



**INSTYTUT BADAŃ SYSTEMOWYCH
POLSKIEJ AKADEMII NAUK**

TECHNIKI INFORMACYJNE TEORIA I ZASTOSOWANIA

Wybrane problemy
Tom 5 (17)

poprzednio

**ANALIZA SYSTEMOWA W FINANSACH
I ZARZĄDZANIU**

Pod redakcją
Andrzeja MYŚLIŃSKIEGO

Warszawa 2015



**INSTYTUT BADAŃ SYSTEMOWYCH
POLSKIEJ AKADEMII NAUK**

**TECHNIKI INFORMACYJNE
TEORIA I ZASTOSOWANIA**

Wybrane problemy
Tom 5 (17)

poprzednio

**ANALIZA SYSTEMOWA W FINANSACH
I ZARZĄDZANIU**

Pod redakcją
Andrzeja Myślińskiego

Warszawa 2015

Wykaz opiniodawców artykułów zamieszczonych
w niniejszym tomie:

Dr inż. Tatiana JAWORSKA

Dr inż. Jan OWSIŃSKI

Dr hab. inż. Andrzej MYŚLIŃSKI, prof. PAN

Prof. dr hab. inż. Piotr SIENKIEWICZ

Prof. dr hab. inż. Andrzej STRASZAK

Dr hab. Dominik ŚLĘZAK, prof. UW

Prof. dr hab. inż. Stanisław WALUKIEWICZ

Copyright © by Instytut Badań Systemowych PAN
Warszawa 2015

ISBN 83-894-7558-8

METODY AGREGACJI WYNIKÓW LOKALNYCH W SIECIACH KOMPARATORÓW

Łukasz Sosnowski

Studia Doktoranckie IBS PAN i Dituel Sp. z.o.o.
e-mail: *l.sosnowski@dituel.pl*

Streszczenie. Artykuł przedstawia skrót najważniejszych informacji teoretycznych o sieciach komparatorów, skupiając uwagę na problemie agregacji w sieciach homogenicznych i heterogenicznych. Opisuje podstawowe mechanizmy używane w modelowaniu jak również przedstawia różne sposoby radzenia sobie z problemem agregacji począwszy od metod statystycznych poprzez metody oparte o algorytmy wyborcze aż do metod inspirowanych mereologią przybliżoną (ang. rough mereology).

Słowa kluczowe: sieci komparatorów, podobieństwo obiektów złożonych, agregacja wyników lokalnych, identyfikacja i klasyfikacja

1 WPROWADZENIE

Sieć komparatorów to powiązane logicznie elementy modelujące podobieństwo obiektów złożonych reprezentujących zjawiska, procesy lub inne byty. Struktura sieci umożliwia rozłożenie zadania analizy obiektu złożonego na czynniki pierwsze, jego lokalne przetworzenie, zagregowanie wyników częściowych oraz zaprezentowanie wyników w postaci wektora bliskości wskazującego na podobieństwo obiektu wejściowego względem pewnego rodzaju bazy wiedzy, jaką jest zbiór obiektów referencyjnych. Dodatkowo, proces konstrukcji sieci może uwzględniać zadanie minimalizacji liczności zbioru referencyjnego, a także wykorzystywać wiedzę o istocie struktury badanych obiektów. Przypadek pierwszy modelują sieci homogeniczne, a drugi sieci heterogeniczne [2].

Wspomniane sieci znajdują zastosowanie m. in. do modelowania procesów rozpoznawania oraz identyfikacji obiektów przy użyciu podobieństwa. Te dwa procesy, stanowią główne obszary zastosowań sztucznej inteligencji (SI). Sieci komparatorów poszerzają zatem zbiór dostępnych metod modelowania tych skomplikowanych zagadnień, przy jednoczesnym

zapropnowaniu spójnej metodyki postępowania dla różnych typów problemów.

Celem niniejszej pracy jest podsumowanie badań nad komparatorami obiektów złożonych, zwłaszcza w aspekcie syntezy wyników uzyskiwanych podczas przetwarzania sieci komparatorów. Praca ta bazuje na rezultatach otrzymanych w ramach przygotowywanej rozprawy doktorskiej i stanowi swoisty wyciąg z opracowanych tam zagadnień. Rozdział drugi przedstawia w formie skróconej podstawy matematyczne sieci komparatorów. Rozdział trzeci przedstawia problematykę syntezy danych oraz proponowane rozwiązania. Ostatni rozdział stanowi podsumowanie najważniejszych elementów poruszanych w pracy.

2 PODSTAWY MATEMATYCZNE SIECI KOMPATORÓW

Sieci operują elementem zwanym komparatorem, który został obszernie opisany w kilku poprzednich publikacjach [3–5]. Jego rolą jest wskazanie podobieństwa obiektu wejściowego do obiektów referencyjnych pod względem pewnej ustalonej, często elementarnej cechy. Komparator może być interpretowany jako funkcja postaci:

$$\mu_{com} : U \times 2^{ref} \rightarrow [0, 1]^{|ref|} \quad (1)$$

gdzie U jest zbiorem obiektów wejściowych, a ref jest zbiorem referencyjnym ustalonym dla zadanego obiektu wejściowego.

Działanie sieci komparatorów może być interpretowane jako funkcja postaci:

$$\mu_{ref}^{net} : U \rightarrow [0, 1]^{|ref_{out}|}, \quad (2)$$

przyjmującą jako argument $u \in U$, gdzie U jest zbiorem obiektów wejściowych sieci, a $|ref_{out}|$ jest licznością zbioru referencyjnego ustalonego dla sieci warstwy wyjściowej sieci. Przeciwdziedziną tej funkcji jest przestrzeń wektorów bliskości. Wektor z tej przestrzeni posiada wartości będące podobieństwami konkretnych obiektów referencyjnych do obiektu wejściowego u oraz oznaczany jest przez v . Ustalając porządek na ref , który jest zbiorem referencyjnym ustalonym dla danej sieci, otrzymujemy

$$ref = \{y_1, \dots, y_{|ref|}\}. \quad (3)$$

A zatem wartość funkcji będzie postaci:

$$\mu_{ref}^{net}(u) = \langle sim_g(u, y_1), \dots, sim_g(u, y_{|ref|}) \rangle, \quad (4)$$

gdzie $sim_g(u, y_i)$ jest globalną wartością podobieństwa w danej sieci pomiędzy obiektem wejściowym u , a konkretnym obiektem referencyjnym y_i . Podobieństwo to jest nazywane globalnym, gdyż uwzględnia w sobie wszystkie podobieństwa lokalne obliczone przez poszczególne komparatory sieci oraz wykonane operacje przetwarzania informacji o lokalnym podobieństwie przy przejściu obiektu wejściowego przez poszczególne jej warstwy.

