PRACE NAUKOWE AKADEMII EKONOMICZNEJ WE WROCŁAWIU
Nr 1064 — 2005
Pozyskiwanie wiedzy i zarządzanie wiedzą

Ewa Szpunar-Huk
Politechnika Wrocławska

POZYSKIWANIE WIEDZY Z DANYCH PRZY WYKORZYSTANIU KLASYFIKATORÓW ZŁOŻONYCH

1. Wstęp

Metody budowy klasyfikatorów złożonych stanowią jeden z bardziej dynamicznie rozwijających się kierunków dziedziny pozyskiwania wiedzy z danych. Wykorzystanie takich klasyfikatorów wymaga co prawda większych zasobów pamięciowych oraz większych nakładów obliczeniowych, jednak eksperymenty pokazują, że systemy takie, w porównaniu z pojedynczymi heurystykami, zwykle osiągają dużo lepsze rezultaty. W połączeniu z możliwością łatwego zrównoleglenia większości algorytmów generowania grup klasyfikatorów czyni to z nich atrakcyjny obiekt badań i nowych poszukiwań. Jednakże budowa takich systemów znacznie utrudnia, a niekiedy wręcz uniemożliwia uzyskiwanie zrozumiałych wyjaśnień podejmowanych decyzji. Traci się w ten sposób jeden z głównych celów stawianych metodom pozyskiwania wiedzy, a mianowicie zdolność klarownego wytłumaczenia podejmowanych decyzji. Stąd wraz z rozwojem algorytmów grupujących klasyfikatory rozwijają się metody pozyskiwania wiedzy z komitetów klasyfikatorów bądź upraszczania ich zapisu tak, aby był bardziej zrozumiały dla człowieka.

W związku z tym wzrasta zainteresowanie teoretycznymi podstawami funkcjonowania klasyfikatorów złożonych (komitetów/systemów klasyfikatorów, ang. *ensemble*). Podstawowe warunki, jakie muszą być spełnione, aby klasyfikatory takie dawały lepsze wyniki niż pojedyncze hipotezy, silnie bowiem rzutują na kształt stosowanych metod tworzenia komitetów klasyfikatorów i zasad podejmowania decyzji przez grupę. Fakt ten zaznacza się jednak najsilniej w procesach upraszczania zapisu oraz pozyskiwania wiedzy z systemów złożonych. Z tego względu prowadzone przez autorkę badania dotyczące pozyskiwania wiedzy z

komitetów klasyfikatorów dobrze wpisują się w ramy aktualnych trendów rozwoju sztucznej inteligencji. Przez ich pryzmat łatwo dostrzec pożądane kierunki dalszego rozwoju tego typu prac.

2. Podstawy budowy klasyfikatorów złożonych

Na potrzeby dalszych rozważań przyjmijmy pewne założenia dotyczące zadania klasyfikacji. Dla każdego problemu określony jest zbiór danych $D = \{t_1, t_2, .., t_N\}$. Każdy przykład $t_i = \langle \underline{x}, c \rangle \in \underline{X} \times C$ ze zbioru D jest wektorem danych $\underline{x} = \langle x_1, x_2, .., x_m \rangle$ skojarzonym z etykietą klasy c, gdzie x_i jest nazywany wartością atrybutu przykładu. Dziedzina atrybutu x_i to X_i, natomiast $\underline{X} = X_1 \times X_2 \times .. \times X_m$ jest dziedziną przykładu, a C – dziedziną etykiet klas. Sposób przyporządkowywania wektorowi danych odpowiedniej etykiety jest określony za pomocą nieznanej funkcji F. Zadaniem klasyfikacji jest znalezienie hipotezy h: $\underline{X}$ -> C, która dla każdego przykładu t_i zwróci odpowiadającą mu, możliwie najbardziej prawdopodobną wartość etykiety klasy. Hipotezę tę nazywamy klasyfikatorem. Klasyfikator złożony to zbiór pojedynczych klasyfikatorów, których decyzje łączone są w pewien sposób, tak aby przyporządkowywać etykiety dla nowych przykładów.

Metody łączenia klasyfikatorów stały się w ostatnich latach popularne, gdyż o ile dla wielu zastosowań uczenia nadzorowanego można znaleźć hipotezy, które z dużą dokładnością przyporządkowują dane do klas dla wybranych fragmentów dziedziny atrybutów, o tyle nie sprawdzają się one w całej dziedzinie. Złożenie klasyfikatorów może w znaczny sposób poprawić zdolność predykcji całego systemu. Badania empiryczne pokazały, że dla licznych problemów klasyfikatory złożone osiągają lepsze rezultaty niż klasyfikatory pojedyncze.

