УДК 001.891:65.011.56
DOI https://doi.org/10.32838/2663-5941/2019.4-1/18

Повхан І.Ф.
ДВНЗ «Ужгородський національний університет»

Лавер В.О.
ДВНЗ «Ужгородський національний університет»

АЛГОРИТМИ ПОБУДОВИ ЛОГІЧНИХ ДЕРЕВ КЛАСИФІКАЦІЇ В ЗАДАЧАХ РОЗПІЗНАВАННЯ ОБРАЗІВ

Робота присвячена важливому питанню теорії розпізнавання – алгоритмам побудови логічних дерев класифікації. Простий, ефективний, економний метод побудови логічного дерева класифікації навчальної вибірки дає змогу забезпечити необхідну швидкодію, рівень складності схеми розпізнавання, що гарантує проведення простого та повного розпізнавання дискретних об'єктів. Результуюче правило класифікації (схема), яке побудоване довільним методом або алгоритмом розгалуженого вибору ознак (методом логічного дерева), має деревоподібну логічну структуру. Логічне дерево складається з вершин (ознак), які групуються по ярусах і отримані на певному кроці (етапі) побудови дерева розпізнавання. На відміну від наявних методів, головною особливістю деревоподібних систем розпізнавання є те, що важливість окремих ознак (групи ознак чи алгоритмів) визначається щодо функції, яка задає розбиття об'єктів на класи, причому числова величина важливості характеризує собою помилку розподілу об'єктів на класи. Отже, в методах та алгоритмах розпізнавання на основі логічних дерев класифікації необхідно до тих пір повторювати такий вибір вершин (ознак, аргументів, алгоритмів – у випадку алгоритмічного дерева), доки не буде отримано необхідний рівень якості розпізнавання дискретних об'єктів.

Основні наявні методи обробки навчаючих вибірок при побудові функції розпізнавання не дають змоги досягти потрібного рівня точності системи розпізнавання та регулювати їх складність у процесі конструювання цих систем. Цей недолік відсутній у методах побудови систем розпізнавання, які базуються на методах дерев класифікації. При цьому особливістю методу логічного дерева є можливість комплексного використання для розв'язання кожної конкретної задачі побудови схеми розпізнавання багатьох відомих алгоритмів (методів) розпізнавання. У роботі розглядаються деревоподібні схеми розпізнавання.

Ключові слова: *розпізнавання дискретних об'єктів, логічні дерева класифікації, функція розпізнавання, навчальна вибірка, алгоритм виправлення помилок.*

Постановка проблеми. Станом на сьогодні відомі різні алгоритми побудови логічних дерев класифікації (ЛДК) [1−3]. Проте всі вони, як правило, зводяться до побудови одного дерева класифікації за даними фіксованої навчальної вибірки (НВ). Зазначимо, що в літературі дуже мало алгоритмів побудови ЛДК для НВ великого об'єму. Зрозуміло, що це має під собою об'єктивні фактори, пов'язані з особливостями генерації таких складних структур, методиками роботи з ними та зберігання [4]. Навіть використовуючи інструментарій Java або C#, необхідно забезпечити реалізацію спеціальних структур даних для роботи з логічними деревами, а готові бібліотеки (LightGBM, XGBoost) хоча і близькі ідейно (схема логічного дерева), але не дають змоги реалізувати концепцію алгоритмічного дерева класифікації, яке складається з набору вершин – різнотипних автономних алгоритмів класифікації. Проте основним недоліком у питанні побудови ЛДК є відсутність алгоритмів та методів, котрі би дали змогу одноманітно описувати різні алгоритми розпізнавання образів у вигляді ЛДК.

Можливість представлення функції розпізнавання у вигляді логічного дерева має великі переваги порівняно з іншим представленням схем класифікації [5]. Зауважимо, що алгоритми генерації дерев класифікації за даними НВ доповнюють методологію підходу розгалуженого вибору ознак та дають змогу будувати прості та ефективні правила класифікації дискретних об'єктів [6].

