Національна академія наук України
Інститут кібернетики імені В. М. Глушкова

На правах рукопису

ЦВЄТКОВ Олександр Михайлович

ДОСЛІДЖЕННЯ ІНДУКТИВНИХ МЕТОДІВ ВИВОДУ ЗНАНЬ В ЕКСПЕРТНИХ СИСТЕМАХ

05.13.16 — застосування обчислювальної техніки, математичного моделювання та математичних методів в наукових дослідженнях

Автореферат дисертації на здобуття наукового ступеня кандидата фізико-математичних наук

Київ 1994

Робота виконана в Науково-учбовому центрі прикладної інформатики НАН України.

Науковий керівник: доктор фізико-математичних наук
ГУПАЛ Анатолій Михайлович

Офіційні опоненти: член-кореспондент НАН України, доктор фізико-математичних наук
МАР'ЯНОВИЧ Тадеуш Павлович,
кандидат фізико-математичних наук ГАЛАГАН Микола Іванович

Провідна установа: Київський університет імені Т. Г. Шевченка.

Захист відбудеться «23» вересня 1994 р. о 14 год. на засіданні спеціалізованої вченої ради Д 016.45.01 при Інституті кібернетики імені В. М. Глушкова НАН України за адресою:

252650 Київ МСД 22, проспект Академіка Глушкова, 40.

З дисертацією можна ознайомитися у бібліотеці Інституту

Автореферат розісланий «1» августа 1994 р.

Учений секретар
спеціалізованої вченої ради СИНЯВСЬКИЙ В. Ф.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність проблеми. В даний час при дослідженнях зі штучного інтелекту сформувався потужний самостійний напрямок - розробка і використання експертних систем. Його мета полягає у розробці систем, спроможних вирішувати важкі для людини-експерта задачі за допомогою обчислювальної техніки. У більшості випадків експерні системи розв'язують задачі, що їх важко формалізувати, чи задачі, які не мають алгоритмічного розв'язання.

Клас важкоформалізованих задач володіє однією чи декількома із наступних характеристик: задача не може бути виражена у числовій формі, неможливо точно визначити цільову функцію, не існує алгоритмічного розв'язку задачі, алгоритмічний розв'язок існує, але недосяжний внаслідок обмеженості ресурсів.

Отримання знань є основним етапом у розробці будь-якої експертної системи. Процес отримання знань для більшості діючих експертних систем можна розподілити на три етапи: отримання знань від експерта, організація знань, що забезпечує ефективну роботу системи, та представлення знань. Очевидно, що найбільш вузьким місцем у цьому процесі є добування знань від експертів. Основна складність полягає у тому, що експерт досить часто не в змозі коректно сформулювати принципи, котрими він користувався при розв'язанні задачі. Тому для сучасного рівня розвитку експертних систем особливо актуальна проблема побудови методів автоматичної генерації знань. У рамках систем штучного інтелекту цей напрямок отримав назву - машинне навчання (machine learning), інакше - індуктивний вивід. Основна мета цих методів - отримання знань про предметну область (різного роду залежностей, причинно-наслідкових зв'язків) на основі аналізу прикладів із цієї предметної області.

Мета роботи :

- розробка і дослідження методів індуктивного виводу з точки зору теорії складності та їх використання у практичних застосуваннях;

- розробка методів представлення даних для задач машинного навчання.

Наукова новизна. З математичної точки зору, найбільш розробленими для розв'язання задач індуктивного виводу є методи прикладної статистики - факторний і дискримінантний аналіз, методи перевірки статистичних гіпотез. Незважаючи на глибоку теоретичну і практичну розробку, застосування чисто статистичних методів в експертних системах далеко не завжди приносить очікувані результати. Це пов'язано з рядом причин: коректне застосування статистичних методів вимагає виконання певних умов на початкових даних (нормальність розподілів, рівність коваріаційних матриць та ін.), котрі часто порушуються на реальних наборах даних, крім того, на великих наборах прикладів статистичні методи вимагають значних обчислювальних ресурсів.

У запропонованій роботі розроблені і обгрунтовані методи індуктивного виводу, які можуть працювати з довільними наборами початкових даних. Принципово новою можливістю запропонованих методів є спроможність породження нових атрибутів, тобто зміни початкового простору описів задачі. Дослідження в області теорії символьних об'єктів привели до нової логічної трактовки поняття "гарна гіпотеза".

Загальна методика досліджень. Математичним апаратом, який використовувався у роботі, є математична логіка, теорія ймовірностей і теорія оптимізації.

