności działania poszczególnych algorytmów. Jednak z uwagi na fakt, że nie można jednoznacznie stwierdzić, które cechy dokumentu okażą się istotne do prawidłowej klasyfikacji, zbieranych jest możliwie wiele cech. W przypadku stron internetowych istotne cechy dotyczą zarówno treści stron (elementów widocznych dla odwiedzającego stronę), jak i ich struktury (rodzajów i treści tagów html) oraz funkcjonalności (m.in. skrypty, linki do innych stron) [18].

Bardziej precyzyjnie, cechy opisujące stronę HTML podzielić można na kilka kategorii:

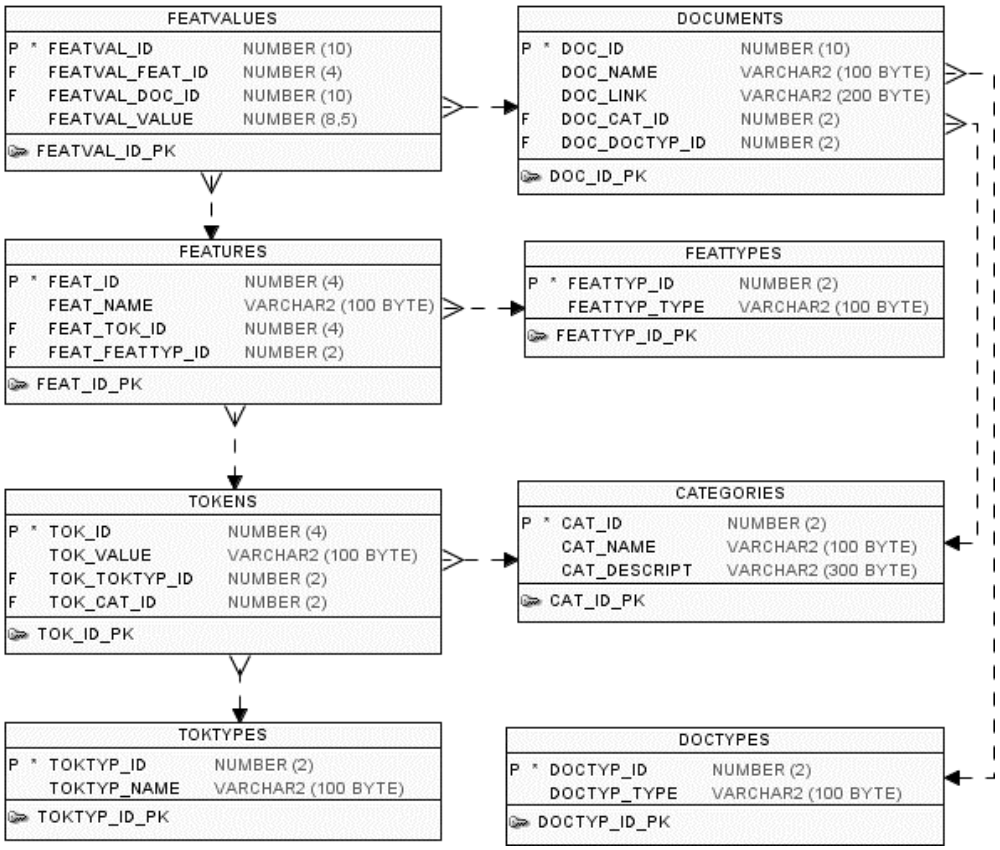
- Cechy strukturalne: stosunek ilości tagów html do ogólnej ilości treści na stronie, stosunek ilości wystąpień sekwencji tagów (tzw. *N*-gramów) do wszystkich tagów, stosunek ilości kodu skryptowego do pozostałej treści, stosunek ilości kodu skryptowego do ilości kodu html, średnia ilość wystąpień poszczególnych tagów związanych ze strukturą dokumentu w odniesieniu do wszystkich tagów.