Oprócz badań empirycznych istnieją też opracowania teoretyczne dotyczące sensowności stosowania klasyfikatorów złożonych. Valentini w [21] podaje trzy powody, dla których zawsze można zbudować dobry klasyfikator złożony. Pierwszy powód, statystyczny w swej naturze, jest następujący: przy ograniczonym rozmiarze danych uczących algorytm przeszukiwania przestrzeni hipotez może znaleźć wiele różnych hipotez o podobnym prawdopodobieństwie popełnienia błędu. Agregacja ich odpowiedzi w procesie głosowania zaś może zmniejszać ryzyko błędnej decyzji ze względu na występujące w przypadku różnych klasyfikatorów nieskorelowanie popełnianych przez nie błędów. Z kolei z obliczeniowego punktu widzenia, wiele algorytmów budujących klasyfikatory proste, stosując przeszukiwanie lokalne, „utyka" w lokalnych minimach. Dopiero budowa klasyfikatora złożonego z wygenerowanych w taki sposób różnych hipotez prostych pozwala na zadowalającą aproksymację badanej funkcji. Trzeci powód natomiast związany jest z ograniczeniami wprowadzanymi przez wybraną reprezentację hipotez: każda skończona reprezentacja pozwala odwzorowywać jedynie skończoną liczbę hipotez. W związku z tym w większości przypadków poszukiwana funkcja nie może być dokładnie reprezentowana przez żadną z hipotez w ramach danej reprezentacji.

Określenie zaś pewnego złożenia klasyfikatorów może rozszerzyć ostatecznie stosowaną reprezentację w sposób pozwalający na dokładniejsze przybliżenie poszukiwanego rozwiązania.

2.1. Warunek konieczny i dostateczny

Przytaczanym m.in. w [9] i [21] warunkiem koniecznym i dostatecznym, aby komitet klasyfikatorów działał lepiej niż każdy z jego składników, jest zapewnienie ich dokładności i odpowiedniego zróżnicowania. Przez dokładność rozumie się tu ograniczenie prawdopodobieństwa popełniania błędów przez klasyfikator do poziomu mniejszego niż w przypadku wyboru odpowiedzi metodą losowania. Zróżnicowanie (Hansen, Salamon 1990) ma zaś zapewniać przywoływany już brak korelacji pomiędzy błędnymi decyzjami poszczególnych klasyfikatorów.

Aby to lepiej zobrazować, rozważmy za Dietterichem [9] przykładowy problem klasyfikacyjny. Komitet w liczbie L klasyfikatorów rozwiązuje problem przynależności pewnych elementów do dwóch klas. Jeśli L = 21 i każdy z jurorów popełnia błędy z prawdopodobieństwem p równym 0,3, to przy założeniu braku korelacji pomiędzy popełnianymi przez nie błędami i głosowaniu większościowym prawdopodobieństwo popełnienia błędu P przez cały komitet wyraża się wzorem:

$$P = \sum_{(i=\lceil L/2 \rceil)}^{L} \binom{L}{i} p^i (1-p)^{L-i} \Rightarrow P = 0{,}026 << p = 0{,}3$$

Zgodnie z przewidywaniami jest ono mniejsze od prawdopodobieństw występowania błędów dla poszczególnych klasyfikatorów składowych. Szczegółową analizę tego faktu wykonał Ali w [1]. Wykazał on istnienie liniowej zależności pomiędzy redukcją błędu całego systemu klasyfikatorów a dekorelacją błędów poszczególnych jego elementów (w oryginale drzew decyzyjnych). Dodatkowo pokazał on również, że komitety popełniające błędy we wzajemnie negatywnie skorelowany sposób osiągają lepsze rezultaty niż te, które mylą się jedynie w sposób nieskorelowany.

2.2. Metody osiągania zróżnicowania klasyfikatorów

Ze względu na podstawowe znaczenie zróżnicowania klasyfikatorów elementarnych dla funkcjonowania ich komitetów, rozwój metod tworzenia tego typu systemów polega w dużej mierze na poszukiwaniu sposobów spełnienia tego warunku. Metody dedykowane dla tego zadania bazują przeważnie na odpowiednim manipulowaniu parametrami i pracą klasycznych algorytmów generowania klasyfikatorów. Typowe rozwiązania to losowe zaburzanie wyników lub procesu ich budowy oraz ograniczanie i modyfikacje zbiorów uczących. Stosowane są także rozwiązania hybrydowe, wykorzystujące zarówno różne algorytmy budowy klasyfikatorów, jak i różne ich docelowe reprezentacje [2; 3; 9; 21].