Практична цінність результатів досліджень. Результати досліджень можуть бути безпосередньо використані для розв'язання задач із галузей матеріалознавства (прогнозування властивостей нових матеріалів, визначення оптимальних параметрів технологічних процесів), медицини (постановка діагнозу за спостережуваними симптомами), селекції, захисту рослин і багатьох інших. Ряд практичних прикладів описано у третій главі дисертації, додається відповідний акт упровадження.

Апробація роботи. Результати дисертаційної роботи викладалися і обговорювалися на семінарах і конференціях:

1. Міжнародна конференція "Інтелектуалізація систем баз даних" (Калінінград, 1992р.).

2. Міжнародний науково-технічний семінар "Теоретичні і прикладні проблеми моделювання предметних областей у системах баз даних і знань" (Туапсе, 1992, 1993рр.).

3. Наукові семінари Науково-учбового центру прикладної інформатики АН України у 1991-1994р.

Публікації. Основні результати дисертації опубліковані у роботах [1-9].

Структура і обсяг роботи. Дисертаційна робота складається зі вступної частини, трьох глав, заключної частини, двох додатків та списку літератури (77 найменувань). Обсяг роботи складає 180 сторінок машинописного тексту.

ОСНОВНИЙ ЗМІСТ РОБОТИ

Дисертаційна робота присвячена розробці та реалізації методів індуктивного виводу знань. При цьому автором виконана наступна робота:

- систематизована і доопрацьована теорія символьних об'єктів, що є домінуючою в області представлення знань у задачах машинного навчання, на основі розвитку теорії повних символьних об'єктів розроблені відповідні алгоритми;

- запропоновані і реалізовані алгоритми індуктивного виводу, які спроможні модифікувати дерева рішень по мірі надходження нових навчаючих прикладів, виконані оцінки їх складності;

- представлені методи "підрізання" дерев рішень, що підвищують якість класифікації нових прикладів, проведено порівняння з існуючими алгоритмами, що показало перевагу розроблених методів в якості прогнозу на 5-10 відсотків;

- розроблені алгоритми "конструктивної" індукції на деревах рішень, доведені теореми, що підтверджують доцільність застосування даних алгоритмів;

- запропоновано ряд методів, що дозволяють використовувати експертні знання у процесі індукції;

- розроблені алгоритми побудови так званих листів рішень для багатозначного випадку, доведена теорема про

поліноміальну складність навчання класу k-ЛР(n), який містить листи рішень, кожний терм яких складається не більше ніж з k термів на множині n змінних;

- для комітетних алгоритмів запропоновано нове перетворення, яке дозволяє будувати близькі до мінімальних комітети лінійних нерівностей;

- програмно реалізована оболонка системи індуктивного виводу, яка дозволяє за допомогою розроблених методів розв'язувати широкий клас задач. Її опис надається у Додатку 2.

Запропоновані методи перевірені на практичних задачах, надається відповідний акт упровадження.

У вступній частині дисертації обґрунтовується актуальність теми, наведено ряд основних робіт з цієї тематики і коротко викладені основні результати.

У главі 1 міститься огляд сучасного стану теоретичних розробок в області індуктивного виводу функцій, наводяться основні означення і теореми.

У главі 2 проведена розробка методів представлення знань та алгоритмів індуктивного виводу. Представлено порівняльний аналіз запропонованих алгоритмів на прикладі задачі прогнозування по наданому хімічному складу зварювально-технологічних властивостей покритих електродів. Ця та ряд інших задач докладно розглядаються у третій главі роботи.

У розділі 2.1 розвинена теорія символьних об'єктів, на основі якої запропоновано новий алгоритм індуктивного виводу.

Символьний об'єкт визначається як кон'юнкція значень, що приймаються змінними. Змінна y - це відображення $y:\Omega \to O$, де Ω - множина об'єктів і O - множина спостережуваних значень y. Зауважимо, що кожний рядок у масиві даних, що описує об'єкт, може бути виражений кон'юнкцією логічних тверджень, котру ми назвемо подією. Таким чином, множина Ω розглядається як множина елементарних об'єктів. Нехай $y_1,\dots,y_p$ - змінні, які визначені на множині Ω і набувають значення із $O_1,\dots,O_p$. Множина Ω може розглядатися як

підмножина $\Omega' = O_1 \times \ldots \times O_p$ множини усіх можливих елементарних об'єктів. Елементарна подія визначається як $e_i = [y_i = V_i]$, де $V_i \subset O_i$, тобто як предикат вигляду "змінна y набуває свого значення із V_i". Це означає, що $[y_i = V_i]$ - логічне об'єднання подій $[y_i = (v_j)]$ для усіх v_j у V_i. Розширення e_i визначається як $|e_i|_\Omega = \{\omega \in \Omega: y_i(\omega) \in V_i\}$.