Sieć zbudowana jest z warstw. Istnieją trzy rodzaje warstw: wejściowa, pośrednia i wyjściowa. Warstwa grupuje komparatory ze względu na pewien kontekst związany z przetwarzanymi cechami lub strukturą obiektu. Warstwa odpowiedzialna jest za wykonywanie równoległych, lokalnych porównań przy użyciu znajdujących się w niej komparatorów oraz syntezę uzyskanych wyników do postaci lokalnej decyzji dotyczącej doboru obiektów referencyjnych do dalszego przetwarzania. Warstwa może być zapisana w postaci funkcji:

$$\mu_{ref}^{layer} : U \rightarrow [0, 1]^{|ref_{out}|}, \quad (5)$$

gdzie u jest ustalonym obiektem wejściowym, a ref_{out} ustalonym dla warstwy wyjściowym zbiorem referencyjnym. Oznacza to, że wewnątrz warstwy mogą być używane różne zbiory referencyjne dla poszczególnych komparatorów. Jednakże w procesie agregacji i translacji następuje synteza i przekształcenie obliczonych podobieństw lokalnych pochodzących z poziomu komparatorów, na podobieństwa do obiektów referencyjnych ustalonego wyjściowego zbioru referencyjnego warstwy. W celu przeprowadzenia tego przetwarzania używana jest macierz translacji opisana w [6]. Funkcja (5) stanowi złożenie poszczególnych funkcji składowych takich jak: funkcji komparatora (1), funkcji agregującej dane w warstwie oraz funkcji translacji spełniającej często funkcję filtracji na poziomie danych lokalnie zagregowanych.

Każda sieć komparatorów składa się z jednej warstwy wejściowej, jednej wyjściowej i dowolnej liczby warstw pośrednich [2]. W warstwie wejściowej oraz pośrednich znajdują się komparatory opisane funkcją (1) oraz agregatory lokalne i translatory. Warstwa wyjściowa zawiera agregator globalny oraz translator [7].

Agregator lokalny ma za zadanie syntezę wyników częściowych. Agregator zależy od rodzaju przetwarzanych obiektów referencyjnych (strukturalnych lub monolitycznych). W ogólnej postaci może być zapisany jako:

$$f_{ref_{out}}^{agg} : [0, 1]^{ref_1} \times \dots \times [0, 1]^{ref_k} \rightarrow [0, 1]^{|ref_{out}|}, \quad (6)$$

gdzie k jest liczbą komparatorów w danej warstwie co jednoznacznie przekłada się na liczbę sygnałów wchodzących do agregatora. Natomiast ref_{out} jest wyjściowym zbiorem referencyjnym obowiązującym dla warstwy, zawierającym elementy powiązane z obiektami referencyjnymi lokalnych zbiorów referencyjnych (ref_i , gdzie $i = 1, \dots, k$) używanych przez komparatory poprzez tzw. reguły składania. Reguły bazują na strukturze obiektu i definiują w jaki sposób z podobieństw dotyczących pod-obiektów wyliczyć podobieństwo większego (ogólniejszego) obiektu.

Kolejnym elementem występującym w warstwie jest translator, którego zadaniem jest konwersja wyników jednej warstwy do postaci akceptowalnej przez kolejną warstwę [4]. Translator zdefiniowany jest w postaci macierzy przejścia:

$$M_{ref_i}^{ref_k} = [m_{ij}], \quad (7)$$

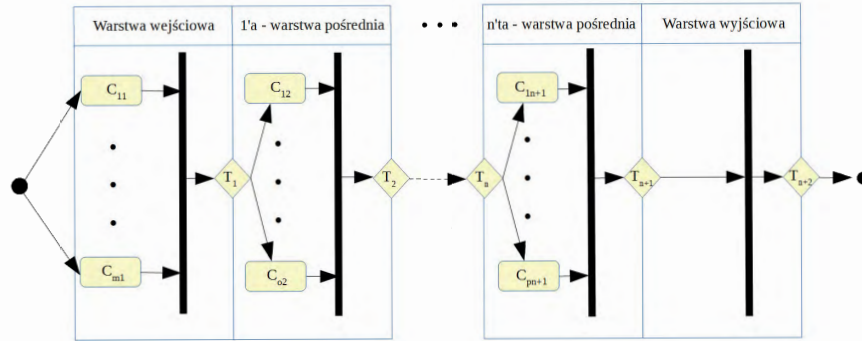
gdzie $i \in \{1, \dots, n\}$, $j \in \{1, \dots, m\}$. Macierz $M_{ref_i}^{ref_k}$ definiuje mapowanie obiektów zbioru ref_k na obiekty zbioru ref_i . W praktyce zbiór ref_k jest sumą zbiorów referencyjnych poszczególnych komparatorów warstwy, a ref_i zbiorem docelowym (z punktu widzenia translacji). Wartościami mapowania mogą być wartości z przedziału $[0, 1]$. W zależności od rodzaju relacji wzajemnej, w jakiej są oba zbiory (uogólnienie lub dekompozycja), macierz przyjmuje różne wartości. Dla dekompozycji jednego elementu zbioru ref_k przeważnie przypada wiele elementów ref_i , co przejawia się poprzez występowanie wielu wartości 1 w wierszu odpowiadającym obiektowi z ref_k . W przypadku uogólnienia wartości macierzy przeważnie są mniejsze od 1.

W warstwie wyjściowej znajduje się agregator globalny. Różnica pomiędzy agregatorem lokalnym a globalnym polega na zasięgu przetwarzanych wyników. Lokalny przetwarza jedynie wyniki częściowe jednej warstwy podczas gdy globalny umożliwia przetwarzanie wszystkich osiągniętych wyników lokalnych w całej sieci. Dla przypadku gdzie w każdej warstwie mamy ten sam zbiór referencyjny, agregator ten można zapisać w postaci funkcji:

$$f_{ref}^{agg} : ([0, 1]^{|ref|})^m \rightarrow [0, 1]^{|ref|}, \quad (8)$$

gdzie m jest liczbą wszystkich komparatorów sieci i jednocześnie liczbą wejść agregatora globalnego.

Dla przypadku różnych zbiorów referencyjnych w warstwach rozważmy wyjściowy zbiór referencyjny ref_{out} . Każdy element $y \in ref_{out}$ rozbija się na y_1 w zbiorze referencyjnym ref_1 , y_2 w zbiorze referencyjnym ref_2 i tak aż do y_m w zbiorze referencyjnym ref_m . Dla elementu wejściowego $u \in U$ mamy wyliczone wektory podobieństw do elementów ze zbiorów



Rys. 1. Ogólny schemat sieci komparatorów. Schemat nie jest wykonany przy użyciu UML, a jedynie używa dostępnej tam analogicznej notacji. Użyte skróty: C_{ji} - komparatory, T_j - translatory. Symbole: owale - komparatory, pionowe pogrubione czarne linie - agregatory, romby - translatory.

ref_1, \dots, ref_m . W celu otrzymania wyniku zagregowanego używamy funkcji postaci:

$$f^{agg} : [0, 1]^{|ref_1|} \times \dots \times [0, 1]^{|ref_m|} \rightarrow [0, 1]^{|ref_{out}|}, \quad (9)$$

Jak widać wzór (9) stanowi analogię do wzoru (6) przedstawionego dla agregatora lokalnego. Różnica polega na wzięciu pod uwagę innej liczby komparatorów. W przypadku wzoru (9) są to wszystkie agregatory a sieci, a w przypadku agregacji lokalnej jedynie komparatory danej warstwy.

