Tomasz ŁUKASZUK[1], Leon BOBROWSKI[1,2]
[1]Wydział Informatyki Politechniki Białostockiej,
[2]Instytut Biocybernetyki i Inżynierii Biomedycznej PAN, Warszawa
E-mail: tomluk@ii.pb.bialystok.pl, leon@ibib.waw.pl

Pozyskiwanie wiedzy z baz danych z pomocą sieci neuropodobnych

1 Wstęp

Ciągły wzrost technicznych możliwości gromadzenia i analizy danych, w których ukryte są potencjalnie cenne informacje dopełniające ludzką wiedzę, stał się motywacją powstania i rozwoju nowej dziedziny informatyki nazywanej pozyskiwaniem wiedzy z baz danych (ang. *knowledge discovery in databases*, w skrócie KDD). Zajmuje się ona tworzeniem i stosowaniem algorytmicznych narzędzi służących do odkrywania wiedzy z danych. Formalnie dziedzinę definiuje się jako: zaawansowana ekstrakcja wnioskowalnych, pierwotnie nieznanych i potencjalnie użytecznych informacji z danych [1]. Dąży się w niej do wydobycia z posiadanych danych istotnych, interesujących i przydatnych zależności, które umożliwiają dokonywanie określonych rodzajów wnioskowania i dają szansę posiadaczowi danych na poprawienie strategii jego działania.

W pozyskiwaniu wiedzy z baz danych często jako narzędzia stosuje techniki zaliczane do sztucznej inteligencji, między innymi drzewa decyzyjne [6], algorytmy genetyczne [2], czy sztuczne sieci neuronowe [7]. Ostatnia wymieniona gałąź, sztuczne sieci neuronowe, mimo kilku niepowodzeń i przerw w badaniach w ciągu swojej historii [4], w ostatnich latach zdobyła szeroką popularność. Stosuje się ją z powodzeniem w wielu dziedzinach naukowych i praktycznych. Rozwijane są wcześniej stworzone modele oraz powstają nowe koncepcje. Szczególnie interesujące, bo najbardziej zbliżone w sensie procesu uczenia do swego pierwowzoru – ludzkiego mózgu, są samoorganizujące sieci neuronowe. Niniejsza praca poświęcona jest temu typowi sieci neuropodobnych oraz ich zastosowaniu w pozyskiwaniu wiedzy z baz danych.

2 Etapy procesu pozyskiwania wiedzy z baz danych

Pozyskiwania wiedzy z baz danych można przedstawić jako proces składający się z sześciu faz. Zaliczają się do niego w kolejności ich następowania: budowa zbioru danych, czyszczenie, wzbogacanie, kodowanie, eksploracja danych, raportowanie.

Na wstępie procesu należy w miarę możliwości sprecyzować cel analizy. Często jednak złożoność natury danych, a także złożoność problemów stawianych podczas ich analizy sprawia, że ostateczny cel krystalizuje się dopiero podczas kolejnych etapów.

Następnie dokonuje się doboru źródeł danych. Dane nie mogą być wybrane w sposób przypadkowy. Wybór zależy oczywiście od pytań, na które ma być dana odpowiedź.

Etap czyszczenia zapewnia efektywność i niezawodność występującej w dalszej perspektywie eksploracji danych. Dane otrzymywane po zakończeniu tego etapu powinny być kompletne, bez luk. Typowe przyczyny „zabrudzenia” danych to: brak określonych wartości w pewnych polach rekordów, zapisanie różnych znaczeń pod tymi samymi nazwami (wartościami), duplikowanie danych lub celowe ich powielanie.

W fazie wzbogacania do danych historycznych dostarczonych przez ich właściciela mogą zostać dodane pewne inne informacje z zewnetrznego źródła. Zabieg taki przeprowadza się w celu poszerzenia wiedzy, na podstawie której wyciągane będą wnioski.