Означення 7. Об'єкт-твердження - це кон'юнкція подій типу $[y_i = V_i]$, тобто $a = [y_1 = V_1] \wedge \ldots \wedge [y_q = V_q]$.

Означення 8. Множина об'єктів із Ω, що задовольняє об'єкт-твердження a позначається $|a|_\Omega$ і утворює розширення a у Ω: $|a|_\Omega = \{\omega \in \Omega: y_i(\omega) \in V_i$ для $i=1,\ldots,q\}$.

Елементарний об'єкт утворюється із елементів Ω відображенням $\gamma: \Omega \to S$: $\gamma(\omega) = [y_1 = y_1(\omega)] \wedge \ldots \wedge [y_p = y_p(\omega)]$.

Для спрощення позначень припустимо, що визначено множину символьних об'єктів S на множині Ω, яка описується змінними $y_i: \Omega \to O_i$. Покладемо, що S - множина об'єктів-тверджень. Елементарний символьний об'єкт, побудований на основі елементів Ω, також належить S. Розширення символьного об'єкта $s \in S$ у множині Ω позначається s' і складається із елементарних об'єктів $\gamma(\omega) \in S$ таких, що $\omega \in \Omega$ належить розширенню s у Ω. Іншими словами,

$$s' = \{\gamma(\omega) \in \Omega : \omega \in |s|_\Omega\}.$$

Означення 9. (Означення порядку). $\forall s_1, s_2 \in S$ $s_1 \leqslant s_2$ тоді і тільки тоді, коли $s_1' \subseteq s_2'$.

Означення 10. (Означення слідства і узагальнення). $\forall s_1, s_2 \in S$ говоримо, що s_1 походить із s_2 і що s_2 більш загальний, ніж s_1, тоді і тільки тоді, коли $s_1 \leqslant s_2$.

Означення 11. (Означення об'єднання і перетину символьних об'єктів). Визначимо $s_1 \cup s_2$ (відповідно $s_1 \cap s_2$) як кон'юнкцію усіх символьних об'єктів із S, розширення якої містить s_1' і s_2' (відповідно міститься водночас у s_1' і s_2').

Теорема 8.

1. Об'єднання і перетин символьних об'єктів існує завжди і є комутативним і асоціативним, тому справедливі такі відношення:

$$s_1 \cap (s_2 \cup s_3) = (s_1 \cap s_2) \cup (s_1 \cap s_3);$$

$s_1 \cup (s_2 \cap s_3) = (s_1 \cup s_2) \cap (s_1 \cup s_3)$.

2. Із $s = s_1 \cup s_2$ не випливає, що $s' = s'_1 \cup s'_2$, а тільки що $(s'_1 \cup s'_2) \subseteq s'$. З іншого боку, коли $s = s_1 \cap s_2$, то $s' = s'_1 \cap s'_2$.

Нехай задані символьний об'єкт s і $d(s)$ - множина елементарних подій, кон'юнкція яких визначає s. Також нехай $c(s)$ - множина усіх елементарних подій, розширення яких містять s' і s^o - кон'юнкція усіх елементів із $c(s)$.

Теорема 9.

1. $d(s^o) = c(s)$ і $(s^o)' = s'$.

2. $s^o = \{ \cup \gamma(\omega_i) : \omega_i \in |s|_\Omega \}$.

3. $s^o = \{ \cap e_i : e_i \subseteq c(s) \}$.

Означення 12. Важливість об'єкта $s \in S$ описується різницею поміж об'єднанням розширень елементарних подій, що визначають s, $U(e_i(s))$, і його розширенням $s' = \Pi(e'_i(s))$, тобто

$$R(s) = \frac{1}{\operatorname{card} \Omega} \operatorname{card} \left(U(e'_i(s)) - \Pi(e'_i(s)) \right).$$

Означення 13. Простота символьного об'єкта $s \in S$ - це найменше число $S_i(s)$ елементарних подій, таких, що розширення їх кон'юнкції збігається з розширенням s. Об'єкт s зветься простим, коли $\operatorname{card} d(s) = S_i(s)$.

Означення 14. Символьний об'єкт s є повним тоді і тільки тоді, коли $c(s) = d(s)$.

Теорема 10.

1. $\forall s \in S$, s^o - повний (тобто $d(s^o) = c(s^o)$).

2. Коли символьний об'єкт s - об'єднання чи перетин символьних об'єктів, тоді s - повний.

Повні символьні об'єкти узагальнюють визначення "поняття" у машинному навчанні як "висловлювання, що описує деяку загальну інформацію про об'єкти, які належать до даного класу". Тому повні символьні об'єкти можуть виступати як змістовні представники відповідних класів у задачах машинного навчання. Загальна схема алгоритму описується таким чином:

1. На основі існуючих даних визначити типи символьних об'єктів. У найпростішому випадку - це елементарні події вигляду $[y_i = V_i]$.