У цій роботі зупинимось на описі алгоритму побудови ЛДК для НВ великого об'єму та покажемо шлях щодо можливості одноманітного опису одного фіксованого класу ЛДК.

Постановка завдання. Розглянемо принципові моменти задачі розпізнавання. Нехай задана

множина M об'єктів w та на ній є розбиття R на кінцеве число підмножин (класів, образів) Ω, $(i = 1,\dots m)$, $M = \cup\frac{m}{i=1}\copyright_i$. Припустимо, що розбиття M визначено неповністю. Задана тільки деяка інформація I про класи Ω_i. Об'єкти w задаються значеннями деяких ознак x_j, $j = 1,\dots,n$ (цей набір один і той самий для всіх об'єктів, тобто однакова розмірність об'єктів). Деяку скінчено – значна функція $f_R(w)$, яка задає розбиття R, задана на множині об'єктів M та дає на виході номер класу i, будемо називати функцією розпізнавання (ФР). Зауважимо, що кожен образ (клас) характеризується певною спільністю деяких властивостей його елементів (об'єктів), а елементи з різних образів не мають цієї спільності. Загальна задача розпізнавання полягає в тому, щоби для довільного об'єкта w встановити його належність певному класу (образу). Множини Ω_i також називаються компонентами розбиття множини *M*.

Сукупність значень ознак x_j, визначає опис (інформацію) $I(w)$ об'єкта w. Кожна з ознак може приймати значення з різних множин допустимих значень ознак. Опис об'єкта $I(w) = (x_1(\mathrm{w}),\dots,x_n(\mathrm{w}))$ будемо називати стандартним, якщо $x_j(w)$ приймає значення лише із множини допустимих значень.

Задача розпізнавання із стандартною інформацією полягає в тому, щоб для фіксованого об'єкта w та набору класів $\Omega_1,\dots,\Omega_m$ за допомогою навчальної інформації $I(\Omega_1,\dots,\Omega_m)$ та опису $I(w)$ розрахувати значення деяких предикатів $P_i(w),(w \in \Omega_i; i = 1,\dots,m)$.

Одним із можливих варіантів початкового завдання навчальної інформації є табличне представлення навчальної вибірки $T_{N,M}$ (наборів об'єктів відомої класифікації). Очевидно, що тут об'єкти $w_1,\supset,w_{r_1}$ належать класу $|_1$, а об'єкти $w_{r_{i-1}},\dots,w_{r_i}$ належать класу $|_i$ та об'єкти $w_{r_{m-1}+1},\dots,w_{r_m}$ – класу $|_m$.

Нехай є розбиття R та деяка система розпізнавання Q. Як система Q може бути людина або програмно-апаратна система (система операцій або логічних елементів). Задача розпізнавання образів буде зводитися до навчання системи Q обчислювати функцію $f_R(x)$. Тобто система має реагувати в разі подачі на вхід деякого сигналу w сигналом $f_R(x)$ (фактичним номером класу належності). Основною інформацією при навчанні системи Q є значення функції $f_R(x)$ в деяких точках n – мірного простору (розмірністю в кількість ознак об'єктів множини M). Останнє означає, що під час навчання системи Q їй подаються пари сигналів $((x_i, f_R(x_i))$. На основі цієї інформації (апріорної інформації) система Q будує схему обчислення $f_R(x)$.

Резюмуючи все вищесказане, приходимо до таких положень:

1) задача розпізнавання образів зводиться до того, щоб навчити систему Q обраховувати деяку функцію $f_R(x)$, яка визначена на множині M та приймає скінчену кількість значень. Функція $f_R(x)$ задає однозначне розбиття R. Причому дві функції $f_R(x)$ та $h_R(x)$ будуть вважатись однаковими, якщо вони представляють одне і те саме розбиття;

2) базовий момент у розпізнаванні образів – навчання (фактична обробка великих масивів інформації). У процесі навчання система Q приймає послідовність навчальних пар $(x_1, f_R(x_1)),((x_2, f_R(x_2)),\dots.$ та на основі цієї інформації будує схему обчислення $f_R(x)$ або її наближення;

3) на етапі навчання системи виникають такі питання, як економія пам'яті системи Q, швидкодія навчання системи Q, побудова такої схеми для обчислення $f_R(x)$, щоб вона була за деякими важливими параметрами оптимальною (об'єм пам'яті, швидкодія, надійність).