W fazie kodowania dane są ostatecznie przygotowywane do użycia w algorytmach eksploracji. Dokonuje się tu potrzebnych przekształceń, normalizacji, sortowania i innych zabiegów mających wpływ na wiarygodność i skuteczność właściwej analizy.

Do etapu eksploracji dane przygotowane są w formie tablicy, której wiersze odpowiadają obiektom, zaś kolumny opisującym je cechom. Eksploracja danych to algorytmiczne wykrywanie zależności pomiędzy wartościami cech. Rozumiana jest jako proces wykrywania w danych interesujących dla użytkownika regularności. Metody eksploracji danych można podzielić, bardzo ogólnie, na 6 zasadniczych klas: odkrywanie asocjacji, klastrowanie (analiza skupień), odkrywanie wzorców sekwencji, odkrywanie klasyfikacji, odkrywanie podobieństw w przebiegach czasowych, wykrywanie zmian i odchyleń.

W ostatniej fazie wiedza pozyskana z danych zawartych w bazie jest przedstawiana w formie jak najbardziej zrozumiałej, wygodnej do interpretacji i wykorzystania. Prezentuje się ją najczęściej w formie graficznych wykresów, różnego rodzaju tabeli, map. Następnie wiedza jest interpretowana pod kątem zakładanych na wstępie celów.

3 Modele samoorgaznizacji w sieciach neuropodobnych

Rozwój sztucznych sieci neuropodobnych został zapoczątkowany w latach czterdziestych ubiegłego wieku badaniami nad matematycznym opisem komórki nerwowej oraz powiązaniem tego opisu z procesem przetwarzania informacji przez komórkę. Celem było dokonanie formalnego opisu procesów zachodzących w komórkach nerwowych mózgu i budowy na tej podstawie układów symulujących wybrane właściwości systemów nerwowych [4]. W następnych latach sieci okazały się wygodnym i sutecznym narzędziem, przydatnym przy realizacji wielu zadań praktycznych.

Istotną rolę w pracy systemów opartych na sieciach neuropodobnych pełni faza treningu. W trakcie treningu sieć uczy się poprawnie reagować na wzorce znajdujące się w zbiorze uczącym. Jednocześnie nabywa zdolności generalizacji, czyli oczekiwanego reagowania na wektory wejściowe, które nie były zawarte w zbiorze uczącym. W teorii sztucznych sieci neuronowych wyróżnia się dwie podstawowe metody treningu[7]: metodę nadzorowaną zwaną również treningiem z nauczycielem i metodę nienadzorowaną. W poniższej pracy uwaga zostanie skupiona na drugiej metodzie.

Sieci nie wymagające udziału nauczyciela w fazie treningu nazywane są sieciami samoorganizującymi. Pomysł na uczenie takich sieci jest bardzo prosty. Na początku nadaje się wagom sieci losowe lub określone w pewien sposób na podstawie zbioru wzorców wstępne wartości. Następnie na wejście sieci podaje się kolejne elementy zbioru wzorców. Kiedy na wejściu pojawi się obiekt, każdy neuron w pewien sposób na niego zareaguje. Uczenie polega na takiej modyfikacji wag, aby neurony, które zareagowały najsilniej na sygnał wejściowy, zaakceptowały go, w przyszłości odpowiadały jeszcze silniej, zaś neurony, które zareagowały silną odpowiedzią ujemną w przyszłości jeszcze bardziej zdecydowanie odrzucały ten sygnał [4].

Wśród mechanizmów samoorganizacji można wyróżnić dwie podstawowe klasy: mechanizm współzawodnictwa między neuronami wykorzystujący ogólnie pojętą regułę Kohonena [7] oraz mechanizm samoorganizacji oparty na regule Hebba [5].