2. Визначити для кожного класу об'єкти, що знаходяться у відношенні часткового порядку, та ізольовані елементи.

3. За допомогою операцій об'єднання для ізольованих елементів і перетину для об'єктів, що знаходяться у відношенні часткового порядку, для кожного класу породити множини повних об'єктів.

4. На основі тестуючої множини прикладів для кожного класу визначити найкращі підмножини повних об'єктів, котрі у подальшому застосовуються як представники відповідних класів при розв'язанні задач класифікації.

У розділі 2.2 розглянуто алгоритми, в основі яких лежить побудоване за тими чи іншими принципами дерево рішень.

Формально дерево рішень визначається як структура, створена із:

а) листових вузлів (чи вузлів відповіді), які утримують назву класу;

б) нелистових вузлів (чи вузлів рішення), які утримують атрибутний тест, пільних з іншими вузлами у відповідності зі значеннями тестуємого атрибута.

Кожний приклад описується списком пар атрибут-значення і приписується до того чи іншого класу. Множину атрибутів, що використовується для опису прикладів, позначимо через A, a - конкретний атрибут, $a \in A$, $1 \leqslant i \leqslant |A|$, де $|A|$ - число атрибутів. Для кожного атрибута a множину можливих значень позначимо через V_i, v_{ij} - конкретне значення атрибута, де $1 \leqslant j \leqslant |V_i|$, і $|V_i|$ - число значень, котре може набувати атрибут a_i.

Алгоритм побудови дерева рішень представлений у розділі 2.2.1 і складається із двох кроків:

1. Коли усі приклади належать одному класу, тоді дерево рішень - листовий вузол, що утримує ім'я класу.

2. У противному випадку:

а) визначається a_{best} як атрибут з найменшою E-мірою;

б) для кожного значення $v_{best,i}$ атрибута a_{best} рекурсивно будуються дерева на основі прикладів, що мають значення $v_{best,i}$ атрибута a_{best}.

E-міра обчислюється наступним чином. Для даного вузла

нехай:

p - число позитивних прикладів;

n - число негативних прикладів;

p_{ij} - число позитивних прикладів із значенням v_{ij} атрибута a_i.

n_{ij} - число негативних прикладів із значенням v_{ij} атрибута a_i.

$$E(a_i) = \sum_{j=1}^{|v_i|} \frac{p_{ij} + n_{ij}}{p + n} I(p_{ij}, n_{ij}),$$

$$I(x,y) = \begin{cases} 0, \text{ при} & x = 0, \\ 0, \text{ при} & y = 0, \\ -\frac{x}{x+y} \log \frac{x}{x+y} - \frac{y}{x+y} \log \frac{y}{x+y} & \text{в іншому випадку.} \end{cases}$$

E-функція - це теоретико-інформаційна міра, що заснована на ентропії. Ця функція оцінює міру невизначеності у класифікації при використанні вибраного атрибута у вузлі рішення. E-метрика володіє деяким недоліком, вона дає при побудові дерева перевагу атрибутам з більшим числом можливих значень. Поліпшений варіант метрики, який переборює цей недолік, зветься метрикой відношення і має вигляд

$$E^r(a_i) = \begin{cases} \frac{I(p,n) - E(a_i)}{IV(a_i)}, & \text{якщо } I(p,n) - E(a_i) \geqslant g, \\ -\infty & \text{у противному випадку.} \end{cases}$$

де $$IV(a_i) = -\sum_{j=1}^{|v_i|} \frac{p_{ij} + n_{ij}}{p + n} \log \frac{p_{ij} + n_{ij}}{p + n};$$

$$g = \frac{1}{|A|} \sum_{i=1}^{|A|} (I(p,n) - E(a_i)).$$

Даний алгоритм дозволяє побудувати для несуперечної множини прикладів повне дерево, тобто дерево, на основі котрого кожний приклад із початкової множини буде класифікуватися вірно.

При навчанні на більших кількостях прикладів очевидна необхідність побудови інкрементних алгоритмів. На відміну

від алгоритмів, котрі будують набори правил на основі всієї множини прикладів, інкрементні алгоритми працюють у режимі почергового надходження прикладів. При надходженні нового прикладу, коли необхідно, набір правил коригується з найменшими витратами.

У розділі 2.2.2 запропоновано чотири інкрементних алгоритми. Відповідні оцінки складності наведені в таблиці, де d - число атрибутів, b - максимальна кількість значень атрибутів, n - число навчаючих прикладів.