Metody wprowadzające losowość stosowane głównie są w przypadku wykorzystania sieci neuronowych, w których poprzez wprowadzanie losowych wag początkowych najłatwiej zmienić punkt startowy algorytmu uczącego [20]. Badania takie były prowadzone m.in. przez Kolena i Pollacka, Sharkeya, Parmanto i Munro [21]. Jednak metoda ta obecnie uważana jest za jedną z najsłabszych metod wprowadzających zróżnicowanie w zbiorze klasyfikatorów. Inny sposobem jest losowe zaszumianie przykładów uczących (taką metodę badali Raviv i Intrator [19]). Losowość także łatwo wprowadzić do algorytmów generujących drzewa decyzyjne, np. metodą C4.5 [16], nad czym badania prowadził Diettriech [9].

Kolejne, bardziej popularne i dające dużo lepsze rezultaty metody osiągania zróżnicowania klasyfikatorów polegają na zmianie zbiorów uczących, na podstawie których generowane są kolejne klasyfikatory składowe. Metody te opierają się albo na redukcji liczby przykładów uczących lub redukcji liczby atrybutów, albo też na dokonywaniu zmian w przyporządkowaniu przykładów do klas. Okazały się one skuteczne dla takich algorytmów generujących klasyfikatory, które są wrażliwe na małe zmiany w zbiorze uczącym, czyli zarówno dla większości algorytmów generowania drzew decyzyjnych, jak i sieci neuronowych.

Najbardziej popularną niedeterministyczną metodą redukcji liczby przykładów uczących jest bagging – metoda opisana przez Breimana w 1996 r. [2], w której z całego zbioru N przykładów uczących wybierane jest losowo również N przykładów, ale możliwe są powtórzenia. Wykorzystywane i badane były również inne sposoby dzielenia zbioru uczącego na podzbiory. Porównanie różnych metod tego typu przedstawił Chawla w [5].

Dane uczące można wybierać także w sposób deterministyczny. Metodą taką jest boosting, zaproponowany przez Freunda i Schapire'a w 1996 r. [12], polegający na tym, że każdy następny klasyfikator generowany jest dla danych uczących, którym nie odpowiadały poprzednie klasyfikatory. W tym celu z każdym przykładem uczącym związana jest pewna waga, która ulega zmianie po wygenerowaniu kolejnego klasyfikatora.

Kolejny sposób zapewnienia różnorodności klasyfikatorów wchodzących w skład komitetu to zmiana liczby atrybutów zbioru uczącego poprzez wycięcie pewnej ich liczby. Jednak metody takie mogą być skuteczne jedynie wtedy, gdy zbiór atrybutów jest w dużej mierze nadmiarowy. Skuteczny klasyfikator złożony, stosując metodę redukcji atrybutów, otrzymał np. Cherkauer [6], który wybierał 8 zbiorów spośród 119 cech jako wejście dla sieci neuronowych identyfikujących wulkany na Wenus, ale w tym podejściu podzbiory atrybutów nie były wybierane w sposób losowy, lecz określane były przez człowieka analizującego semantykę poszczególnych cech.

Dobre rezultaty otrzymali Dietterich i Bakiri, proponując ciekawą metodę bazującą na manipulacjach etykietami klas [9]. Dla każdego klasyfikatora zmieniali oni zbiór uczący następująco: dzielili zbiór klas w losowy sposób na dwa podzbio-

ry A i B, wszystkie przykłady, które miały etykiety należące do zbioru A, otrzymywały nową etykietę 0, a te należące do zbioru B – etykietę 1. Tak wyuczonych N klasyfikatorów podejmowało decyzję podczas specyficznego głosowania: jeśli klasyfikator przyporządkował przykład do klasy 0, to głos otrzymywały wszystkie klasy wchodzące w skład zbioru A, jeśli zaś do klasy 1, to głosy otrzymywały klasy ze zbioru B. Następnie sumowane były głosy dla poszczególnych klas i wybierana była klasa o największej liczbie głosów. Metoda ta nosi nazwę ECOC (*error-correcting output coding*).