- Cechy wizualne:
  - związane z formatowaniem – średnie ilości poszczególnych tagów formatujących,
  - związane z obrazami – stosunek ilości tagu <img> do wszystkich tagów, stosunki wystąpień obrazów w poszczególnych, typowych formatach, stosunki wystąpień obrazów o wielkościach: małych, średnich i dużych,
  - związane z plikami multimedialnymi – stosunki ilości plików w różnych formatach do ilości wszystkich plików multimedialnych,
  - związane ze stylem – w tym również występowanie odwołań do zewnętrznych arkuszy CSS.
- Cechy linków do innych stron: liczba wszystkich linków, stosunek linków prowadzących do tej samej domeny do wszystkich linków, stosunek linków prowadzących do innej domeny do wszystkich linków, stosunek linków "mailowych" do wszystkich linków, stosunek linków związanych z obrazami do wszystkich linków.
- Cechy tekstowe: statystyki słów kluczowych (zawartych w słownikach zbudowanych dla każdej kategorii), inne statystyki bazujące na słownikach, ogólne statystyki tekstu, znaki interpunkcyjne, znaki typograficzne, statystyki części mowy. Z uwagi na fakt, że prezentowane rozwiązanie było rozpatrywane dla stron w dowolnym języku, cechy tekstowe nie były brane pod uwagę w eksperymentach.

Wszystkie cechy gromadzone są w bazie danych, której struktura zaprezentowana jest na rysunku 2. Na podstawie informacji gromadzonych w prezentowanych tabelach konstruowane są dane wejściowe dla procedur z pakietu DB_RS. Dane te muszą zostać poddane procesowi dyskretyzacji, ponieważ wiele cech ma charakter ciągły [17]. Do tego celu wybrany został algorytm dyskretyzacji według równej częstości [14]. Algorytm ten dokonuje podziału dziedziny danego atrybutu na n przedziałów o różnej szerokości tak, aby każdy z nich zawierał taką samą liczbę przypadków (przeciwnie do algorytmu dyskretyzacji według równej szerokości). Istotną cechą tej metody jest fakt, iż wartości dyskretne atrybutu będą miały rozkład równomierny. Co istotne, nie ma konieczności implementowania tego algorytmu w systemie bazodanowym, ponieważ dostępna jest funkcja *width_bucket*, która realizuje właśnie taki podział. Wystarczy wykorzystać tę funkcję w poleceniu modyfikującym stan wybranej kolumny. Istotnym czynnikiem jest liczba przedziałów, do których przydzielane będą wartości. Ma to wpływ na późniejszy proces tworzenia klasyfikatora i jest dobierane na bazie przeprowadzanych eksperymentów.

Zbiór (w postaci tablicy decyzyjnej), na podstawie którego ma zostać skonstruowany klasyfikator, jest poddawany procesowi redukcji cech. Za pomocą zbiorów przybliżonych możliwe jest określenie, czy w tablicy decyzyjnej nie występują niepotrzebnie nadmiarowe informacje, które nie są niezbędne do klasyfikacji – można ustalić minimalny zbiór atrybutów niezbędnych do klasyfikacji. Na ich podstawie określana jest tablica zawierająca tylko warto-

ści niezbędne do utworzenia reguł decyzyjnych. Zbiór reguł przekształcany jest w wyzwalacz bazodanowy działający na określonej tabeli.



Rys. 2. Schemat bazy danych
Fig. 2. Database scheme

Wyzwalacz realizujący proces klasyfikacji działa automatycznie przy każdym poleceniu INSERT kierowanym do tabeli z informacjami o stronie internetowej (tabela Documents). Wartości poszczególnych cech są wówczas wydobywane ze strony i wstawiane do tabeli Featvals. W oparciu o wartości tych cech następuje klasyfikacja, a wartość atrybutu oznaczająca klasę dokumentu jest uzupełniana w tabeli Documents.

Zarówno cechy wykorzystywane do opisu stron, jak również klas, na jakie klasyfikowane są zbiory stron, mogą być dowolnie zmieniane i rozszerzane poprzez modyfikację zawartości tabel: Features oraz Categories.