Таблиця

Алгоритм	Прості операції	Обчислення E-міри
ID3	$O(nd^2)$	$O(b^d)$
ID3'	$O(n^2 \cdot d^2)$	$O(n \cdot b^d)$
ID5R	$O(n \cdot d \cdot b^d)$	$O(n \cdot b^d)$
ID5	$O(n \cdot b^d)$	$O(n \cdot d^2)$

В останні роки було розроблено декілька методів "підрізання" дерев рішень. Ці підходи різняться застосуванням різних критеріїв, які використовуються при підрізанні дерева, що створює деяку складність для їх порівняння.

У розділі 2.2.3 представлено новий підхід до "підрізання" дерев рішень, що базується на мінімізації помилки класифікації методу і модифікованому байєсовському підході до оцінки ймовірностей.

З використанням m-імовірної оцінки "статична помилка" у даному вузлі

$$E_s = 1 - \frac{n + p_{ac} m}{N + m} = \frac{N - n_c + (1 - p_{ac}) m}{N + m},$$

де

N - сумарне число прикладів у вузлі;

T_c - число прикладів класу C, що мінімізує E_s;

для даного m p_{ac} - апріорна ймовірність класу C;

m - параметр методу.

Динамічна помилка E_b обчислюється з врахуванням оцінки помилок класифікації піддеревами:

$$E_b = \sum p_i E_i,$$

де p_i - ймовірність того, що об'єкт буде помилково

класифіковано 1-м піддеревом, E_1 - помилка для 1-го піддерева.

Запропонований алгоритм застосовує правило: коли у даному вузлі динамічна помилка більше, ніж статична помилка, то вузол підрізається.

Вибір конкретного значення параметра m залежить від ступеня зашумленості початкових даних і може запропоновуватися експертом у даній предметній області або визначатися в автоматичному режимі , максимізуючи точність класифікації на незалежному наборі даних (тестовій множині).

При побудові дерев рішень на предметних областях з досить великою кількістю атрибутів це може призводити до знач..ого розростання дерева. Одним з шляхів переборення цієї складності є конструктивна індукція, тобто автоматичне породження нових атрибутів.

У розділі 2.2.4 запропоновано ряд нових алгоритмів конструктивної індукції, які використовують диз'юнктивне породження нових атрибутів у вузлах дерева рішень.

Нехай V - множина із n булевих атрибутів і C - множина усіх кон'юнкцій, заснованих атрибутами із V. Представимо диз'юнктивну нормальну форму (ДНФ) над множиною V у вигляді $f = C_1 + C_2 + \ldots + C_m$, де $C_i \in C$ і m - позитивне число. Для кожного i, $1 \leqslant i \leqslant m$, позначимо через V_i множину атрибутів у термі C_i і через K_i - число елементів цієї множини.

Означення 16. f є μДНФ формулою, коли для кожного $1 \leqslant i, j \leqslant n$, $i \neq j$, $V(C_i) \cap V(C_j) = (\emptyset)$.

Без втрати спільності припустимо, що $\bigcup_{i=1}^{m} C_i = V$. Значення функції f на конкретному наборі значень позначимо через $f(x)$.

Нехай T - дерево рішень над множиною змінних V. Кожний набір значень атрибутів визначає шлях із кореневого вузла T до листових вузлів. Позначимо через $L(x)$ листовий вузол і нехай $T(x)$ - клас, котрим позначено листовий вузол. Будемо говорити, що T еквівалентно f, коли на кожному наборі значень атрибутів виконується $T(x) = f(x)$.

Теорема 11. Нехай f - μДНФ формула над множиною змінних V. Будь-яке дерево рішень, еквівалентне f, має не менше

$k_1 \cdot k_2 \cdot \ldots \cdot k_m$ листів.

Показником спроможності породження нових атрибутів (ПСПНА) для даного поняття може служити відношення числа можливих нових атрибутів до загального числа вузлів дерева, що представляє поняття.

Із попередньої теореми випливає:

Теорема 12. μ-k-term-1 ДНФ, представлена у вигляді бінарного дерева, має не менше ніж 1^k листів і $1^k - 1$ тестових вузлів і ПСПНА не менше ніж $1 - 1^k/(1^k - 1)$.

Класичні системи індуктивного виводу застосовують для представлення прикладів простішу мову представлення знань типу атрибут-значення. Але ж очевидно, що використання знань про предметну область може істотним чином вплинути на якість отриманих гіпотез і обмежити необхідний перебір.