Biorąc pod uwagę powyższe definicje, sieć komparatorów można przedstawić jako złożenie funkcji warstw, w następującej postaci:

$$\mu_{ref_{out}}^{net}(u) = \mu_{ref_{out}}^{layer-out}(\mu_{ref_{k-1}}^{layer-int}(\dots(\mu_{ref_1}^{layer-in}(u))\dots)), \quad (10)$$

gdzie ref_i są ustalonymi dla odpowiednich warstw zbiorami referencyjnymi a ref_{out} jest zbiorem referencyjnym wyjściowym dla całej sieci. Ogólny schemat sieci przedstawiono na Rys. 1.

3 ZAGADNIENIE AGREGACJI WYNIKÓW CZĘŚCIOWYCH

Zgodnie z definicją sieci komparatorów i wzorem (2) na wyjściu sieci pojawia się wektor bliskości. Wymiar wektora jest taki jaka jest liczność zbioru referencyjnego wyjściowego określonego dla sieci. Problem agregacji wyników częściowych polega na przekształceniu macierzy zbudowanej z lokalnych wektorów bliskości w globalny wektor bliskości. Pod pojęciem globalnego wektora bliskości rozumiany jest zbiór lokalnych wektorów bliskości połączony w pojedynczy wektor w taki sposób, że wybrane

są wartości podobieństwa obliczone na podstawie funkcji agregacji zgodnej z wzorami (8) lub (9). Proces ten jest optymalizacją doboru elementów zbioru referencyjnego do rankingu, budowanego z wartości podobieństwa pomiędzy obiektami. Macierz zbudowana z wektorów lokalnych jest wymiaru $m \times n$, gdzie m jest liczbą komparatorów a n jest mocą zbioru referencyjnego traktowanego jako suma podzbiorów lokalnych. Zbiór ten budujemy poprzez dodawanie elementów kolejnych lokalnych zbiorów referencyjnych niewystępujących dotychczas dodawanych zbiorach. W przypadku agregatora lokalnego brane pod uwagę są wartości dla wszystkich komparatorów z danej warstwy. W przypadku agregatora globalnego dostępne są wartości wszystkich komparatorów całej sieci, lecz mogą być brane pod uwagę dowolne z nich, w zależności od intencji projektanta sieci. Załóżmy, iż w danej warstwie mamy n komparatorów. Dla ustalonego zbioru referencyjnego ref dla warstwy lub sieci, każdy z komparatorów zwróci wektor bliskości rzędu m . Z poszczególnych wektorów bliskości komparatorów $\mu_{com_1}, \mu_{com_2}, \dots, \mu_{com_n}$ budujemy macierz, traktując każdy kolejny wektor jako wiersz macierzy A_m^n . Wartości danego wektora wstawiamy w pozycje odpowiadające odpowiednim obiektom referencyjnym. Pozostałe pozycje wiersza macierzy uzupełniamy wartościami zerowymi. Poszczególne wartości macierzy odpowiadają wartością funkcji podobieństwa danego komparatora oraz zadanego obiektu referencyjnego, czyli poszczególnym współrzędnym wyliczonego wektora bliskości. Dzięki temu uzyskujemy obraz wyników poszczególnych komparatorów dla odpowiadających im obiektów referencyjnych. Dla przedstawionego przykładu macierz wygląda następująco:

$$A_m^n = \begin{bmatrix} sim_1(u, y_1) & sim_1(u, y_2) & \dots & sim_1(u, y_m) \\ sim_2(u, y_1) & sim_2(u, y_2) & \dots & sim_2(u, y_m) \\ \dots & \dots & \dots & \dots \\ sim_n(u, y_1) & sim_n(u, y_2) & \dots & sim_n(u, y_m) \end{bmatrix} \quad (11)$$

Bazując na macierzy (11), agregacja polega na dokonaniu operacji optymalizacyjnej na wartościach poszczególnych kolumn macierzy. Poniżej zostały opisane wybrane metody możliwe do zastosowania w celu dokonania tej optymalizacji.

3.1 METODY STATYSTYCZNE

Intuicyjnym podejściem do agregacji wyników lokalnych do postaci globalnego wyniku jest zastosowanie metod statystycznych. Poszczególne kolumny macierzy (11) reprezentują wyniki badania podobieństwa obiektu

wejściowego do konkretnego obiektu referencyjnego. Poszczególne wartości w wierszach reprezentują jednak różne badane cechy. Dla badanych cech szukana jest jedna wartość reprezentująca najlepiej ogólne (sumaryczne) podobieństwo uwzględniające po części każdą z badanych cech. Jednym z podejść jest estymacja wartości podobieństwa poprzez wartość średnią. Poniżej przedstawiono wybrane funkcje agregacji skonstruowane przy użyciu miar statystycznych, które mogą być użyte zarówno w sieciach komparatorów typu I (sieci homogeniczne) jak i typu II (sieci heterogeniczne).

Domyślną metodą agregacji dla sieci komparatorów jest średnia arytmetyczna. Polega na wyliczeniu wartości średniej z podobieństw poszczególnych obiektów referencyjnych biorąc pod uwagę wartości pochodzące od różnych komparatorów (różnych cech). Stanowi operację uśredniania wartości kolumn macierzy, co może być zapisane w następującej postaci:

$$f_{mean}^{agg}(A_m^n) = \left\langle \frac{\sum_{i=1}^n sim_i(u, y_1)}{n}, \dots, \frac{\sum_{i=1}^n sim_i(u, y_m)}{n} \right\rangle, \quad (12)$$

gdzie $\langle \dots, \dots, \dots \rangle$ oznacza wektor.

Nieco zmodyfikowaną metodą jest agregacja oparta na wagach, wskazujących które cechy (komparatory) są bardziej istotne od pozostałych. W tym przypadku również wyliczana jest średnia wartość z poszczególnych kolumn macierzy, lecz tym razem poszczególne wartości są przemnożone przez wagi. Ogólny wzór tego rodzaju agregacji ma następującą postać:

$$f_{wa}^{agg}(A_m^n) = \left\langle \frac{\sum_{i=1}^n w_i sim_i(u, y_1)}{\sum_{i=1}^n w_i}, \dots, \frac{\sum_{i=1}^n w_i sim_i(u, y_m)}{\sum_{i=1}^n w_i} \right\rangle \quad (13)$$

Ten rodzaj agregacji jest bardzo efektywny, pod warunkiem posiadania dobrze dobranych wag. Daje możliwość wyuczenia wag, co podnosi dodatkowo atrakcyjność tej metody [7].