Питання синтезу логічних дерев класифікації. У цій роботі ставиться задача дослідження та розробки таких методів розпізнавання, які би давали змогу у процесі навчання побудувати за можливості просту деревоподібну схему розпізнавання (схему у вигляді ЛДК), яка забезпечує необхідну ефективність та складність системи Q. Основні переваги алгоритмів представлення схем розпізнавання у вигляді ЛДК полягають у такому:

1) метод побудови ЛДК не потребує одночасного запам'ятовування всіх даних НВ: можливе поетапне введення даних (по векторах) і навіть окремих значень ознак об'єктів. Після введення вектор (об'єкт, який подається на вхід системі) використовується для побудови ЛДК і в подальшому в пам'яті комп'ютера не зберігається. Таке введення інформації та специфіка алгоритму значно економлять пам'ять машини і дають змогу розв'язувати задачі розпізнавання з НВ великої потужності (об'єму);

2) алгоритм забезпечує безпомилкову класифікацію НВ у разі початкової відмінності між об'єктами різних класів;

3) робота алгоритму не залежить від кількості образів (класів) у НВ. Зокрема, алгоритм можна застосовувати за наявності в НВ тільки одного образу. Відсутність такої можливості в деяких випадках представляє собою негативний фактор;

4) кількість помилок, що допускаються ЛДК на фіксованій НВ, не збільшується, якщо зростає об'єм НВ. Ця вимога дозволяє визначити збіжність цього алгоритму;

5) алгоритм включає в себе просту схему донавчання та усунення знайдених помилок розпізнавання;

6) алгоритм дає змогу сформулювати підхід, що дозволяє одноманітно описувати доволі широкий клас алгоритмів у вигляді ЛДР.

Опишемо цей алгоритм. Нехай НВ задана у вигляді матриці:

$$\begin{matrix} x_{1,1}, x_{1,2}, \ldots\ldots\ldots\ldots\ldots\ldots, x_{1,n} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ x_{m,1}, x_{m,2}, \ldots\ldots\ldots\ldots\ldots\ldots, x_{m,n} \\ x_{m+1,1}, x_{m+1,2}, \ldots\ldots\ldots\ldots, x_{m+1,n} \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ x_{k,1}, x_{k,2}, \ldots\ldots\ldots\ldots\ldots\ldots, x_{k,n} \end{matrix}, \qquad (1)$$

$\{x_{i,j}\}$ – значення j-вої ознаки i-го об'єкта НВ, $(i = 1,\ldots k; j = 1,\ldots,n)$. Для спрощення викладання вважаємо, що $x_{i,j} \in \{0,1\}$, m – рядків НВ характеризує клас $|_1$, а інші – $|_2$.

Побудуємо за НВ вигляду (1) ЛДК. У першу вершину логічного дерева ставимо ознаку x_1 і від неї будуємо повний шлях, який відповідає набору $x_{1,1}, x_{1,2}, \supset, x_{1,n}$, тобто з вершини з ознакою x_j виходить стрілка, що входить у вершину з ознакою x_{j+1} (номер стрілки залежить від значення, яке приймає $x_{1,j}$, в кінцевій вершині ставимо значення f_R (функції розпізнавання).