Reguła Kohonena [7]

Przykładem sieci neuropodobnej, w której wykorzystuje się uczenie konkurencyjne jest sieć SOM Kohonena (ang. *Self-Organising Map*). W sieci tej neurony ułożone najczęściej w postać dwuwymiarowej kraty służą do uformowania odwzorowania z prze-

strzeni wielowymiarowej do dyskretnej przestrzeni dwuwymiarowej. Odwzorowanie to ma zachowywać bliskość, to znaczy punkty z przestrzeni wejść leżące w małej odległości od siebie są odwzorowywane na te same lub bliskie sobie neurony.

Każdy neuron może być traktowany jako określony punkt z przestrzeni wejść. Równocześnie każdy z nich jest powiązany z pozostałymi neuronami pewną relacją sąsiedztwa określającą topologię (strukturę) sieci.

W podstawowym algorytmie SOM liczba neuronów i topologia sieci są ustalone od początku. Jeżeli przestrzeń wejść jest przestrzenią d-wymiarową, to SOM może być zdefiniowany jako zbiór n neuronów:

$$\mathrm{W} = \{\mathrm{W}_1, \mathrm{W}_2, \ldots, \mathrm{W}_\mathrm{n}\}, \tag{1}$$

z których każdy jest d-wymiarowym wektorem:

$$\mathrm{W}_i = (w_{1i}, w_{2i}, \ldots, w_{di}), \quad i = 1,2,\ldots,n. \tag{2}$$

Uczenie sieci jest procesem iteracyjnym skojarzonym z momentami czasowymi $t = 1,2,\ldots$. Po prezentacji wektora wejściowego $\mathrm{x} = [x_1, x_2, \ldots, x_d]$ podstawowym zadaniem SOM jest wybranie spośród wszystkich neuronów, neuronu W_c, najbliższego w stosunku do x względem wybranej metryki:

$$c = \arg\min_i \|\mathrm{x} - \mathrm{W}_i\|. \tag{3}$$

Następnie wagi wygrywającego neuronu W_c oraz jego sąsiadów $N_c(t)$ są aktualizowane w taki sposób, by upodobnić je do aktualnie prezentowanego wzorca x. Zbiór sąsiadów $N_c(t)$ zależy od wybranej topologii sieci i jest ograniczany w miarę postępu procesu uczenia.

Podstawowy algorytm uczenia Kohonena ma postać:

$$w_{ij}(t+1) = w_{ij}(t) + \alpha(t) h_{ci}(t)[x_j - w_{ij}(t)], \tag{4}$$
$$i = 1,2,\ldots,n, \; j = 1,2,\ldots,d, \; t = 1,2,\ldots,$$

gdzie $0 < \alpha(t) < 1$ jest malejącym w czasie współczynnikiem uczenia, a $h_{ci}(t)$ określa funkcję sąsiedztwa względem wygrywającego neuronu W_c. Funkcja sąsiedztwa i współczynnik uczenia odgrywają istotną role w procesie uczenia.

Reguła Hebba [5]

Bazując na obserwacjach neurobiologicznych Hebb stwierdził, że powiązanie dwóch komórek nerwowych jest wzmacniane, gdy obie komórki są pobudzone w tym samym czasie. W przeniesieniu na sztuczne sieci neuronowe reguła Hebba mówi, iż jeżeli aktywny neuron A jest cyklicznie pobudzany przez neuron B, to staje się on jeszcze bardziej czuły na pobudzenie tego neuronu. Jeżeli x_A i x_B oznaczają stany aktywacji odpowiednio neuronów A i B, a w_{AB} - wagę ich połączenia synaptycznego, to regułę Hebba można przedstawić w postaci:

$$w_{AB}(k+1) = w_{AB}(k) + \alpha \cdot x_A(k) \cdot x_B(k), \tag{5}$$

gdzie α oznacza dodatni współczynnik sterujący procesem uczenia. Dla pojedynczego neuronu zapisuje się ją następująco:

$$\Delta\mathrm{w}(k) = \alpha \cdot \mathrm{x}(k) \cdot \mathrm{y}(k). \tag{6}$$