Innym wariantem bazującym na statystyce jest średnia harmoniczna. Analogicznie używana jest do policzenia wartości z elementów kolumny macierzy (11). Wzór opisujący ten typ agregacji jest następującej postaci:

$$f_{hm}^{agg}(A_m^n) = \left\langle \frac{n}{\sum_{i=1}^n \frac{1}{sim_i(u, y_1)}}, \dots, \frac{n}{\sum_{i=1}^n \frac{1}{sim_i(u, y_m)}} \right\rangle \quad (14)$$

Wprowadzając dodatkowo wagi dla poszczególnych elementów próby, otrzymujemy wariant z użyciem średniej harmonicznej ważonej. Może on zostać przedstawiony w następującej postaci:

$$f_{hw}^{agg}(A_m^n) = \left\langle \frac{\sum_{i=1}^n w_i}{\sum_{i=1}^n \frac{w_i}{sim_i(u, y_1)}}, \dots, \frac{\sum_{i=1}^n w_i}{\sum_{i=1}^n \frac{w_i}{sim_i(u, y_m)}} \right\rangle \quad (15)$$

W tym przypadku również napotykamy na problem wyznaczenia wag w_i , co jednocześnie jest dodatkowym atutem tej metody ze względu na możliwość ich uczenia przy pomocy znanych metod z innych dziedzin, np. przy wykorzystaniu algorytmów genetycznych [8].

Jako ostatnie zostały wybrane przykłady średniej geometrycznej oraz wariantu ważonego tej funkcji. Wzory dla tych przypadków funkcji agregujących przedstawiają się następująco:

$$f_{gm}^{agg}(A_m^n) = \langle \sqrt[n]{\prod_{i=1}^n sim_i(u, y_1)}, \dots, \sqrt[n]{\prod_{i=1}^n sim_i(u, y_m)} \rangle \quad (16)$$

dla średniej geometrycznej oraz

$$f_{gw}^{agg}(A_m^n) = \langle (\prod_{i=1}^n sim_i(u, y_1)^{w_i})^p, \dots, (\prod_{i=1}^n sim_i(u, y_m)^{w_i})^p \rangle \quad (17)$$

dla średniej geometrycznej ważonej, gdzie $p = \frac{1}{\sum_{i=1}^n w_i}$.

Bazując na statystyce można by zaproponować jeszcze wiele innych przykładów funkcji agregacji. Jak widać z powyższych przykładów metoda działania jest w każdym przypadku analogiczna. Punktem wyjścia jest macierz wyników częściowych (11), na kolumnach której dokonywane są operacje uśredniania. Chociaż metody te są względnie proste, to w wielu przypadkach wyniki uzyskane za ich pomocą są wystarczająco dobre [9]. Niewątpliwą ich zaletą jest niska złożoność obliczeniowa oraz niewyrafinowane metody ich obliczania. Daje to możliwość szybkiego i sprawnego przygotowania wyników do celów dalszych obliczeń.

3.2 METODY BAZUJĄCE NA GŁOSOWANIU

Agregacja wyników lokalnych jest syntezą danych niekoniecznie polegającą na typowych funkcjach agregujących. Przykładem jest metoda optymalizacji zaczerpnięta z zupełnie innej dziedziny jaką są algorytmy wyborcze. Metoda opiera się na algorytmach przeprowadzania głosowania, które prowadzą do powstania rankingów wyborczych. Metody te mają na celu dobre odwzorowanie preferencji wyborców, tzn. wybór takich kandydatów, którzy są popierani przez jak największą liczbę wyborców i w jak najwyższym stopniu. Część z tych metod spełnia kryterium Condorceta [10], które gwarantuje, że wybrany kandydat ma poparcie większe od poparcia każdego innego. Jak widać metody te, chociaż używane w zupełnie odmiennej dziedzinie, sprowadzają się do optymalizacji pewnej funkcji

reprezentującej poglądy wyborców. Metoda ta operuje dwoma podstawowymi pojęciami: kandydat oraz wyborca. Pierwsze pojęcie reprezentuje obiekt poddawany pod głosowanie i wchodzący w skład potencjalnych rozwiązań. Wyborca natomiast to obiekt oddający głos na kandydata. W zależności od metody głosowania, różnie mogą wyglądać metody oddawania głosów. W niektórych przypadkach dany wyborca oddaje tylko jeden głos, w innych może oddać ich więcej. Różne są również metody liczenia głosów. W każdym jednak przypadku efektem końcowym jest ranking szeregujący kandydatów i wyznaczający miejsca w rankingu zgodnie z przyjętą regułą liczenia głosów (ordynacją wyborczą [11]). Ordynacja wyborcza określa również metodę wyznaczania zwycięzców, tzn. jak oddane głosy przekładają się na decyzję o wyborze kandydata.

W przypadku zastosowania tych metod do problemów optymalizacji, również będziemy operowali podstawowymi pojęciami używanymi przez te metody. Będziemy przypisywać znaczenie (rolę) dla kandydatów oraz dla wyborców. W sieciach komparatorów problem optymalizacji związany jest z agregatorami. W tych elementach składowych sieci możliwe jest zastosowanie algorytmów wyborczych jako metod wyboru rozwiązań. Podobnie jak we wcześniejszych opisywanych rodzajach agregatorów, po przetwarzaniu obiektów przez komparatory dostajemy wiele wektorów bliskości. Należy dokonać wyboru w taki sposób aby na końcu otrzymać jeden wektor bliskości. Kandytaci są reprezentowani przez obiekty referencyjne. To z nich powstają rozwiązania, na ich podstawie konstruujemy wektory bliskości. Każdy wektor posiada ponadto wartości określające podobieństwo. W przypadku algorytmów wyborczych wartość ta będzie interpretowana jako siła głosów oddanych na kandydatów. Komparatory utożsamiane są z wyborcami. To komparatory oddają głosy na kandydatów, przydzielając odpowiedni poziom podobieństwa dla poszczególnych obiektów referencyjnych. W ten sposób mamy określone dwa podstawowe pojęcia niezbędne do przeprowadzenia wyborów. W dalszej części zostanie opisane dostosowanie wybranych algorytmów wyborczych do użycia w kontekście sieci komparatorów.