У розділі 2.2.5 як мова представлення гіпотез використовується нерекурсивна мова типізованих хорновських класів з запереченням, тобто гіпотези, які мають вигляд $A \leftarrow L_0, \ldots, L_n$, де A - атом (предикатний символ над термом) L_i - літерал (атом з/без запереченням). Під типізованістю мови мається на увазі, що для кожного аргументу визначено тип, тобто множина його можливих значень. Це може бути дискретний набір, неперервний інтервал і т.д. Позитивні і негативні приклади описуються у вигляді відношень поміж атрибутами об'єкта, знання про предметну область складаються із предикатних визначень у формі хорновських класів з запереченням (можливо, рекурсивних).

Основна ідея полягає в об'єднанні методів індуктивного виводу з більш широкими можливостями мов представлення знань типу предикатної логіки першого порядку, що дозволяє на етапі індукції використовувати експертні знання.

На першому кроці алгоритму знання про предметну область, позитивні і негативні приклади перетворюються у представлення атрибут-значення. Один із відомих алгоритмів індуктивного виводу генерує набір правил у вигляд if-then, після чого ці правила перетворюються знову у предикатну форму. Знання про предметну область визначають відношення поміж атрибутами і описуються предикатними визначеннями.

Відрізняються утилітні предикати та утилітні функції. Утилітна функція - це предикатне визначення з вказівкою вхідних і вихідних аргументів, утилітний предикат припускає наявність тільки вхідних аргументів.

Утилітні предикати можуть бути симетричними відносно деяких пар аргументів одного типу. Для прикладу бінарний предикат q(X,Y) симетричний по X і Y, коли X, Y належать до одного типу T і q(X,Y)=q(Y,X) для усіх значень X і Y. На аргументах одного типу визначається симетричний утилітний предикат "рівність" (=).

Запропонований підхід має ряд переваг у порівнянні з чисто індуктивними алгоритмами. По-перше, допускається опис прикладів у вигляді відношень, що більш зрозуміло з точки зору експерта, по-друге, на етапі індукції можуть використовуватися знання про предметну область, що у кінцевому підсумку приводить до отримання більш загальних описів.

Одним із теоретичних напрямків в області машинного навчання є визначення максимально можливих класів понять, котрі допускають навчання на основі прикладів за прийнятний час. Зручним представленням для булевих функцій є так звані листи рішень.

У розділі 2.3.1 поняття листа рішень узагальнено на багатомірний випадок, модифіковано та проведено розробку відповідних алгоритмів. Нехай N - множина атрибутів; a_i - i-й атрибут, $1 \leqslant i \leqslant n$; A_i - множина значень i-го атрибута, $|A_i|$ - кількість різних значень атрибута a_i; a_{ij} - значення i-го атрибута j-го об'єкта, де $1 \leqslant j \leqslant m$, m - число об'єктів, які описують поняття. Кожний об'єкт приписується до одного із класів $P_k \in P$, $1 \leqslant k \leqslant l$, де l - число класів. Очевидно, що $l \leqslant m$. Кожний об'єкт з відповідним класом зветься прикладом S. Назвемо k-КНФ кон'юнктивну нормальну форму, кожний клас якої містить не більше k літералів, відповідно k-ДНФ - диз'юнктивну нормальну форму, кожний терм якої містить не більше k літералів. K-clause-КНФ означає КНФ, що складається не більше ніж із k класів, k-term-ДНФ - ДНФ, яка складається не більше ніж із k термів. Відмітимо, що k-term-ДНФ є

підмножиною k-КНФ і k-clause-КНФ - підмножиною k-ДНФ.

Означення 17. Лист рішень - це список L пар $(f_1,v_1),\ldots,(f_r,v_r)$, де кожний f_i - це терм із числа $W=\sum_{i=1}^{k} C_i^n 2^i = O(n^k)$, кожне v_i - значення із P і остання функція f_r - постійна функція "істина". У випадку, коли $v_r\in\{0,1\}$, певним способом визначається булева функція: для кожного набору $x\in X_n$ $L(x)$ дорівнює v_j, де j - найменший індекс, такий, що $f_j(x)=1$. Лист рішень формалізує правило типу if-then-elseif...else.. Позначимо лист рішень, що складається із термів довжини, меншої чи рівної k на множині із n змінних, через k-ЛР(n). Аналогічно k-ДНФ і k-КНФ на множині із n змінних позначимо через k-ДНФ(n) і k-КНФ(n), через k-ДР(n) - дерево рішень з довжиною шляху від кореневого до листових вузлів не більшею k.

Для бінарного випадку справедливі наступні теореми:

Теорема 13. Для $0\leqslant k\leqslant n$ k-КНФ(n) і k-ДНФ(n) є строгими підмножинами k-ЛР(n).

Теорема 14. Для $0\leqslant k\leqslant n$ k-clause-КНФ(n) і k-term-ДНФ(n) - строгі підмножини k-ЛР(n).

Теорема 15. Для $0\leqslant k\leqslant n$ і $n>2$ (k-КНФ(n) $\cup$ k-ДНФ(n)) - строга підмножина k-ЛР(n).