Na koniec warto jeszcze wspomnieć o rozwiązaniach hybrydowych, generujących zmiany w architekturze poszczególnych klasyfikatorów. Na przykład Opitz [15] zaproponował algorytm ewolucyjny do optymalizacji topologii sieci neuronowych wchodzących w skład komitetu. W jego metodzie łączone są klasyfikatory uzyskane różnymi algorytmami uczenia. Z kolei Wang [22] łączył sieci neuronowe z drzewami decyzyjnymi, a Seewald, Petrak i Widmer [19] zaproponowali klasyfikator złożony z drzew decyzyjnych uzyskiwanych za pomocą różnych algorytmów ich budowy. Woods natomiast łączył sieci neuronowe, metodę k-najbliższych sąsiadów, drzewa decyzyjne i klasyfikator bayesowski, ale dla danego przykładu nie łączył on rezultatów dawanych przez wszystkie wygenerowane klasyfikatory, lecz wybierał odpowiedni klasyfikator, który zwracał wynik [3].

2.3. Stosowane techniki głosowania

Kolejną ważną kwestią dotyczącą klasyfikatorów złożonych jest sposób łączenia wyników dawanych przez pojedyncze klasyfikatory. Najprostszym sposobem podziału istniejących algorytmów jest podział na metody pasywne i aktywne. W metodach pasywnych sposób podejmowania decyzji jest niezależny od sposobu generowania pojedynczych klasyfikatorów, natomiast przy metodach aktywnych wyniki uzyskiwane przez kolejne klasyfikatory determinują parametry tworzenia następnych, które mają wejść w skład komitetu.

Metody pasywne polegają ogólnie rzecz biorąc, na łączeniu wyników dawanych przez już wygenerowane, pojedyncze klasyfikatory. Są to w głównej mierze metody głosowania, takie jak głosowanie większościowe (występujące w opisanym wcześniej baggingu), głosowanie ważone, w którym wagi określane są np. w zależności od dotychczasowych wyników dawanych przez pojedyncze klasyfikatory [23], czy też głosowanie specjalizowane [12], w którym klasyfikatory mogą wstrzymać się od głosu. Bardziej skomplikowanym sposobem jest metoda zaproponowana przez Kunchevę, polegająca na generowaniu szablonów decyzyjnych dla poszczególnych klas z osobna przy dodatkowym wykorzystaniu logiki rozmytej [14].

Można tu też wyszczególnić metody hierarchiczne, w których podejmowanie decyzji jest kilkustopniowe. Na przykład w metodach Chan i Stolfo ostateczną decyzję podejmuje nadrzędny klasyfikator, zwany konsultantem, otrzymujący jako wejście wyniki klasyfikatorów wchodzących w skład komitetu, lub arbiter, który

rozwiązuje problemy, kiedy jest to potrzebne, a uczony jest na przykładach, dla których podrzędne klasyfikatory dają złe odpowiedzi [5].

Spośród metod aktywnych najbardziej popularnym sposobem łączenia klasyfikatorów, mającym wpływ na proces generowania całego komitetu, jest wspomniany wcześniej boosting [16], który doczekał się licznych modyfikacji. Do metod aktywnych należą również dynamiczne metody selekcji klasyfikatorów wchodzących w skład komitetu. Na przykład Chu i Zaniolo w 2004 r. opracowali metodę pozyskiwania wiedzy ze strumieni danych przy wykorzystaniu zbioru klasyfikatorów, z którego najstarsze klasyfikatory (tzn. wygenerowane na podstawie najstarszych danych ze strumienia) zastępowane są młodszymi [7].

Kwestia doboru sposobu podejmowania decyzji poprzez komitet klasyfikatorów nie jest kwestią zamkniętą, a ponieważ jest często, jak w przypadku metod aktywnych, powiązana z metodami generowania grup klasyfikatorów, można się w najbliższych latach spodziewać kolejnych ciekawych metod i dotyczących jej analiz.

2.4. Pozyskiwanie wiedzy z klasyfikatorów złożonych

Chociaż liczne eksperymenty wykazały znacznie większą skuteczność klasyfikatorów złożonych w porównaniu z klasyfikatorami elementarnymi, to niezależnie od postaci tych ostatnich, w przypadku komitetów zatraca się możliwość klarownego wyjaśniania sposobu podejmowanych decyzji. Jest to mankament na tyle istotny we wszystkich praktycznych zastosowaniach – szczególnie związanych z pozyskiwaniem wiedzy z danych, że wraz z rozwojem technik generowania klasyfikatorów złożonych rozpoczęto prace nad metodami upraszczania ich opisu. Proces ten zakłada pewien akceptowalny stopień utraty poprawności klasyfikacji, w związku z czym jest nazywany często wtórnym pozyskiwaniem wiedzy z komitetów.