Opisuje ona zmianę wektora wag $\Delta\mathrm{w}$ w chwili k, $\mathrm{x}(k)$, $\mathrm{y}(k)$ oznaczają obraz na wejściu neuronu i stan wyjścia neuronu w chwili k,

$$\mathrm{y}(k) = \mathrm{w}^T(k) \cdot \mathrm{x}(k) = \mathrm{x}^T(k) \cdot \mathrm{w}(k). \tag{7}$$

Zakres stosowania oryginalnej reguły Hebba jest dość ograniczony. Przy $\mathrm{x}(k), \mathrm{y}(k) \geq 0$ może ona prowadzić do nieograniczonego wzrostu wielkości wag. Reguła Oji[4] poprzez wprowadzenie ograniczenia na długość wektorów wag pozwala zachować stabilność. Uzupełnia ona regułę Hebba o dodatkowy składnik zmniejszający wagi proporcjonalnie do kwadratu wartości wyjścia:

$$\Delta w(k) = \alpha \cdot y(k) \cdot (x(k) - y(k)w(k)) . \quad (8)$$

4 Zastosowanie samoorganizujących sieci neuropodobnych w pozyskiwaniu wiedzy z baz danych

Od struktury i metody uczenia sieci samoorganizującej zależy rodzaj wykonywanego przez nią zadania. Wśród możliwych zadań można wyróżnić [3]:

podobieństwo – Sieć zawiera pojedynczy element wyjściowy o aktywacji przyjmującej wartości z pewnego przedziału. Wartość wyjścia informuje na ile podobny jest obraz na wejściu do obrazu uśrednionego po dotychczasowych prezentacjach.

analiza składowych głównych – Sieć posiada wyjście wieloelementowe, a każdy z elementów wyjściowych odpowiada za jedną ze składowych głównych, według których określane jest podobieństwo. Stan aktywności każdego elementu wyjściowego jest miarą nasycenia prezentowanego obrazu danym czynnikiem (składową) głównym w stosunku do poprzednio prezentowanych obrazów uczących.

klasyfikacja – Sieć zawiera wieloelementowe wyjście o wartościach binarnych. Przy prezentacji obrazu uaktywnia się tylko jeden element wyjściowy. Obraz wejściowy zostaje zatem przyporządkowany klasie rezentowanej przez aktywny element. Zadaniem procesu uczenia jest podział wzorców uczących na klasy obrazów zbliżonych do siebie i przyporządkowanie każdej klasie osobnego elementu wyjściowego.

poszukiwanie pierwowzoru – Sieć spełnia podobną rolę jak w przypadku klasyfikacji, z tą różnicą, że na wyjściu powstaje obraz najbardziej typowy dla danej klasy. Jest to pewnego rodzaju pamięć skojarzeniowa, w której obraz wyjściowy określany jest jedynie na podstawie informacji zawartych w zbiorze obrazów uczących, bez jakiejkolwiek ingerencji zewnętrznej.

kodowanie – Wektor wyjściowy sieci jest zakodowaną wersją wektora wejściowego zachowując zawarte w nim najistotniejsze informacje w postaci minimalnej.

tworzenie map cech – Elementy warstwy wyjściowej są geometrycznie uporządkowane. Podczas prezentacji obrazu wejściowego uaktywnia się tylko jedno wyjście. Idea zawiera się w postulacie, aby podobne obrazy wejściowe uaktywniały bliskie geometrycznie elementy wyjściowe. Warstwa wyjściowa jest zatem pewnego rodzaju mapą topograficzną danych wejściowych.