GŁOSOWANIE WIĘKSZOŚCIOWE - Głosowanie większościowe jest bardzo prostą metodą, łatwo implementowalną w sieciach komparatorów. Oddanie głosu na kandydata realizowane jest poprzez interpretację wartości podobieństw w wektorze bliskości. Dla wektora bliskości postaci

$$\mu_{com}(u, ref) = \langle sim(u, y_1), \dots, sim(u, y_{|ref|}) \rangle \quad (18)$$

pochodzącego od danego komparatora na poszczególnych współrzędnych znajdują się wartości podobieństwa związane z danym obiektem referencyjnym. Na wejściu znajduje się tyle wektorów ile jest komparatorów. Każdy wektor bliskości traktowany jest jako oddany głos. W tego rodzaju głosowaniu oddawany jest maksymalnie jeden głos przez danego wyborcę. Przyjmuje się, iż głos danego komparatora zostaje oddany na najwyższą wartość podobieństwa w wektorze bliskości. Funkcja określająca fakt głosowania dla j 'tego komparatora może być zapisana następująco:

$$f_{vote}^j(y_k) = \begin{cases} 1 & : k = \operatorname{argmax}_{i=1}^{|ref|} v[i], i, k \in I \\ 0 & \text{w przeciwnym przypadku} \end{cases} \quad (19)$$

gdzie I jest zbiorem indeksów współrzędnych wektora bliskości v , a j jest numerem komparatora. Powstaje jednak problem, iż wektor może posiadać wiele wartości takich samych, a w szczególności może posiadać wartość 1 na każdej współrzędnej. Następuje tu pierwsze odstępstwo od oryginalnej metody głosowania. Dopuszczamy jednak takie rozwiązania, gdyż wartość 1 jest wartością świadczącą o nierozróżnialności względem zadanej cechy, a więc jest wartością wielce pożądaną. Każdy z komparatorów oddaje swój głos analogicznie. Na końcu następuje konstrukcja rankingu obiektów referencyjnych względem przypisanej sumy głosów, im więcej tym wyżej w rankingu. Funkcja określająca wartość głosów branych do rankingu może być zapisana w postaci następującej funkcji:

$$score(y_i) = \sum_{j=1}^n f_{vote}^j(y_i) \quad (20)$$

gdzie n jest liczbą komparatorów. Rozwiązaniem jest obiekt o najwyższej liczbie głosów policzonych zgodnie ze wzorem (20). W przypadku sieci komparatorów dozwolony jest wynik składający się z więcej niż jednego zwycięzcy. To jest drugie odstępstwo od oryginalnej metody głosowania lecz w tym przypadku jest to porządana własność. Zwycięzcy głosowania dostają wartości 1 jako wartość współrzędnej wektora bliskości a pozostałe obiekty otrzymują wartość 0.

Metoda ta posiada złożoność obliczeniową rzędu $O(N)$. Z tego punktu widzenia jest bardzo dobra do użycia. Jej walory optymalizacyjne są jednak ograniczone.

METODA BORDY - W metodzie Bordy z każdego wektora bliskości konstruowany jest ranking dla obiektów referencyjnych na podstawie punktów

przyznanych każdemu kandydatowi według jego wartości podobieństwa. Punkty przyznawane są wg. następującej reguły:

$$f_{vote}^j(y_i) = |ref| + 1 - RankPos(y_i), \quad (21)$$

gdzie $|ref|$ jest liczbą wszystkich obiektów referencyjnych, j jest indeksem komparatora, $RankPos$ jest natomiast funkcją zwracającą miejsce w rankingu wynikającego z posortowania malejąco obiektów względem ich wartości podobieństwa. Obiekty posiadające tę samą wartość podobieństwa otrzymują tę samą wartość pozycji (ex-equo).

Z każdego z komparatorów otrzymywany jest ranking punktowy. Dla każdego obiektu referencyjnego sumowane są wartości punktowe z poszczególnych rankingów zgodnie z wzorem:

$$score(y_i) = \sum_{j=1}^n f_{vote}^j(y_i) \quad (22)$$

gdzie $score_n$ oznacza wartość punktową przyznaną obiektowi y_i przez n -ty komparator. W ten sposób otrzymujemy ranking końcowy, na podstawie którego wyznaczany jest końcowy wektor bliskości. Wartości wektora mogą być wyznaczone przy użyciu następującej funkcji:

$$v[i] = \frac{score(y_i)}{n * |ref|} : |ref| > 0, n > 0 \quad (23)$$

Metoda ta charakteryzuje się bardzo dobrymi własnościami obliczeniowymi. Jej złożoność obliczeniowa jest rzędu $O(N)$. Istotnym elementem jest metoda przyznawania punktów za miejsce w rankingu. W tym przypadku zaproponowano model liniowy, aczkolwiek częściej spotykane są modele nieliniowe faworyzujące wyższe miejsca w rankingu.

METODA COPELAND'A - Algorytm ten jest metodą turniejową rozpatrującą indywidualne preferencje wyborców. Może być stosowana tylko dla przypadków, gdy istnieje więcej niż jeden kandydat. Kandydatami są obiekty referencyjne, a zatem rozważania będą ograniczone do przypadków gdy zbiór referencyjny spełnia warunek: $card(ref) > 1$. Wyborcami są komparatory warstwy lub całej sieci (w zależności od rodzaju agregatora). Dla każdego komparatora tworzona jest tabela wyników przedstawiająca jego głosy na poszczególnych kandydatów. W celu wyliczenia punktów dla poszczególnych kandydatów generowany jest iloczyn kartezyjski ze zbioru obiektów referencyjnych $ref \times ref$. Dla każdej powstałej w ten

sposób pary przydzielamy punkty: +1 za zwycięstwo, -1 za porażkę, 0 za remis. Wynik dla pojedynczej pary określany jest na podstawie wartości podobieństwa danego obiektu względem obiektu wejściowego u , wg. następującej reguły:

$$f_{result}(y_i, y_j) = \begin{cases} 1 & : \text{sim}(u, y_i) > \text{sim}(u, y_j) \\ 0 & : \text{sim}(u, y_i) = \text{sim}(u, y_j) \\ -1 & : \text{sim}(u, y_i) < \text{sim}(u, y_j) \end{cases} \quad (24)$$

przy czym wartości punktowe są przyznawane dla obu obiektów odpowiednio w pozycji zwycięstwo lub porażka. W ten sposób powstaje tabela wyników stanowiąca ranking kandydatów. Globalna wartość punktowa dla obiektu y_i do wyznaczenia rankingu k 'tego komparatora jest wykonywana poprzez zsumowanie punktów za zwycięstwo z punktami za porażki (stanowiących wartość ujemną) i może być zapisane w postaci:

$$f_{vote}^k(y_i) = \sum_{j=1}^{|ref|} f_{result}(y_i, y_j) \quad (25)$$

W analogiczny sposób powstają tabele wyników dla poszczególnych komparatorów. W celu skonstruowania globalnego rankingu stanowiącego wynik wyborów sumowane są punkty poszczególnych kandydatów. Wynik punktowy pojedynczego kandydata może być obliczony na podstawie następującego wzoru:

$$score_{global}(y_i) = \sum_{j=1}^n f_{vote}^j(y_i) \quad (26)$$

Wartości wektora bliskości zwracanego przez agregator wyznaczone są w następujący sposób:

$$\left\langle \frac{n(|ref| - 1) + score_{global}(y_1)}{2n(|ref| - 1)}, \dots, \frac{n(|ref| - 1) + score_{global}(y_{|ref|})}{2n(|ref| - 1)} \right\rangle : |ref| > 1, n > 0 \quad (27)$$

gdzie n jest liczbą komparatorów oraz ref jest zbiorem referencyjnym o minimum dwóch elementach.