Najprostsze metody polegają na wyborze spośród uzyskanych pojedynczych klasyfikatorów najbardziej ogólnego lub najbardziej semantycznie podobnego do całego systemu, tzn. dającego najbardziej podobne błędy klasyfikacji [11]. Większość obecnych metod jednak dotyczy komitetów złożonych z sieci neuronowych i niewiele jest rozwiązań dla systemów drzew decyzyjnych. Do ciekawszych metod należy np. zaproponowana w [10] metoda polegająca na wprowadzeniu nowej struktury służącej do zapisu kilku drzew, nazwanej *multitree*. Dobre rezultaty osiągali także Park i Kargupta, upraszczając drzewa przy wykorzystaniu transformaty Fouriera [16].

3. Przeprowadzone badania

Celem przeprowadzonych eksperymentów było sprawdzenie możliwości poprawy jakości pojedynczego drzewa decyzyjnego, wygenerowanego z algorytmem pozyskującym wiedzę systemu klasyfikatorów, poprzez wstępną reduk-

cję komitetu przy użyciu zaproponowanej metody wyboru drzew. Na potrzeby badań zbudowano klasyfikator złożony, składający się z K drzew decyzyjnych, wygenerowanych algorytmem C4.5 Quinlana [17] przy wykorzystaniu zbiorów przykładów uczących i testowych pobranych z UCI Machine Learning Repository [20]. Wyniki dawane przez pojedyncze klasyfikatory, wchodzące w skład komitetu, łączone były przez głosowanie większościowe. Do pozyskiwania wiedzy z utworzonych klasyfikatorów złożonych został wykorzystany algorytm Trepan [8], który powstał jako narzędzie pozyskiwania wiedzy z sieci neuronowych, a na potrzeby opisywanych badań został dostosowany do pozyskiwania wiedzy z drzew decyzyjnych.

3.1. Testy komitetu klasyfikatorów

Na początku warto przedstawić uzyskane wyniki dotyczące oceny poprawności klasyfikacji konstruowanych systemów złożonych. Komitety były generowane przy zmiennych wartościach parametrów, takich jak liczba drzew wchodzących w skład grupy oraz wielkość zbiorów uczących wykorzystanych do generowania kolejnych klasyfikatorów.

W tab. 1 przedstawiono średnie błędy procentowe klasyfikacji dla stu testów, jakie dla zbiorów danych testowych dawał klasyfikator składający się z jednego drzewa, przy zbiorze uczącym zawierającym 100% przykładów uczących, a także średnie błędy procentowe klasyfikatorów złożonych z 11 bądź 50 drzew, generowanych dla zbiorów uczących o liczbie przykładów ograniczonej od 50 do 100%, wybieranych losowo (bez powtórzeń) spośród całego zbioru. Testy przeprowadzono dla zbiorów o nazwach: *soybean*, *monk_1* i *vote*.

Badania potwierdziły skuteczność klasyfikatora złożonego w porównaniu do pojedynczego drzewa decyzyjnego. Poprawa jest już widoczna dla komitetu złożonego z 11 drzew. Manipulacja liczbą przykładów uczących daje bardzo dobre rezultaty, jednak nie poprawia wyników dla niewielkich zbiorów danych, takich jak *monk_1* (jedynie 100 krotek uczących), który dodatkowo, jako zbiór sztucznie wygenerowany, charakteryzuje się bardzo małą nadmiarowością danych, dlatego wyeliminowanie w jego przypadku już 20% przykładów ze zbioru uczącego znacznie pogarsza uzyskiwane wyniki.

Tabela 1. Procentowy błąd klasyfikacji w zależności od warunków generowania komitetu

Liczba drzew w komitecie	1	11					50				
% przykładów uczących	100	100	80	70	60	50	100	80	70	60	50
soybean	11%	10%	9,7%	9,1%	9,4%	9,7%	10%	10%	9%	8,2%	8,1%
vote	5,5 %	5,4%	3,3%	2,9%	2,6 %	3,1	5,4	2,6%	2,6 %	2,4%	2,6%
monk_1	14%	14%	12%	14%	18%	20%	11%	12%	13%	16%	19%

Źródło: opracowanie własne.

Można również zauważyć dwie następujące tendencje: dla każdego z przebadanych problemów istnieje charakterystyczna, optymalna wielkość zbioru uczącego, zapewniająca minimalizację błędu klasyfikacji, poza tym dla ustalonej liczby przykładów uczących wykorzystanej do treningu komitetu błąd klasyfikacji spada w miarę wzrostu liczby drzew w klasyfikatorze złożonym. Oba te fakty wypływają bezpośrednio z podstaw funkcjonowania systemów klasyfikatorów. W przypadku wykorzystania zbiorów uczących zawierających mniej niż 50% danych całego, dostępnego zbioru uczącego, błąd klasyfikacji ponownie wzrastał powyżej akceptowalnego poziomu 20%.