Для наступного об'єкта $w_2 = (x_{2,1},\ldots,x_{2,n})$ при $w_1 \uparrow w_2$ ЛДК змінюється таким чином: при $x_{2,j} \uparrow x_{1,j}$ з вершини з ознакою x_j проводимо другу стрілку, яка відповідає значенню $x_{2,j}$ (для зручності стрілки з номером 0 розташовуємо з лівої сторони, а 1 – з правої), та наприкінці стрілки ставимо вершину з ознакою x_{j+1}. Наступним кроком із цієї вершини проводимо стрілку, відповідну значенню ознаки x_{j+1} і так далі. Зрозуміло, що в кінцеву вершину ставимо значення $f_R(w_2)$ – (рис. 1).

Аналогічним чином, добудовуємо ЛДК для всіх інших наборів (1). Якщо на одному ЛДК зустрічаються два або кілька однакових наборів із різних класів (різних значень функції розпізнавання), то кількість значень функції f_R (відповідних цим наборам) фіксуємо в кінцевій вершині і остаточно записуємо те значення функції f_R, для якої кількість наборів буде максимальною.

Якщо кількість значень функції f_R однакове в деякій кінцевій вершині, то перевагу віддаємо образу, потужність якого менше. Після цього ЛДК за НВ будемо вважати закінченим.

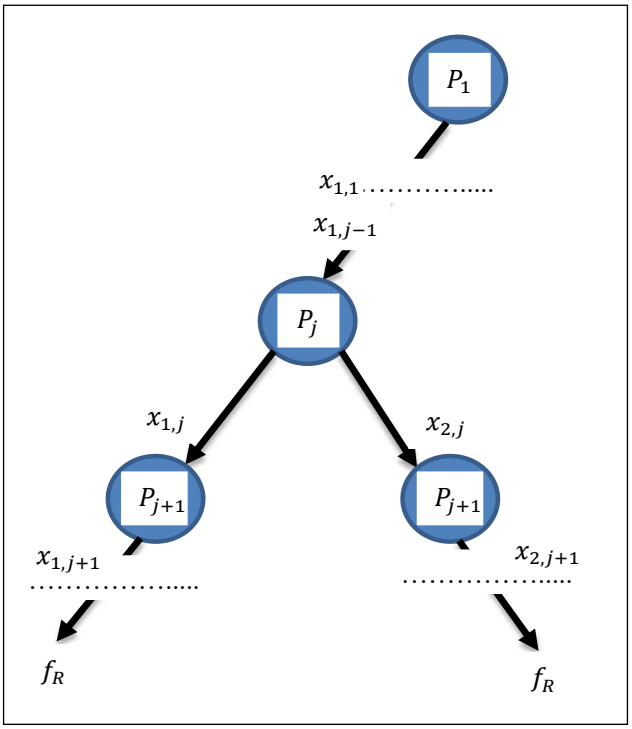

Рис. 1 Загальна схема побудови ЛДК

Задача розпізнавання образів полягає в тому, щоби побудоване ЛДК класифікувало всі набори ТВ, які не входять у НВ (зауважимо, що набори об'єктів НВ, логічне дерево розпізнає без помилок завдяки їх фіксації у структурі самого дерева).

Після побудови ЛДК за цим алгоритмом будемо мати таку структуру, де не з усіх вершин виходять дві стрілки. Це дає змогу провести просту мінімізацію: якщо з вершини виходить одна стрілка, то її разом із стрілкою видаляємо зі структури ЛДК. Мінімізоване таким чином ЛДК позначимо через ЛДК*.

Властивості логічних дерев класифікації. Зазначимо такі загальні властивості ЛДК:

1) складність (під складністю дерева розуміємо загальну кількість вершин у ЛДК [7]) ЛДК* – менше ніж складність початкового ЛДК (під час мінімізації відбувається відсів найменш важливої інформації);

2) ЛДК*, як і початкове ЛДК, безпомилково класифікує об'єкти НВ та забезпечує екстраполяцію наборів ТВ, котрі не зустрічались у масиві початкових даних НВ.