Metoda ta posiada złożoność obliczeniową rzędu $O(N^2)$. Nie wymaga dostarczenia żadnych dodatkowych parametrów do jej przeprowadzenia.

GŁOSOWANIE AKCEPTACYJNE - W tej metodzie komparatory w roli wyborców oddają głos na każdego z kandydatów w postaci obiektów referencyjnych. Głosowanie to zakłada oddanie przez każdego wyborcę tylu głosów ilu jest kandydatów. Głosy są dwójakiego rodzaju, w postaci akceptacji lub dezaprobaty. Metoda ta wymagała nieco szerszej adaptacji aby mogła być z powodzeniem użyta do agregacji wyników częściowych w sieciach komparatorów. Podobnie jak w pozostałych metodach głównym wyznacznikiem dotyczącym oddania głosu jest wartość podobieństwa zapisana w wektorze bliskości. Jednakże w tym przypadku każdy obiekt referencyjny należy zaakceptować lub odrzucić. Dlatego też została wprowadzona wartość graniczna podobieństwa, poniżej której głos był traktowany jako dezaprobaty, a w przeciwnym przypadku akceptacja. Wartość ta nazywana współczynnikiem akceptacji została oznaczona jako K_{af} . Wartość początkowa tego parametru to 1. Korzystając z wektora bliskości danego komparatora dokonujemy głosowania dla każdego kandydata korzystając z następującej funkcji:

$$f_{vote}^j(y_i) = \begin{cases} 1 & : sim(u, y_i) \geq K_{af} \\ 0 & : w.p.p. \end{cases} \quad (28)$$

gdzie 1 interpretowane jest jako akceptacja a 0 jako dezaprobaty. Głosowanie przeprowadzane jest dla każdego komparatora wykorzystując jego wektor bliskości. Następnie dokonywane jest zsumowanie głosów oddanych przez poszczególne komparatory. Całkowita liczba głosów oddanych na kandydata mierzona jest przy użyciu następującej formuły:

$$score(y_i) = \sum_{j=1}^n f_{vote}^j(y_i) \quad (29)$$

gdzie n oznacza liczbę komparatorów. Ze względu na fakt, że wartość początkowa współczynnika $K_{af} = 1$, może okazać się, iż żaden kandydat nie uzyskał większości. Przez *większość* rozumiane tu jest uzyskanie liczby głosów akceptacyjnych w liczbie większej od połowy możliwych do uzyskania. Dlatego też metoda ta została zaadaptowana w sposób iteracyjny. Pierwsza iteracja wykonywana jest dla wartości startowej współczynnika akceptacji. Jeśli żaden z kandydatów nie osiągnie większości to wartość współczynnika zostaje pomniejszona o 0.01. Operacja ta jest powtarzana aż do uzyskania większości przez jednego z kandydatów lub osiągnięcia przez współczynnik akceptacji najniższej możliwej wartości. Wartości końcowe wektora bliskości są ustalane wg. następującej zależ-

ności:

$$\mathbf{v} = \langle h(y_1), \dots, h(y_{|ref|}) \rangle \quad (30)$$

gdzie $h(y)$ jest postaci:

$$h(y_i) = \begin{cases} \frac{score(y_i)}{n} & : score(y_i) > \frac{n}{2} \\ 0 & : w.p.p. \end{cases} \quad (31)$$

Ogólna złożoność obliczeniowa tego algorytmu głosowania jest rzędu $O(N)$. W przypadku sieci komparatorów znajdują się jednak dodatkowe operacje które mają wyraźny wpływ na efektywność czasową algorytmu.

GŁOSOWANIE WAŻONE - Ten rodzaj głosowania operuje wagami decydującymi o sile danego głosu. Podobnie jak w poprzednich przypadkach głos oddawany jest przez komparatory stanowiące wyborców. Kandydatami są obiekty referencyjne, a oddanie głosu utożsamiane jest z wartością podobieństwa w wektorze bliskości. Wartości wektora wyjściowego mogą być wyznaczone w następujący sposób:

$$\mathbf{v}[i] = \frac{\sum_{j=1}^n w_j sim^j(u, y_i)}{\sum_{j=1}^n w_j} : \sum_{j=1}^n w_j > 0, w_j \geq 0 \quad (32)$$

gdzie n jest liczbą komparatorów. Metoda ta została wybrana, ze względu na niezwykłą prostotę oraz bardzo ważny czynnik jakim jest możliwość zastosowania uczenia maszynowego dla wag. O sile i skuteczności tej metody decydują wagi.

3.3 METODY STRUKTURALNE INSPIROWANE MERELOGIĄ PRZYBLIŻONĄ

W sieciach heterogenicznych w większości przypadków do poprawnego zagregowania wyników potrzebna jest wiedza o strukturze obiektu i jego relacjach. Rozpatrywane relacje dotyczą bycia częścią jednego obiektu w drugim, wyszukiwanie części wspólnych lub podobnych pomiędzy obiektami, jak również analizowanie części składowych. Takimi zagadnieniami zajmuje się dziedzina o nazwie mereologia. Jej rozszerzenie operujące stopniami spełnialności relacji nosi nazwę mereologii przybliżonej [12]. Funkcje inkluzji przybliżonej mogą stanowić dobre źródło funkcji podobieństwa jak również ich składania [13]. Taką rolę ma właśnie w tym kontekście agregator. Innymi słowy, poszukuje on odpowiedzi na pytanie, jak z

zebranych informacji o podobiekatach i ich podobieństwie, wywnioskować podobieństwo całego obiektu.