3.2. Redukcja liczby klasyfikatorów

Kolejnym krokiem zmierzającym w kierunku pozyskania wiedzy z systemu klasyfikatorów było dokonanie redukcji liczby drzew, tak aby można było zachować akceptowalny poziom błędu klasyfikacji przykładów testowych przez komitet drzew. Zaproponowano i przebadano następującą metodę redukcji: najpierw dla każdego drzewa d_i, wchodzącego w skład komitetu, dla każdej z klas c z osobna, określana była liczba prawidłowo przyporządkowanych do tej klasy przez dane drzewo przykładów testowych: $(S_c(d_i))$. Następnie dla każdej klasy z osobna, spośród wszystkich drzew wybierane było to, które dawało najwięcej prawidłowych odpowiedzi dla wybranej klasy, i to drzewo dodawane było do zredukowanego komitetu. W przypadku, gdy dwa lub więcej drzew prawidłowo przyporządkowywało taką samą, największą liczbę przykładów do danej klasy, do komitetu wybierano to, które miało największą sumę poprawnie sklasyfikowanych przykładów dla wszystkich klas. Wybrane drzewo nie było dodawane wówczas, gdy już wchodziło w skład komitetu.

Testy pokazały, że klasyfikatory ze zredukowaną w ten sposób liczbą drzew cechowały niższe wartości błędów klasyfikacji dla zbiorów testowych. Na przykład dla komitetu K_1 utworzonego z 50 drzew dla zbioru *soybean* średni błąd klasyfikacji po 20 testach dla zbioru testowego, na podstawie którego dokonywana była redukcja, wynosił 8%, natomiast dla komitetu zredukowanego K_2 – jedynie 5,6%. Co więcej, na innym zbiorze testowym K_1 miał średni błąd równy 8,9%, natomiast K_2 jedynie 7%. Średnia liczba drzew w K_2 wynosiła 5,5.

Zredukowanie liczby drzew wchodzących w skład komitetu przy zachowaniu lub redukcji błędu klasyfikacji jest korzystne ze względu na zmniejszenie nakładów obliczeniowych potrzebnych do klasyfikacji nowych przykładów, a także ułatwia pozyskiwanie wiedzy z klasyfikatora złożonego, czego dotyczył trzeci rodzaj badań.

3.3. Pozyskiwanie wiedzy z komitetu klasyfikatorów

Do ekstrakcji wiedzy z komitetu klasyfikatorów zastosowany został wspomniany wcześniej algorytm Trepan [8], zapisujący pozyskaną wiedzę w formie drzewa. Wiedza ekstrahowana była najpierw z komitetu składającego się z 50 kla-

syfikatorów, a następnie z komitetu o zredukowanej, za pomocą metody opisanej w poprzednim punkcie, liczbie drzew.

Drzewa uzyskane Trepanem dla wielu przypadków okazały się lepsze niż pojedyncze drzewo uzyskane algorytmem C4.5 i dawały akceptowalnie większy błąd na zbiorze testowym od błędu dla wyjściowego klasyfikatora zarówno złożonego, jak i zredukowanego. Średnie wartości błędów klasyfikacji pojedynczego drzewa decyzyjnego, komitetów klasyfikatorów i drzew uzyskanych Trepanem, uzyskane dla zbiorów *soybean, vote* i *monk_1* prezentuje tab. 2.

Tabela 2. Średnie wartości błędów klasyfikacji danych testowych dla różnych klasyfikatorów

Rodzaj klasyfikatora	Pojedyncze drzewo	Klasyfikator złożony (K)	Klasyfikator zredukowany (KR)	Drzewo uzyskane Trepanem z K	Drzewo uzyskane Trepanem z KR
soybean	11 %	8,2 %	5,6 %	9,6 %	8 %
vote	5,5 %	2,4 %	4,4 %	3,7 %	2,6 %
monk_1	13 %	10 %	10 %	16 %	11,9 %

Źródło: opracowanie własne.

Dodatkowo zauważono następującą zależność: rozmiar drzew pozyskanych ze zredukowanego komitetu jest większy niż pozyskanych z wyjściowego klasyfikatora złożonego z 50 drzew, ale drzewa te charakteryzują się mniejszym błędem klasyfikacji przykładów ze zbioru testowego średnio o ok. 1,5%. Dzieje się tak najprawdopodobniej dlatego, że po redukcji liczby drzew przestrzenie decyzyjne komitetu mają wyraźniej zarysowane granice, co pozwala Trepanowi zbudować, przy tych samych wartościach parametrów sterujących procesem poszukiwania rozwiązań, drzewa o niższym błędzie na danych testowych. Z kolei na zbiorach uczących wyekstrahowane drzewa dają podobne wyniki bez względu na to, czy były generowane na podstawie całego, czy zredukowanego komitetu.