7. Podsumowanie

W pracy zaprezentowano metodę tworzenia klasyfikatora stron internetowych. Podejście wykorzystuje elementy teorii zbiorów przybliżonych oraz baz danych. Teoria zbiorów przybliżonych została wykorzystana w celu zredukowania liczby atrybutów, na podstawie któ-

rych tworzony jest klasyfikator. Podstawową operacją jest ustalenie rdzenia atrybutów warunkowych, składającego się ze wszystkich nieusuwalnych atrybutów. Na jego podstawie wyznaczany jest redukt względny, który zawiera wszystkie atrybuty niezbędne do prawidłowej klasyfikacji. Korzystając z reduktu i wyznaczonej na jego podstawie tablicy reduktu, możliwe jest określenie zbioru reguł decyzyjnych tworzących klasyfikator. Z uwagi na fakt, że operacje prowadzące do otrzymania takiego klasyfikatora, opierające się na teorii zbiorów przybliżonych, cechują się dużą złożonością obliczeniową, w prezentowanym podejściu dokonano integracji zbiorów przybliżonych i baz danych. Celem takiej integracji jest wykorzystanie wydajnych rozwiązań dostępnych w systemach bazodanowych. Podstawowe operacje wykorzystywane w prezentowanym podejściu: zliczanie, projekcja, selekcja, są w środowisku bazodanowym realizowane bardzo efektywnie, nawet dla wielkich zbiorów danych. Dodatkowym atutem takiej integracji jest przetwarzanie danych w miejscu ich przechowywania, bez konieczności przenoszenia ich do osobnych aplikacji działających w oparciu o teorię zbiorów przybliżonych. W pracy zaprezentowano implementację omawianego podejścia w standardowym środowisku bazodanowym. Klasyfikator zbudowany z wykorzystaniem przykładowego zbioru stron internetowych został następnie przetestowany na zbiorze testowym. Wyniki testów potwierdziły skuteczność takiego klasyfikatora. Wykorzystanie baz danych do generowania klasyfikatora pozwala na ponowne jego wyznaczanie przy każdym znaczącym zwiększeniu się zbioru zawierającego opis stron, poprawiając tym samym jego skuteczność.

Czas trwania procesu wyznaczania klasyfikatora oraz przede wszystkim jego skuteczność zależą od wielu czynników, m.in. od ilości danych, na podstawie których wyznaczany jest redukt, a następnie konstruowane są reguły decyzyjne. Na skuteczność klasyfikatora wpływa również liczba rozpatrywanych cech stron internetowych oraz liczba klas, na jakie dzielone są zbiory stron. Dodatkowymi istotnymi parametrami są: wartość progowa dla współczynnika Laplace'a, pozwalająca odrzucić niedostatecznie ogólne reguły, a także liczba przedziałów dyskretyzacji wartości cech. Wpływ tych czynników na działanie klasyfikatora jest obecnie obiektem dalszych prac. Kolejne prace w tym zakresie obejmować będą również zwiększenie wydajności algorytmów konstruujących klasyfikator oraz porównanie wyników tej metody z innymi podejściami.