Теорема 16. Для $0\leqslant k\leqslant n$ k-ДР(n) - строга підмножина k-ЛР(n).

Задача навчаючого алгоритму - вибрати із простору гіпотез F ту гіпотезу (функцію), котра відповідає навченому поняттю. Весь простір F можна розбити по числу змінних, що визначають поняття $F=\bigcup_{n=1}^{\infty} F_n$. Взагалі навчаючий алгоритм починає функціонування з невеликого n, збільшуючи його за необхідністю. Складність навчання поняттю, котре вибирається із F_n, залежить від розміру F_n. Показником такої складності може служити величина $\lambda=\log|F_n|$.

Означення 18. Простір F називається поліноміально-визначеним, коли існують алгоритм A і поліном $p(\lambda,m)$, такі, що для даного числа n і множини із m прикладів поняття S, A за час $p(\lambda_n,m)$ генерує функцію $f\in F_n$, відповідну S, якщо така існує.

Нехай $Y_n=(y_1,\ldots,y_n)$ - множина змінних, $y^*=(y_1^*,\ldots,y_n^*)$ -

конкретний набір значень змінних $(y_1,\ldots,y_n)$. Покладемо, що на множині Y_n визначено ймовірний розподіл $\mathfrak{P}_n$ і навчаюча множина прикладів формується у відповідності з $\mathfrak{P}_n$. Мірою якості алгоритму A є ймовірність того, що новий приклад, вибраний у відповідності з $\mathfrak{P}_n$, буде класифікуватися невірно. Для формалізації визначимо розходження $f \odot g$ поміж двома функціями f і g на F_n як ймовірність того, що f і g різні:

$$f \odot g = \sum \mathfrak{P}_n(y^*),$$
$$y^* \mid f(y^*) \neq g(y^*).$$

Коли f - "вірна функція", то $f \odot g$ визначає помилку класифікації при використанні функції g. Ми будемо створювати алгоритми, генеруючі функції, які визначають поняття з вибраною раніше помилкою класифікації ε, $0 \leqslant \varepsilon \leqslant 1$. Однак ми не можемо вимагати від таких алгоритмів отримання бажаної точності завжди. Для формалізації "найгіршого випадку" у виборі прикладів у відповідності з $\mathfrak{P}_n$ введемо параметр γ, $0 \leqslant \gamma \leqslant 1$ і вимагатимемо, щоб з ймовірністю не меншою $1-\gamma$ алгоритми генерували потрібні рішення.

Означення 19. Простір F називається таким, що поліноміально навчається, коли існує алгоритм A і поліном $s(\cdot,\cdot,\cdot)$, такі що для усіх n, ε і γ, усіх ймовірних розподілів P_n на Y_n і усіх функцій $f \in F_n$, A з ймовірністю не меншою, ніж $1-\gamma$ на основі множини навчаючих прикладів розміром $m = s(\lambda_n, 1/\varepsilon, 1/\gamma)$, вибраної у відповідності з $\mathfrak{P}_n$, генерує функцію $g \in F_n$, таку, що $g \odot f < \varepsilon$.

Теорема 17. Клас k-ЛР(n) є таким, що поліноміально навчається.

У розділі 2.3.2 запропоновано два алгоритми "підрізання" листів рішень, що підвищують функціональні можливості відповідних методів.

У розділі 2.4 розглянуто генетичні алгоритми, котрі засновані на стандартних моделях спадковості та еволюції із області популяційної генетики і є модельним втіленням природних механізмів адаптації. Описані і досліджені джерела ефективності генетичних операторів - кросинговера, мутації та інверсії, розроблена експериментальна система Genat, що

заснована на генетичних алгоритмах. Система використовує модель представлення знань типу "атрибут - значення" і генерує правила вигляду "коли Умова, то ...". На відміну від більшості аналогічних систем Genat дозволяє генерувати правила різної довжини, що фактично змінює початковий простір представлення. Отримані таким чином визначальні фрагменти потім розглядаються як нові атрибути і використовуються для побудови дерев рішень. Система дозволяє прослідкувати роль оператора інверсії у побудові "гарних" правил і має можливості для гнучкого регулювання ймовірності застосування оператора мутації.

Для рішення задач машинного навчання також можуть бути використані теоретично добре розроблені комітетні методи. Для побудови комітета з мінімальним числом членів застосовується метод згортання З.Н.Черникова. Однак практичне використання цього методу для систем з великим числом нерівностей обмежене через значну обчислювальну складність. Більш застосовуються на практиці алгоритми, які використовують зниження розмірності простору.