Dla sieci heterogenicznych operacja agregacji jest ściśle powiązana z operacją translacji. W praktyce jedna bez drugiej nie może się wykonać. Dzieje się tak dlatego, iż macierz translacji dana wzorem (7) zawiera w wierszach tzw. *reguły składania*. Reguły te definiują jak wyrazić obiekt warstwy dalszej obiektami warstwy poprzedniej danej sieci. W sieciach typu II kontekst warstwy związany jest z poziomem dekompozycji przetwarzanego obiektu. Dlatego też reguła składania do pewnego stopnia pozwala na wyliczenie wartości podobieństwa obiektów ogólniejszych, za pomocą podobieństw podobiektów. Reguły składania są postaci:

$$y_i = w_1 y_{i1} + w_2 y_{i2} + \dots + w_k y_{ik} \quad (33)$$

gdzie y_i jest obiektem na wyższym poziomie ogólności, y_{ij} są obiektami warstwy aktualnie przetwarzanej, gdzie $j = 1, \dots, k$, a $i = 1, \dots, l$ dla pewnych ustalonych k i l oznaczających odpowiednio liczbę podobiektów oraz liczbę obiektów głównych, a w_i stanowią wartości wiersza macierzy translacji (7). Dla relacji uogólnienia suma wag danego wiersza macierzy translacji musi spełniać następujący warunek:

$$\sum_{i \in I} w_i \leq 1 \quad (34)$$

gdzie I jest zbiorem indeksów w wierszach macierzy translacji o liczebności k . W przypadku gdy suma jest równa 1, przejście na wyższy poziom agregacji jest kompletne. Jeśli natomiast suma jest mniejsza od 1, to podobieństwa podobiektów nie determinują w pełni podobieństwa na wyższym poziomie ogólności. Wtedy wartość sumy wag określa stopień kompletności translacji. Dla sytuacji odwrotnej, gdy implementowana jest relacja dekompozycji, warunek z oczywistych względów nie jest spełniony.

Pełny proces agregacji strukturalnej jest dość skomplikowany. Zależy on od trzech głównych parametrów: struktury obiektu, komparatorów z których wyniki są agregowane oraz zbiorów referencyjnych używanych przez komparatory. Ogólne należy przyjąć, iż każdy zbiór referencyjny w sieci typu II jest podzbiorem ogólnego zbioru referencyjnego ref . Innymi słowy możemy mieć wiele zbiorów przetwarzanych oznaczonych jako $ref_1, ref_2, \dots, ref_m$, ale wszystkie one stanowią podzbiory zbioru głównego $ref_i \subset ref$.

Stan wejściowy agregacji strukturalnej to wektory bliskości pochodzące z komparatorów. Dodatkowym utrudnieniem jest fakt, że od każdego z

nich pojawia się tyle wektorów bliskości, na ile podobieństw zdekomponowany został obiekt wejściowy w danej warstwie. Łącznie pojawia się $d * m$ wektorów bliskości, gdzie d - oznacza liczbę zdekomponowanych podobieństw, a m liczbę komparatorów. Dodatkowo poszczególne wektory opisane różnymi podzbiarami referencyjnymi mogą mieć różne wymiary. Analizując je jednak względem komparatorów, można dla każdego z nich zbudować macierz postaci:

$$M_{sim} = \begin{bmatrix} \mathbf{v}_1 \\ \dots \\ \mathbf{v}_d \end{bmatrix} \quad (35)$$

gdzie \mathbf{v}_i to wektory bliskości poszczególnych zdekomponowanych podobieństw u_j , gdzie $j \in \{1, \dots, d\}$, postaci:

$$\mathbf{v}_j = \langle sim(u_j, y_{11}), \dots, sim(u_j, y_{1n}) \rangle \quad (36)$$

Proces agregacji dzieli się na dwa etapy. W pierwszym dokonywana jest fizyczna synteza, czyli ustalenie wyników na poziomie wektora bliskości. Proces ten można wykonać w przypadku, gdy więcej niż jeden komparator używa tego samego podzbioru referencyjnego, np. porównania dotyczą tych samych obiektów lecz różnych ich cech. W tym przypadku stosowane są metody agregacji wykorzystywane dla sieci typu I, np. bazujące na statystyce lub głosowaniu. W wyniku tej operacji, powstaje $d * k$ wektorów bliskości, gdzie k oznacza liczbę unikalnych zbiorów referencyjnych wykorzystywanych w warstwie, a d tak jak uprzednio, przy czym $k \leq m$.

Drugi etap wiąże się ściśle z translacją obiektów i składaniem podobieństw. Korzystając z reguł składania. W ten sposób następuje wyliczenie podobieństw wektora bliskości na wyższym poziomie. Do reguły składania, podstawiane są obiekty referencyjne wyższego poziomu, tworząc tyle równań, ile jest obiektów referencyjnych na wyższym poziomie ogólności. Wyliczając pojedynczą wartość wektora, podstawiamy za dany obiekt referencyjny regułę postaci podanej we wzorze (33) oraz korzystamy z wyników niższego poziomu zapisanych w macierzy zdefiniowanej wzorem (35). Ponieważ na niższym poziomie mamy podobieństwa dla poszczególnych podobieństw, do wzoru wstawiamy poszczególne z nim powiązane S -normą [1]. A zatem pojedynczy wzór wyliczający wartość wektora bliskości na wyższym poziomie będzie postaci:

$$sim(u, y_i) = S(w_1 sim(u_j, y_{i1}) + \dots + w_k sim(u_j, y_{ik}), \dots, w_1 sim(u_l, y_{i1}) + \dots + w_k sim(u_l, y_{ik})) \quad (37)$$

gdzie wewnątrz S-normy znajdują się wszystkie kombinacje podobieństw referencyjnych oraz podobieństw wejściowych złożonych zgodnie z odpowiednią regułą składania wynikającą z macierzy przejścia. W dalszej części sekcji zostanie przedstawiony ogólny przykład wyjaśniający działanie tej metody.

PRZYKŁAD - Załóżmy, iż mamy dany obiekt wejściowy u_1 dzielący się na dwa podobieństwa u_{11}, u_{12} oraz cztery obiekty referencyjne y_1, y_2, y_3, y_4 dzielące się analogicznie na podobieństwa postaci:

$y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}, y_{41}, y_{42}$. Załóżmy również, iż architektura sieci złożona jest z dwóch komparatorów. Istnieją również dwa podzbiory referencyjne $ref_1 = \{y_{11}, y_{21}, y_{31}, y_{41}\}$ oraz $ref_2 = \{y_{12}, y_{22}, y_{32}, y_{42}\}$ obsługiwane odpowiednio przez komparator C_1 i C_2 . Dana jest również macierz przejścia w postaci:

$$M_{ref}^{ref_1 \cup ref_2} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad (38)$$

Macierz ta definiuje reguły składania w postaci:

$$y_1 = \frac{1}{2}y_{11} + \frac{1}{2}y_{12} \quad (39)$$

$$y_2 = \frac{1}{2}y_{21} + \frac{1}{2}y_{22} \quad (40)$$

$$y_3 = \frac{1}{2}y_{31} + \frac{1}{2}y_{32} \quad (41)$$

$$y_4 = \frac{1}{2}y_{41} + \frac{1}{2}y_{42} \quad (42)$$

Warstwa pierwsza sieci zajmuje się przetwarzaniem podobieństw, a zatem jej zadanie sprowadza się do identyfikacji obiektów u_{11}, u_{12}, u_{13} względem podobieństw elementu zbioru referencyjnego ref . W wyniku przetwarzania wykonanego w tej warstwie powstały dwie macierze podobieństwa, oznaczone: $M_{sim}^{C_1}, M_{sim}^{C_2}$. Każda z nich zbudowana jest z dwóch czterowymiarowych wektorów bliskości. Każdy z nich przedstawia podobieństwo pomiędzy poszczególnymi podobieństwami wejściowymi do podobieństw referencyjnych.