BIBLIOGRAFIA

1. Pawlak Z.: Some Issues on Rough Sets. Transactions on Rough Sets I, LNCS, Springer 2004, s. 1÷58.

2. Komorowski J., Pawlak Z., Polkowski L., Skowron A.: A Rough Set Perspective on Data and Knowledge. Rough Fuzzy Hybridization (S. K. Pal, A. Skowron, Eds.), Springer-Verlag, 1999, s. 107÷121.
3. Ferdinandez-Baizan A., Ruiz E., Sanchez J.: Integrating RDMS and Data Mining capabilities using rough sets. In Proc. IPMU'96, Granada, Spain, 1996, s. 1439÷1445.
4. Hu X., Lin T., Han J.: A New Rough Sets Model Based on Database Systems. Fundamenta Informatica 59, IOS Press 2004, s. 135÷152.
5. Hu X.: Using rough sets theory and database operations to construct a good ensemble of classifiers for data mining applications. Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference, s. 233÷240.
6. Cercone N., Ziarko W., Hu X.: Rule discovery from databases: A Decision matrix approach. In Proc. ISMIS'96, Zakopane, 1996, s. 653÷662.
7. Hu X., Shun N., Cercone N., Ziarko W.: DBROUGH: A Rough Set Based Knowledge Discovery System. Lecture Notes in Computer Science, Springer-Verlag, 1994, s. 386÷395.
8. Nguyen S. H., Nguyen H. S.: Some efficient algorithms for rough set methods. Proceedings of the Sixth International Conference, Information Procesing and Management of Uncertainty in Knowledge-Based Systems (IPMU-96), 2, Granada, Spain, 1996, s. 1541÷1457.
9. Skowron A., Rauszer C.: The Discernibility Matrices and Functions in Information Systems. Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory, K. Slowinski (ed), Kluwer, Dordrecht, 1992, s. 331÷362.
10. Mrózek A., Płonka L.: Analiza danych metodą zbiorów przybliżonych – Zastosowania w ekonomii, medycynie i sterowaniu. Akad. Oficyna Wyd. PLJ, Warszawa 1999.
11. Olson D., Delen D.: Advanced Data Mining Techniques, Springer Berlin 2008.
12. A Rough Set Toolkit for Analysis of Data: http://www.lcb.uu.se/tools/rosetta/ .
13. Rough Set Exploration System: http://logic.mimuw.edu.pl/~rses/ .
14. Knut Magne Risvik, Discretization of Numerical Attributes. Preprocessing for Machine Learning, Norwegian University of Science and Technology – Department of Computer and Information Science, 1997.
15. Czajkowski K., Drabowski M.: Wybrane zagadnienia integracji zbiorów przybliżonych i baz danych, Studia Informatica, Gliwice, Vol. 30, No. 2A(83), 2009, s. 355÷372.
16. Czajkowski K., Drabowski M.: Relational database and core, relative reducts in rough sets. Proceedings of the IASTED International Conference on Artificial Intelligence and Soft Computing, Palma de Mallorca, ACTA Press, Anaheim, USA, 2009, s.14÷20.

17. Yin S., Wang F., Xie Z., Qiu Y.: Study on Web-Page Classification Algorithm Based on Rough Set Theory, Proceedings of ISIP'2008, s. 202÷206.
18. Dong L., Watters C., Duffy J., Shepherd M.: An Examination of Genre Attributes for Web Page Classification. Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008).

Recenzent: Dr inż. Dariusz Rafał Augustyn

Wpłynęło do Redakcji 20 stycznia 2010 r.

Abstract

This paper concerns applying of rough sets theory and database system to web pages classification. One of the possibilities of classifying web pages is to generate sets of rules consisted of values of attributes describing web pages. Due to a large number of attributes characteristic for Internet documents, it is important to apply approaches which give possibility to reduce its number. In this place it is possible to use elements of rough sets theory to eliminate those features which are not important to appropriate classification. One of the problems that appears in practical application of rough sets theory is complexity of computations performed during computing of core attributes and reducts. It is purposeful to integrate rough sets with databases. Such solution gives possibilities to take advantages of efficient mechanisms existing in database management systems. It could improve the efficiency of application of rough sets theory for large data sets. The use of SQL language, embedded in each database system, gives the possibility of relatively easy implementation. Operations such as counting, projection, selection, could be efficiently realized even for large amount of data. Additional advantage in using database systems is lack of necessity to transfer, usually large, data sets to separate applications, to realize rough sets operations.

In this work the approach of integrating elements of rough sets theory with databases was proposed, which aim is improving the efficiency as well as processing data in the place of storing them. The paper includes implementations of the algorithms of core attributes and reducts selection as well as decision rules determining. The aim is web pages classifications on the basis of decision rules and the set of features describing individual web pages. Algorithms presented in this paper were implemented in Oracle database environment using PL/SQL language are efficient and could be used for analyzing any data sets.

Adres

Krzysztof CZAJKOWSKI: Politechnika Krakowska, Katedra Teleinformatyki,
Wydział Fizyki, Matematyki i Informatyki Stosownej, ul. Warszawska 24, 31-155 Kraków,
Polska, kc@pk.edu.pl .