5 Symulacje procesów samouczenia

W niniejszym rozdziale opisany jest przebieg i wyniki eksperymentów symulacyjnych. W pierwszym z nich wykorzystano samoorganizującą sieć Kohonena i zbór danych zebrany przez Fishera opisujący kwiaty irysa. W skład zbioru wchodzi 150 czterowymiarowych obserwacji. Każdy wektor zawiera wyniki pomiaru długości i szerokości pręcika kwiatu oraz długości i szerokości płatka. Wektory podzielone są na trzy klasy zawierające po 50 elementów. Klasy reprezentują trzy odmiany kwiatów irysa. Na potrzeby eksperymentu informacja o odmianie została usunięta. Doświadczenie polegało na dokonaniu nowego, niezależnego podziału zbioru wejściowego.

W literaturze traktującej o sieciach samoorgranizujących nie ma jednoznacznie podanego sposobu na określenie błędu sieci Kohonena. Wynika to w znacznej mierze z faktu, że sieci uczą się bez nadzoru, nie można więc mówić o złych, czy dobrych ich odpowiedziach. Jednym ze sposobów pomiaru błędu może być suma różnic pomiędzy wektorami ze zbioru uczącego, a najbliższymi względem nich wektorami wag neuronów. Suma ta może w pewnym stopniu informować o zdolności do grupowania

danych. O jakości sieci może również świadczyć ilość tak zwanych martwych neuronów. Przydatna jest także wszelka informacja *a priori* na temat badanych danych.

Uwzględniając powyższe wytyczne wybrana została sieć Kohonena o wymiarach 2x2 i optymalnych parametrach. Następnie przy jej pomocy dokonano podziału zbioru wejściowego. Porównaie wyników osiągniętych przy użyciu sieci z oryginalnym podziałem zaprezentowano w tabeli 1.

	SETOSA	VIRGINIC	VERSICOL
(0,0)			42
(0,1)	50		
(1,0)		50	5
(1,1)			3

Tab. 1. Porównanie podziału dokonanego przy użyciu sieci Kohonena z oryginalnym podziałem Fishera, zbiór Iris

Tab. 1. The comparison of divisions: made with the use of Kohonen's net and the original Fisher's division, the Iris data set

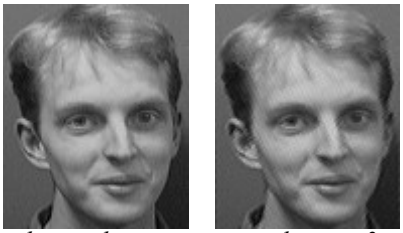
Jak widać dokonany przez nas podział w znacznym stopiniu pokrywa się z podziałem dokonanym przez Fishera. Jedynie część obserwacji należących do odmiany *Versicol* zostało sklasyfikowanych niezgodnie z pierwotnym podziałem.

W drugim eksperymencie wykorzystano sieć analizy składowych głównych. Zbiór wejściowy utworzono na podstawie pliku graficznego w formacie PGM (*face.pgm*) dzieląc dane opisujące obraz na pięcioelementowe fragmenty. Następnie sieć poddano procesowi uczenia, którego rezultatem było wyznaczenie 3 pierwszych składowych głównych w zbiorze wejściowym. W tabeli 2 zawarto wyliczone składowe oraz ilość zawartych w nich informacji. Rysunek 1 przedstawia oryginalny obraz i obraz odtworzony na podstawie wyliczonych 3 pierwszych składowych głównych.

Lp.	Współczynniki składowych głównych	Ilość informacji
1	- 0,474*x1 - 0,476*x2 - 0,513*x3 - 0,42*x4 - 0,331*x5	89,16%
2	- 0,254*x1 - 0,31*x2 - 0,283*x3 + 0,356*x4 + 0,795*x5	6,832%
3	0,624*x1 + 0,195*x2 - 0,585*x3 - 0,409*x4 + 0,25*x5	2,473%

Tab. 2. Składowe główne wyznaczone w zbiorze danych utworzonym na podstawie obrazu face.pgm

Tab. 2. The principial components calculated in data set made on the basis of the picture fase.pgm