4. Podsumowanie

Kończąc rozważania na temat przedstawionej metody ekstrakcji wiedzy z klasyfikatorów złożonych, warto jeszcze raz podkreślić, iż udało się pokazać, że umożliwia ona uzyskiwanie opisów komitetów klasyfikatorów lepszych niż w przypadku niewykorzystywania metody głosowania z opisaną redukcją liczby drzew. Stanowi to wskazanie na potencjalne korzyści z jej stosowania we wszystkich dziedzinach, gdzie od aplikacji metod sztucznej inteligencji (np. medyczne czy bankowe systemy doradcze) wymaga się wyjaśniania podjętych decyzji poprzez podanie przesłanek i toku wnioskowania w formie czytelnej dla człowieka. Jednak rozmiar problemu redukcji klasyfikatorów złożonych wskazuje na konieczność przeprowadzenia dużo szerszych badań, np. na większej liczbie zbiorów testowych i z zastosowaniem innych metod głosowania oraz pozyskiwania wiedzy.

Niezwykle interesujące byłoby także sprawdzenie funkcjonowania zaproponowanej metody w przypadku wybranych problemów praktycznych (z dziedzin takich jak ekonomia, chemia czy medycyna), dla których nie są znane optymalne, analityczne metody poszukiwania rozwiązań. Doskonałymi przykładami są tu zadania zarządzania portfelem inwestora giełdowego i identyfikacji własności cząsteczek białek w zależności od ich struktury przestrzennej. W obu bowiem przypadkach rozmiar przestrzeni rozwiązań istotnie utrudnia stosowanie klasycznych algorytmów przeszukiwania i skłania do stosowania heurystyk.

Należy jednak pamiętać, iż dobór konkretnych zadań stawianych w trakcie badań przed nowym rozwiązaniem jest niezwykle istotny. Powinien on bowiem nie tylko zapewnić możliwość prostej interpretacji rezultatów oraz bezpośredniego porównania uzyskanych wyników z pracami innych badaczy (popularność i standaryzacja znanych baz testowych), ale także wykluczyć wszelki nieobiektywizm oceny badanej metody. Zachodzi bowiem możliwość nadmiernego dopasowania danych testowych do badanego rozwiązania (bądź odwrotnie).

Problemy te można jednak ominąć, odwołując się ponownie do powszechnie wykorzystywanych i stanowiących standard przy testowaniu metod drążenia danych zasobów UCI Machine Learning Repository [20]. Zawierają one bowiem, obok zbiorów wygenerowanych sztucznie, również dane dotyczące wielu, także wspomnianych wyżej, praktycznych problemów (np. udostępnione przez NASA zbiory pomiarów satelitarnych, na podstawie których określa się obfitość plonów). Charakter tych danych jest dobrze znany i szeroko opisany, co pozwala na wszechstronną i dogłębną weryfikację własności badanej metody. Z tego właśnie względu, oraz przez to, iż proponowane rozwiązania znajdują się w dalszym ciągu w fazie badań laboratoryjnych, dalsze prace nad przedstawionymi metodami związane będą głównie z ich testami na kolejnych standardowych zbiorach danych.

Literatura

[1] Ali K.M., Pazzani M.J., *Error Reduction Through Learning Multiple Descriptions*, „Machine Learning", 1996.

[2] Breiman L., *Bagging Predictors*, „Machine Learning", 2000.

[3] Brown G., Wyatt J., Harris R., Yao Xin, *Diversity Creation Methods: A Survey and Categorisation* Information Fusion Journal, (Special issue on Diversity in Multiple Classifier Systems), 2004.

[4] Chan P.K., Stolfo S.J., *A Comparative Evaluation of Voting and Meta-learning on Partitioned Data*, Proc. Twelfth International Conference on Machine Learning,. Tahoe City, CA: Morgan Kaufmann, 1995.

[5] Chawla N.V., Moore T.E., Hall L.O., Bowyer K.W., Kegelmeyer W.P., *Bagging is a Small--Data-Set Phenomenon*, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, Springer C., 2001.

[6] Cherkauer K., *Human Expert-Level Performance on a Scientific Image Analysis Task by a System Using Combined Artificial Neural Networks*, Thirteenth National Conference on Artificial Intelligence. Portland, AAAI Press (1996).