$$M_{sim}^{C_1} = \begin{bmatrix} sim(u_1, y_{11}), sim(u_1, y_{21}), sim(u_1, y_{31}), sim(u_1, y_{41}) \\ sim(u_2, y_{11}), sim(u_2, y_{21}), sim(u_2, y_{31}), sim(u_2, y_{41}) \end{bmatrix} \quad (43)$$

oraz

$$M_{sim}^{C_2} = \begin{bmatrix} sim(u_1, y_{12}), sim(u_1, y_{22}), sim(u_1, y_{32}), sim(u_1, y_{42}) \\ sim(u_2, y_{12}), sim(u_2, y_{22}), sim(u_2, y_{32}), sim(u_2, y_{42}) \end{bmatrix} \quad (44)$$

Następnie rozpoczyna się opisywany drugi etap agregacji. Celem jest obliczenie wektora bliskości postaci:

$$v = \langle sim(u, y_1), sim(u, y_2), sim(u, y_3), sim(u, y_4) \rangle \quad (45)$$

Dla uproszczenia rachunków zostanie pokazane wyliczenie pierwszej współrzędnej. Pozostałe wylicza się analogicznie. Wyliczając $sim(u, y_1)$ bazujemy na regule składania postaci:

$$y_1 = \frac{1}{2}y_{11} + \frac{1}{2}y_{12} \quad (46)$$

Podstawiając tę regułę oraz bazując na pierwszej kolumnie macierzy (43) oraz (44). Otrzymujemy:

$$\begin{aligned} sim(u, y_1) = S & \left(\frac{1}{2}sim(u_1, y_{11}) + \frac{1}{2}sim(u_1, y_{12}), \right. \\ & \left. \frac{1}{2}sim(u_1, y_{21}) + \frac{1}{2}sim(u_2, y_{12}), \right. \\ & \left. \frac{1}{2}sim(u_2, y_{11}) + \frac{1}{2}sim(u_1, y_{12}), \right. \\ & \left. \frac{1}{2}sim(u_2, y_{11}) + \frac{1}{2}sim(u_{21}, y_{12}) \right) \end{aligned} \quad (47)$$

gdzie S jest S-normą.

4 PODSUMOWANIE

Problem agregacji wyników częściowych (lokalnych) oraz globalnych jest bardzo istotny z punktu widzenia efektywności działania sieci. Ten problem optymalizacyjny daje się rozwiązać wieloma metodami, lecz w poszczególnych przypadkach różne metody sprawdzają się najlepiej. Dlatego też dobrze skorzystać z metod potrafiących uczyć się na podstawie dostępnych danych. Dla sieci komparatorów istnieje kilka algorytmów uczących, które są stosowane do różnych celów. Jedne z nich optymalizują strukturę sieci, inne zajmują się doбором wag właśnie dla opisywanych w niniejszym artykule agregatorów.

Literatura

1. J. Kacprzyk. *Multistage Fuzzy Control: A Model-based Approach to Fuzzy Control and Decision Making*. John Wiley & Sons, Limited, 2012.
2. L. Sosnowski and D. Ślęzak, "Networks of compound object comparators," In Proceedings of the *FUZZ-IEEE*, India, Hyderabad 2013, pp. 1–8.
3. Ł. Sosnowski, "Framework of Compound Object Comparators," *Intelligent Decision Technologies*, 2015, to appear.
4. Ł. Sosnowski "Applications of comparators in data processing systems," *Technical Transactions, Automatic Control*, pp. 81–98, 2013.
5. Ł. Sosnowski and D. Ślęzak, "Comparators for Compound Object Identification," in *Proc. of RSFDGrC 2011*, ser. Lecture Notes in Computer Science, vol. 6743, 2011, pp. 342–349.
6. L. Sosnowski and D. Slezak, "Fuzzy set interpretation of comparator networks," in *Pattern Recognition and Machine Intelligence - 6th International Conference, PReMI 2015, Warsaw, Poland, June 30 - July 3, 2015, Proceedings*, 2015, pp. 345–353. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-19941-2_33.
7. Ł. Sosnowski, A. Pietruszka, and S. Łazowy, "Election algorithms applied to the global aggregation in networks of comparators," in *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, M. P. M. Ganzha, L. Maciaszek, Ed., vol. 2. IEEE, 2014, pp. pages 135–144. [Online]. Available: <http://dx.doi.org/10.15439/2014F494>.
8. D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, ser. Artificial Intelligence. Addison-Wesley, 1989. [Online]. Available: http://books.google.pl/books?id=3_RQAAAAMAAJ.
9. Ł. Sosnowski, "Identification with compound object comparators - technical aspects," in *Techniki informacyjne teoria i zastosowania*, J. Hołubiec, Ed. IBS PAN, 2011, vol. 1, pp. 168–179.
10. E. Elkind, J. Lang, and A. Saffidine, "Choosing collectively optimal sets of alternatives based on the condorcet criterion." in *IJCAI*, T. Walsh, Ed. IJCAI/AAAI, 2011, pp. 186–191. [Online]. Available: <http://dblp.uni-trier.de/db/conf/ijcai/ijcai2011.html#ElkindLS11>.
11. D. Nohlen, *Prawo wyborcze i system partyjny: o teorii systemów wyborczych*. Scholar, 2004. [Online]. Available: <http://books.google.pl/books?id=ckODAAAACAAJ>.
12. L. Polkowski, *Approximate Reasoning by Parts: An Introduction to Rough Mereology*, ser. Intelligent Systems Reference Library. Springer, 2011.
13. A. Gomolinska and M. Wolski, "Rough inclusion functions and similarity indices." in *CS&P*, ser. CEUR Workshop Proceedings, M. S. Szczuka, L. Czaja, and M. Kacprzak, Eds., vol. 1032. CEUR-WS.org, 2013, pp. 145–156. [Online]. Available: <http://dblp.uni-trier.de/db/conf/csp/csp2013.html#GomolinskaW13>.

AGGREGATION OF LOCAL RESULTS IN NETWORK OF COMPARATORS

Abstract. This article presents the brief key information of the theory of network of comparators, focusing their attention on the problem of aggregation in homogeneous and heterogeneous networks. It describes the basic mechanisms and methods used in modeling not limited to only one type of methods. It shows the different ways of handling the problem of aggregation from statistical methods, through election algorithms based method up to rough mereology inspired ones.

Keywords: networks of comparators, complex objects similarity, local results aggregation, identification and classification

IBS PAN *serie*

47 789

Bibl. podręczna

ISBN 83-894-7558-8