Кожному набору Δ з ТВ відповідає своя $f_R(\tau_\Delta), \Delta \in (0,1,\ldots,n)$, кінцевої вершини на ЛДК, що визначає приналежність цього набору до того чи іншого класу.

Зрозуміло, що різні алгоритми дають різні результати на цій ТВ – можлива ситуація, коли фіксований об'єкт буде зарахований до різних

образів. Причина цього полягає в тому, що задача розпізнавання не має єдиного розв'язку і кожен з алгоритмів дає свій розв'язок.

Розглянемо якість розв'язку задачі РО на простому прикладі. Нехай маємо справу з НВ, яка представлена в (табл. 1) об'ємом 5 наборів. Значення $f_R(P_1, P_2, P_3)$ вказує на приналежність наборів НВ до того чи іншого класу. Потрібно знайти функцію $f_R^*(P_1, P_2, P_3)$, яка забезпечить апроксимацію $f_R(P_1, P_2, P_3)$. Очевидно, що є 8 таких різних функцій. Якщо $P_j \in \{0,1\}, j = 1,\dots n$, n – кількість ознак, m – кількість наборів НВ, то кількість різних розв'язків, які не входять до наборів НВ, дорівнює $L = 2^{2^n - m}$. Якщо $P_j \in \{0,1,\dots,k-1\}$, то $L = k^{k^n - m}$.

Таблиця 1

Таблична форма не повністю визначеної $f(P_1, P_2, P_3)$

Номер набору	P_1	P_2	P_3	$f_R(P_1, P_2, P_3)$
1	0	0	0	0
2	0	0	1	0
3	0	1	0	1
4	0	1	1	1
5	1	0	0	1
6	1	0	1	?
7	1	1	0	?
8	1	1	1	?

Зазначимо, що якщо задано m різних наборів НВ, то кількість помилок не може бути більше ніж $2^n - (m-1)$. Таким чином, лише за даними НВ не можна зробити висновок, який алгоритм класифікації дає кращий результат для фіксованої задачі розпізнавання. У зв'язку з вищесказаним не можна класифікувати алгоритми РО лише за вузькою специфікою області застосування. З огляду на цей факт кожен метод розпізнавання має включати механізм (алгоритм), котрий усуває знайдені помилки, та забезпечувати необхідну гнучкість щодо області застосування.

Схема донавчання та виправлення помилок класифікації. Запропонуємо схему простого та ефективного алгоритму донавчання та виправлення помилок (алгоритм ДВП) у ЛДК. Нехай маємо деяке ЛДК*, яке побудоване на основі фіксованої НВ (розмірність об'єктів якої – n ознакам). Нехай на деякому наборі – $\tau = (x_{i,1}, \dots, x_{i,n})$ відбувається помилка. Тоді результуюче ЛДК будемо міняти таким чином. У кінцеву вершину, що знаходиться на шляху $(x_{i,1}, \dots, x_{i,n})$, ставимо одну з цих ознак та добудовуємо ЛДК так, щоби гілка цих ознак відповідала шляху, в кінцеву вершину котрого ставимо номер класу, відповідний набору $(x_{i,1}, \dots, x_{i,n})$.

Для вершин ЛДК з однією стрілкою на шляху $(x_{i,1}, \dots, x_{i,n})$ додаємо стрілки, яких бракує, та наприкінці їх записуємо те значення $f_R(\tau_{\cdot\cdot})$, котре було в кінцевій вершині до виявлення помилки. Зрозуміло, що в разі такого виправлення помилок кількість вершин результуючого ЛДК дещо збільшується, але всі набори НВ розпізнаються безпомилково (на наборі $(x_{i,1}, \dots, x_{i,n})$ помилок не відбувається).

Розглянемо простий приклад застосування описаного алгоритму. Нехай НВ задана у вигляді наборів об'єктів $w(P_1, \dots, P_n)$ загальною кількістю $m = 5$ та розмірністю ознак $n = 3$ (Табл. 2), а відповідно ТВ (Табл. 3). Побудуємо за даними НВ результуюче ЛДК та перевіримо його роботу (включаючи алгоритм донавчання та виправлення помилок) на ТВ.