[7] Chu Fang, Zaniolo C., *Fast and Light Boosting for Adaptive Mining of Data Streams*, PAKDD, 2004.

[8] Craven M.W., Shavlik J.W., *Extracting tree-structured representations of trained networks*. [in:] Touretzky D., Mozer M., and Hasselmo M. (eds.), *Advances in Neural Information Processing Systems 8*, Cambridge, MA: MIT Press, 1996.

[9] Dietterich T.G., *Ensemble methods inmachine learning*, [in:] J. Kittler and F. Roli, (eds.), *Multiple Classifier Systems*, First International Workshop, MCS 2000, Cagliari, Italy, Springer--Verlag, 2000.

[10] Estruch V., Ferri C., Hernández-Orallo J., Ramírez-Quintana M.J., *Re-designing Cost-sensitive Decision Tree Learning*, [in:] Herrera F., Riquelme J.C., Aguilar J.S., Workshop of „Data Mining and Learning" in the „VIII Conferencia Iberoamericana de Inteligencia Artificial", Iberamia'2002, Universidad de Sevilla, pp. 33-42, 2002.

[11] Ferri C., Hernández-Orallo J., Ramírez-Quintana M.J., *From Ensemble Methods to Comprehensible Models*, Proceedings of the 5th International Conference on Discovery Science, 2002.

[12] Freund Y., Iyer R., Schapire R.E., Singer Y., *An Efficient Boosting Algorithm for Combining Preferences*, ICML-98.

[13] Freund Y., Schapire R. E., *Experiments with a New Boosting Algorithm*, in Proceedings of the 13th International Conference on Machine Learning, 1996.

[14] Kuncheva L., Bezdek J., Sutton M.A., *On Combining Multiple Classifiers by Fuzzy Templates*, Proceedings of the IEEE Conference of the North American Fuzzy Information Processing Society, Piscataway, NJ: IEEE, 1998.

[15] Opitz D., *Feature Selection for Ensembles*, [in:] Proceedings of 16th National Conference on Artificial Intelligence, AAAI, 1999.

[16] Park B., Kargupta H., *Constructing Simpler Decision Trees from Ensemble Models Using Fourier Analysis*, Proceedings of the 7th Workshop on Research Issues in Data Mining and Knowledge Discovery, ACM SIGMOD 2002.

[17] Quinlan J.R., *C4.5 Programs for Machine Learning*, Morgan Kauffman, 1993.

[18] Raviv Y., Intrator N., *Bootstrapping with Noise: An Effective Regularization Technique*, „Connection Science", 1996.

[19] Seewald A.K., Petrak J., Widmer G., *Hybrid Decision Tree Learners with Alternative Leaf Classifiers: An Empirical Study*, FLAIRS-2001, AAAI Press, Menlo Park, California, 2001.

[20] UCI Machine Learning Repository, [http://www.ics.uci.edu/~mlearn/MLRepository.html].

[21] Valentini G., Masulli F., *Ensembles of Learning Machines*, [in:] Neural Nets WIRN Vietri-02, Series Lecture Notes in Computer Sciences, M. Marinaro and R. Tagliaferri, Eds.: Springer-Verlag, Heidelberg (Germany), 2002.

[22] Wang W., Jones P., Partridge D., *Diversity between Neural Networks and Decision Trees for Building Multiple Classi¯er Systems*, [in:] Proc. Int. Workshop on Multiple Classi¯er Systems (LNCS 1857), Springer, Calgiari, Italy, 2000.

[23] Wang H., Fan W., Yu P., Han J., *Mining Concept-drifting Data Streams Using Ensemble Classifiers*. In ACM SIGKDD, 2003.

KNOWLEDGE ACQUISITION FROM DATA USING ENSEMBLE CLASSIFIERS

Summary

The article discusses main problems connected to the issues of knowledge discovery from data using heterogenous ensemble classifiers (or committee). There are presented main types of such classifiers and their architectures, methods of building and ways of making decisions. In this study are described main achievements in this domain and theoretical backgrounds, which explain principles and interesting properties of committee classifiers. The article also points to the need of knowledge extraction from ensemble of classifiers and within the framework of this domain there is presented a short survey of some used techniques and knowledge extraction methods. The study contains proposal of the method of knowledge extraction from ensemble classifiers based on reduction of number of simple classifiers , which are a part of ensemble (in particularly decision trees), and on using Trepan algorithm. To show the legitimacy of proposed method there are presented descriptions of experiments, along with analyses of their results. As a completion there are presented proposals of some improvements, which demand yet further analysis and research.