У розділі 2.5 запропоновано нове перетворення, яке дозволяє будувати комітети, близькі до оптимальних, за прийнятний час. Для системи нерівностей $(c_j,x)>0$, $j\in J$, де J - множина індексів, $J^1\subset J$ відображення $\Phi_{min}: R_n \to R_2$ визначається матрицею P із стовпцями f і l, що мінімізує

$$\sum_{p=1}^{f}\left[\frac{\sum_{i=1}^{n} c_{ki}f_i}{\sum_{i=1}^{n} c_{ki}l_i}-\frac{\sum_{i=1}^{n} c_{pi}f_i}{\sum_{i=1}^{n} c_{pi}l_i}\right]^2,$$

де $f=|J^1|-1$, $p\neq k$, $k\in J^1$.

Проведено порівняльний аналіз комітетних алгоритмів, які використовують різні перетворення, який показав практичну цінність запропонованого перетворення.

У главі 3 описуються практичні застосування запропонованих у роботі методів індуктивного виводу. Всі розрахунки виконувались з використанням розроблених автором системи індуктивного виводу під назвою АВС і системи TOPOS,

орієнтованих на IBM PC/AT. Успішно розв'язувалися задачі прогнозування властивостей нових наплавочних матеріалів, задачі із галузі захисту рослин та задачі прогнозування органолептичних показників безалкогольних напоїв.

ОСНОВНІ ВИСНОВКИ

Основні результати роботи полягають у наступному:

- розроблені та досліджені алгоритми індуктивного виводу знань, які засновані на побудові дерев і листів рішень;
- розроблені алгоритми "конструктивної" індукції на деревах рішень, які дозволяють автоматично будувати нові атрибути, що найкращим чином описують предметну область;
- запропоновано методи, які дозволяють використовувати експертні знання у процесі індукції;
- всі розроблені методи програмно реалізовані у вигляді оболонки експертної системи індуктивного виводу;
- доведено теореми про складність відповідних методів;
- розроблена теорія символьних об'єктів, яка дозволяє дати логічну трактовку поняттю "гарна гіпотеза".

Застосування у системах штучного інтелекту методів автоматичної генерації знань суттєво підвищує їх функціональні можливості, прискорює при цьому процес утворення відповідних баз знань.

Запропоновані у дисертаційній роботі методи дозволили успішно вирішити цілий ряд практичних задач із різних предметних областей (біології, захисту рослин та ін.).

1. Гупал А. М., Цветков А. М. Разработка алгоритмов индуктивного вывода знаний с использованием деревьев решений // УСиМ. — 1992. — № 5. — С. 27—31.

2. Гупал А. М., Цветков А. М. Разработка алгоритмов индуктивного вывода знаний с использованием листьев решений // Материалы 1-го междунар. науч.-техн. семинара «Теоретические и прикладные проблемы моделирования предметных областей». — Киев. — 1992. — С. 64—69.

3. Гупал А. М., Цветков А. М. Разработка и реализация алгоритмов индуктивного вывода // Использование математических методов и ЭВМ в системах управления и проектирования. — Киев: Ин т кибернетики им. В. М. Глушкова АН Украины, 1991. — С. 33—37.

4. Цветков А. М. Инкрементные алгоритмы построения деревьев решений // Материалы 2-го междунар. науч.-техн. семинара «Теоретические и прикладные проблемы моделирования семинара «Теоретические и прикладные проблемы моделирования предметных областей». — Киев. — 1993. — С. 129—133.

5. Цветков А. М. Использование экспертных знаний в процессе индукции // Там же. — С. 62—66.

6. Гупал А. М., Цветков А. М. Об одном методе индуктивного вывода, основанном на комитетных конструкциях // Кибернетика и системный анализ. — 1992. — № 5. — С. 159—161.

7. Цветков А. М. Разработка алгоритмов индуктивного вывода, основанных на построении деревьев решений // Там же. — 1993. — № 3. — С. 174—178.

8. Гупал А. М., Цветков А. М., Пономарев А. А. Об одном методе индуктивного вывода, основанном на «подрезании» деревьев решений // Там же. — 1993. — № 5. — С. 174—178.

9. Гупал А. М., Цветков А. М., Пономарев А. А. Методические указания по изучению экспертных систем. — Киев, 1993. — 48 С.

Підп. до друку 14.07.94. Формат 60×84/16. Папір друк. № 2. Офс. друк. Ум. друк. арк. 0,93. Ум. фарбо-відб. 1,05. Обл.-вид. арк. 1,0. Тираж 100 прим. Зам. 794.

Редакційно-видавничий відділ з поліграфічною дільницею
Інституту кібернетики імені В. М. Глушкова НАН України
252650 Київ МСД 22, проспект Академіка Глушкова, 40.