Усі етапи побудови результуючого ЛДК за даними НВ показані на рис. 2. Зауважимо, що остаточне мінімізоване ЛДК буде отримано на останньому етапі (етап 7).

Таблиця 2

Таблична форма НВ об'ємом $m = 5$

Номер набору	P_1	P_2	P_3	$f_R(P_1, P_2, P_3)$
1	0	0	0	1
2	0	0	1	0
3	0	1	1	0
4	1	0	1	1
5	1	1	0	0

Перевіримо роботу ЛДК на ТВ:

1) 1-й і 2-й набори ТВ ЛДК розпізнає правильно, бо вони входили (збігалися) в НВ;

2) 3-й набір належить до класу 0 (насправді він із класу 1). Після застосування алгоритму донавчання та виправлення помилок ЛДК матиме вигляд, показаний на рис. 2 (етап 7). Таким чином, відбулося донавчання на n – вому наборі ТВ. Інші набори ТВ ЛДК розпізнає без помилок.

Таблиця 3

Таблична форма ТВ об'ємом $m = 8$

Номер набору	P_1	P_2	P_3	$f_R(P_1, P_2, P_3)$
1	0	0	1	0
2	0	1	1	0
3	0	1	0	1
4	0	1	0	1
5	1	0	0	1
6	1	1	1	0
7	1	0	0	1
8	1	1	1	0