Rys. 1. Obraz face.pgm i obraz odtworzony na podstawie 3 pierwszych składowych głównych wyznaczonych przy użyciu sieci analizy składowych głównych

Fig. 1. The picture face.pgm and the picture reconstructed on the basis of 3rd firsts principial components calculeted with the use of the principial components analysis net

6 Podsumowanie i wnioski

Przeprowadzone eksperymenty poparte wcześniejszą wiedzą teoretyczną pozwalają na stwierdzenie, iż samoorganizujące sieci neuropodobne stanowią cenne narzędzie w pozyskiwaniu wiedzy z baz danych. Z drugiej strony są tylko narzędziem. Niestety nie są w stanie, po wskazaniu źródła danych, automatycznie wygenerować wiedzę spełniającą oczekiwania, przydatną i istotną statystycznie. Uzyskanie zadowalających wyników wymaga wiele wysiłku i pomysłowości. Nie można również przecenić udziału w procesie ludzkiego eksperta z badanej dziedziny, który we właściwy sposób oceni przydatność i zinterpretuje otrzymane wyniki.

Badania nad sieciami samoorganizującymi są istotne także z innego względu. Wszelkie procesy życiowe zawierają w sobie elementy samoorganizacji. Symulowanie tego zjawiska być może przyczyni się do lepszego zrozumienia niektórych mechanizmów zachodzących w systemach naturalnych.

Literatura

1. Frawley W.J., Piatetsky-Shapiro G., Matheus C.: *Knowledge Discovery In Databases;* AAAI Press/MIT Press, Cambridge, MA., 1991, pp. 1-30
2. Goldberg D.E.: *Algorytmy genetyczne i ich zastosowania*, WNT, Warszawa 1995
3. Hertz J., Krogh A., Palmer R.G.: *Introduction to the Theory of Neural Compuutation.*; Addison – Wesley Publ. Com., 1991; Wydanie polskie: Hertz J., Krogh A., Palmer R.G.*: Wstęp do teorii obliczeń neuronowych.*; WNT, Warszawa 1993
4. Korbicz J., Obuchowicz A., Uciński D.: *Sztuczne sieci neuronowe. Podstawy i zastosowania.*; Akademicka Oficyna Wydawnicza PLJ, Warszawa 1994; str. 59 – 93
5. Osowski S.: *Sieci neuronowe w ujęciu algorytmicznym.*; WNT Warszawa 1996
6. Quinlan J.R.: *Introduction of Decision Trees.*; Machine Learning, vol. 1, 1986, pp. 81-106,
7. *Sieci neuronowe*, pod redakcją Macieje Nałęcza; Akademicka Oficyna Wydawnicza Exit, Warszawa 2000

Streszczenie

W artykule przedstawiono zagadnienie pozyskiwania wiedzy z baz danych ze szczególnym naciskiem na zastosowanie samoorgranizujących sieci neuropodobnych w tej dziedzinie. Opisane zostały kolejne etapy procesu. Omówiono dwa mechanizmy samoorganizacji: mechanizm współzawodnictwa między neuronami zgodnie z regułą Kohonena oraz mechanizm oparty na regule Hebba. Podane zostały przykłady sieci neuropodobnych pracujących w oparciu o poszczególne mechanizmy samoorganizacji, sieć SOM Kohonena i sieć analizy składowych głównych. Przedstawiono także wyniki eksperymentów symulacyjnych przeprowadzonych przy użyciu wymienionych sieci.

Knoledge discovery in databases whit the aid of artificial neural networks

Summary

It was presented in the article the problem of knowledge discovery in databases emphatically the use of self-organising artificial neural networks in the domain. It was described steps of the process. It was discussed two self-organising types: the type of the competition between neurones by Kohonen's rule and the type by Hebb's rule. It was made known the examples of artificial neural networks worked on the basis of particular self-organising types: the Kohonen's SOM net and the principal components analysis net. It was presented the results of simulation experiments with the use of mentioned networks.