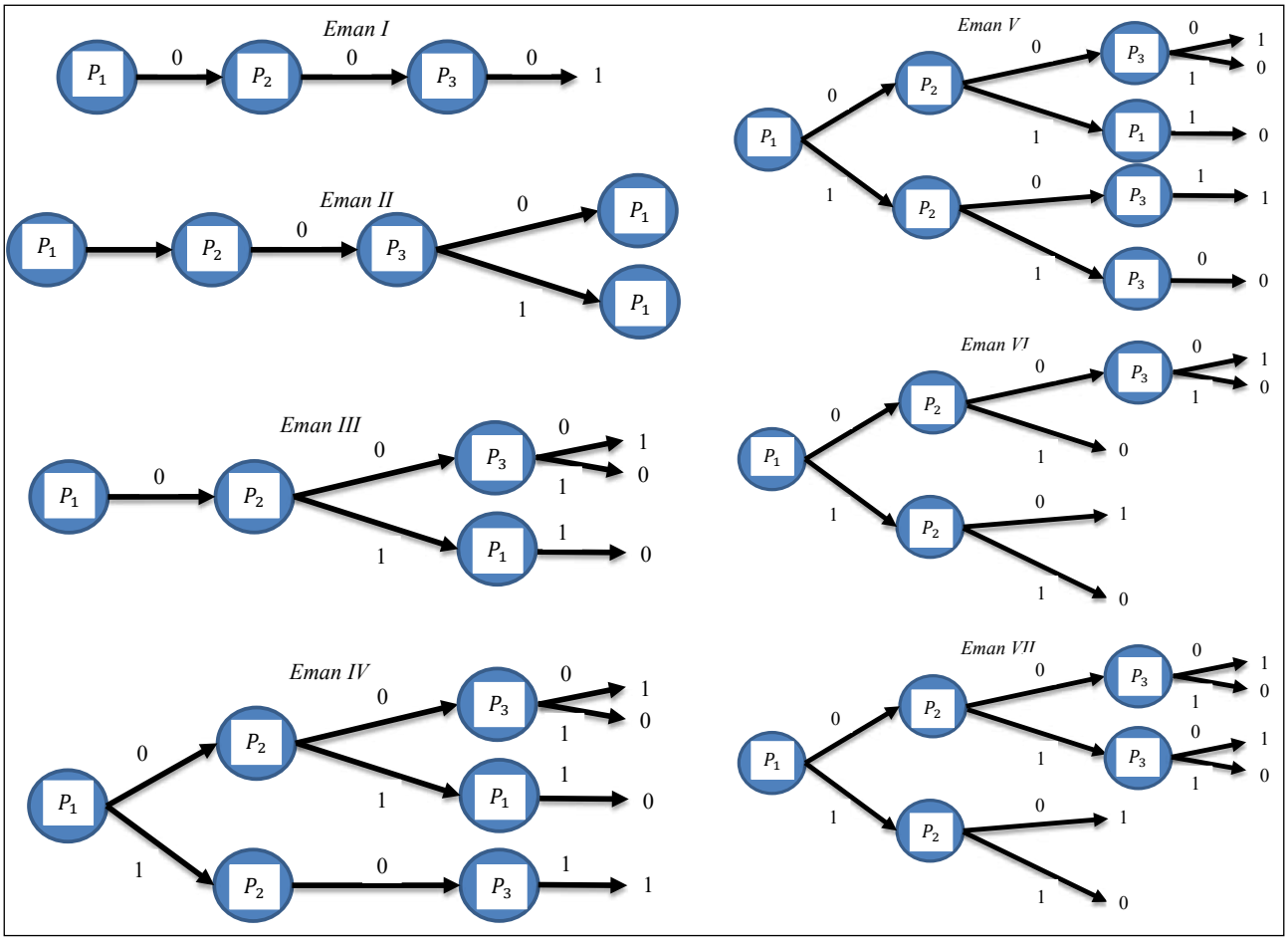

Рис. 2. Всі етапи побудови результуючого ЛДК за даними НВ

Таблиця 4

Порівняльна таблиця алгоритмів

АЛГОРИТМ		ЗАДАЧА		
		№ 1	№ 2	№ 3
№ 1	Лінійна розв'язуюча функція «Кульбаба»	64	64	55
№ 2	Нелінійна розв'язуюча функція «Едельвейс»	76	68	55
№ 3	Алгоритм «Кора-3» (алгоритм М. Вайнцвайга та М. Бонгарда)	32	68	65
№ 4	Алгоритм «Тест-2»	72	65	70
№ 5	Алгоритм «Ентропія–1»	73	64	72
№ 6	Алгоритм «Ентропія–2»	56	56	25
№ 7	Алгоритм «Ентропія–3»	56	76	15
№ 8	Алгоритм «Голотип»	44	52	50
№ 9	Алгоритм на основі потенціальних функцій (для тесту взята реалізація ПК «Оріон ІІІ»)	44	60	50
№ 10	Алгоритм «Геолог-1»	72	36	55
№ 11	Алгоритм «Південь»	76	68	55
№ 12	Алгоритм РВО з покроковою оцінкою важливості ознак (модифікація алгоритму Ю. Василенка)	56	68	70
№ 13	Алгоритм побудови повного ЛДК	74	70	65
№ 14	Запропонований вище алгоритм ДВП у ЛДК	74	80	72

Очевидно, що за допомогою цього алгоритму при розв'язуванні практичних задач розпізнавання не завжди досягаються прийнятні результати. Цей алгоритм можна звести до вигляду, що дає змогу описувати доволі широкий клас методів розпізнавання у вигляді ЛДК. Зауважимо, що ознаки у

процесі побудови ЛДК вибиралися послідовно та впорядковано – один за одним. Якщо відмовитися від цієї умови та вибрати ознаки в процесі побудови ЛДК довільним чином (наприклад, за допомогою програмного генератора псевдовипадкових чисел), то в результаті щоразу у процесі застосування цього алгоритму буде виходити інше за структурою ЛДК. Зауважимо, що вибір того чи іншого ЛДК у процесі розв'язування практичних задач може бути різноманітний.

Висновки. Результати застосування описаного вище алгоритму та порівняння його ефективності щодо інших алгоритмів на реальних задачах за масивами даних геології, геохімії та геофізики відображені (для простоти візьмемо добре відомі приклади з [8]) в табл. 4. Для кожного з 14 алгоритмів (на кожній із трьох задач) наведена кількість правильних відповідей (в %), одержаних на відповідних НВ.

Зауважимо, що для порівняльної таблиці частина алгоритмічних реалізацій була взята з програмного комплексу «Оріон ІІІ» [2] – наприклад, алгоритм потенціальних функцій та алгоритм розгалуженого вибору ознак із покроковою оцінкою їх важливості [3]. Зважаючи на отримані дані щодо проценту успішної класифікації, можна зробити висновок про доволі високу ефективність у цьому плані алгоритму ДВП логічного дерева (не враховується інформаційна ємність логічного дерева та швидкодія результуючої схеми класифікації).

Список літератури:

1. Повхан І.Ф. Метод розгалуженого вибору ознак в математичному конструюванні багаторівневих систем розпізнавання образів. Науково-технічний журнал «Штучний Інтелект». 2003. № 7. С. 246−249.
2. Повхан І.Ф., Василенко Ю.А., Василенко Е.Ю. Концептуальна основа систем розпізнавання образів на основі метода розгалуженого вибору ознак. Науково технічний журнал «European Journal of Enterprise Technologies». 2004. № 7[1]. С. 13−15.
3. Повхан І.Ф., Василенко Ю.А. Групова та індивідуальна оцінка важливості бульових аргументів. Вісник національного технічного університету «ХПІ». 2011. № 53. С. 57−64.
4. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning. 2008. P.768
5. Повхан І.Ф. Проблема функціональної оцінки навчальної вибірки в задачах розпізнавання дискретних об'єктів. Вчені записки Таврійського національного університету. 2018. Серия: Технічні науки. 2018. Том 29(68). № 6. С. 217−222.
6. Василенко Ю.А. Логики-алгебраический подход к обработке обучающей выборке. Науковий вісник УжДІІЕП. 1997. № 1. С. 4−10.
7. Povhan I. Designing of recognition system of discrete objects. IEEE First International Conference on Data Stream Mining & Processing (DSMP), Lviv, 2016, Ukraine, P. 226−231.
8. Воронин Ю.А. Геология и математика. Новосибирск, 1970. С. 223.

Povhan I.F., Laver V.O. THE ALGORITHMS FOR CONSTRUCTING A LOGICAL TREE OF CLASSIFICATION IN PATTERN RECOGNITION PROBLEMS

The work is devoted to the important issue of the theory of recognition – algorithms for constructing a logical tree of classification. A simple, effective, economical method of constructing a logical classification tree of the training sample allows you to provide the necessary speed, the level of complexity of the recognition scheme, which guarantees a simple and complete recognition of discrete objects. The resulting classification rule (scheme), which is constructed by an arbitrary method or algorithm of branched feature selection (logical tree method), has a tree-like logical structure. The logical tree consists of vertices (features), which are grouped in tiers and obtained at a certain step (stage) of building the recognition tree. In contrast to the existing methods, the main feature of tree recognition systems is that the importance of individual features (groups of features or algorithms) is determined relative to the function that defines the division of objects into classes, and the numerical value of importance characterizes the error of dividing objects into classes. Therefore, in the methods and algorithms of recognition based on logical classification trees, it is necessary to repeat such a choice of vertices (features, arguments, algorithms – in the case of an algorithmic tree) until the required level of quality of recognition of discrete objects is obtained.

The main existing methods of processing training samples in the construction of recognition functions do not allow to achieve the desired level of accuracy of the recognition system and adjust their complexity in the process of designing these systems. This drawback is absent in the methods of construction of recognition systems based on the methods of classification trees. At the same time, a feature of the logical tree method is the possibility of complex use for solving each specific problem of constructing a recognition scheme for many well-known algorithms (methods) of recognition. The paper considers tree-like recognition schemes.

Key words: *recognition of discrete objects, logical classification trees, detection, training sample, error correction algorithm.*