І н ф о р м а т и з а ц і я т а б е з п е к а

УДК 681.3

ЛАНДЕ Д.В., *доктор технічних наук, старший науковий співробітник,*
Інститут проблем реєстрації інформації НАН України,
ФУРАШЕВ В.М., *кандидат технічних наук, старший науковий співробітник, доцент*

ПИТАННЯ КЛАСИФІКАЦІЇ ТА РОЗПІЗНАВАННЯ ІНФОРМАЦІЇ ПРИ ПОБУДОВІ ІНФОРМАЦІЙНО-АНАЛІТИЧНИХ СИСТЕМ

Анотація. *Методологічні підходи до класифікації та розпізнавання інформації при побудові автоматизованих інформаційно-аналітичних систем.*

Аннотация. *Методологические подходы к классификации и распознаванию информации при создании автоматизированных информационно-аналитических систем.*

Summary. *The methodological approaches to classification and recognition of information when creating automated informative- analytical systems.*

Ключові слова: *класифікація інформації, розпізнавання інформації, інформаційно-аналітична система, методи розпізнавання інформації.*

В умовах все зростаючої швидкоплинності процесів та їх динаміки, які відбуваються у сучасному світі, суттєво зростає роль оперативності реагування на ці зміни, зокрема, оперативності прийняття різноманітних рішень, причому у визначений проміжок часу. “Обтяжуючим” фактором при цьому для особи, яка приймає рішення, є фактор постійно зростаючих обсягів потрібної інформації та її виокремлення зі всього масиву доступної інформації. Цілком зрозуміло, що без “помічників” здійснити це неможливо. Такими “помічниками”, причому досить ефективними, на даний час виступають різноманітні автоматизовані інформаційно-аналітичні системи, на які покладаються функції пошуку, класифікації, розпізнавання, первісної обробки даних та, за необхідністю, наступне відслідковування її змін у визначеному проміжку часу.

Створення автоматизованих інформаційно-аналітичних систем – складна задача, яка вимагає зчислених теоретичних і експериментальних досліджень, спрямованих на впорядкування потоків інформації, розробки та впровадження оптимальних алгоритмів класифікації та розпізнавання даних, потужного математичного та програмного апарату. До найбільшого впливу різних чинників схильна саме класифікаційна система ознак. Від того, наскільки добре буде продумана ця система, багато в чому залежить ефективність і достовірність результатів всієї інформаційно-аналітичної системи.

Відомо, що класифікація – це процес групування об’єктів дослідження або спостереження відповідно до їх загальних ознак. В результаті розробленої класифікації створюється класифікаційна система (часто називають так само, як і процес – класифікацією). З питаннями класифікації тісно пов’язана теорія розпізнавання образів – це розділ кібернетики, що розвиває теоретичні основи і методи класифікації та ідентифікації предметів, явищ, процесів, сигналів, ситуацій і т. п. об’єктів, які характеризуються скінченним набором деяких властивостей і ознак. Розпізнавання в порівнянні з класифікацією має додаткову функцію – ідентифікації. Наведемо деякі задачі, що вимагають застосування методів розпізнавання та класифікації інформації:

• аналіз мережних структур: роумінг, біллінг; Інтернет, соціальні мережі, форуми, блоги; бази даних юридичних осіб (спільна участь у бізнесі); спеціальні бази даних (спільна участь суб’єктів у злочинах, спільне покарання); виявлення кореляцій, прихованих зв’язків;

• аналіз інформаційних потоків, інформаційного простору: ЗМІ; Інтернет; виявлення тенденцій та аномалій;

• обробка інформації з великою кількістю параметрів: графічні файли; звукові файли; біометричні дані.

Деякі методи класифікації інформації. Стандартна постановка завдання розпізнавання (класифікації) полягає в наступному [1]. Досліджується деяка множина об’єктів D. Об’єкти цієї множини описуються деякою системою ознак. Передбачається, що множину D можна представити у вигляді об’єднання непересічних підмножин (класів) $K_1,...,K_l$.

Нехай є скінченний набір об’єктів $S_1,...,S_m$ з D, і відомо, яким класам вони належать (це прецеденти або навчальні об’єкти). Потрібно по пред’явленому набору значень ознак об’єкта визначити, до якого класу він належить.

Для вирішення прикладних завдань класифікації і розпізнавання успішно застосовуються методи, засновані на комбінаторному аналізі ознакових описів об’єктів, які особливо ефективні у разі, коли інформація цілочисельна і число припустимих значень кожної ознаки невелике. При конструюванні цих методів використовується апарат дискретної математики, зокрема, булевої алгебри, теорії диз’юнктивних нормальних форм і теорії покриттів булевих і цілочисельних матриць. Основоположними роботами є роботи Ю.І. Журавльова, С.В. Яблонського і М.Н. Вайнцвайга [2 – 5].

Головною особливістю процедур розпізнавання, які будемо називати надалі дискретними або логічними процедурами, є можливість отримання результату за відсутності інформації про функції розподілу значень ознак і за наявності малих навчальних вибірок. Не потрібне також завдання метрики в просторі описів об’єктів. В даному випадку для кожної ознаки визначається бінарна функція близькості між її значеннями, що дозволяє розрізняти об’єкти і їх підписи.

Методи класифікації лежать на стику двох областей – машинного навчання (machine learning, ML) та інформаційного пошуку (information retrieval, IR) [6 – 8]. Відповідно автоматична класифікація може здійснюватися:

- на основі заздалегідь заданої схеми класифікації і вже наявної множини класифікованих об’єктів;
- повністю автоматизовано.

При застосуванні підходів машинного навчання класифікаційне правило будується на основі тренувальної колекції текстів.

Завдання класифікації полягає у визначенні приналежності об’єкта, який розглядається, одному або декільком класам. Класифікація може визначатися загальною тематикою текстів (якщо об’єкти – тексти), наявністю певних дескрипторів або виконанням певних умов, іноді досить складних.

Для кожного класу експерти відбирають набори типових об’єктів, які використовуються системою класифікації в режимі навчання. Після того як навчання закінчене, система за допомогою спеціальних алгоритмів зможе розподіляти вхідні потоки інформації щодо об’єктів за класами.

Класифікацію можна розглядати як завдання розпізнавання образів, при такому підході для кожного об’єкта виділяються набори ознак. У разі текстів ознаками є слова і взаємозалежні набори слів, які містяться в текстах. Для формування набору ознак для

кожного об’єкта використовуються лінгвістичні і статистичні методи. Ознаки групуються в спеціальну таблицю – інформаційну матрицю. Кожен рядок матриці відповідає одному з класів, кожен елемент рядка – одній з ознак; чисельне значення цього елементу визначається в процесі навчання системи класифікації. Існуючі алгоритми дозволяють проводити класифікацію з досить високою точністю, проте результати досягаються за рахунок великих розмірів інформаційної матриці.

Наведемо формальний опис задачі класифікації. Нехай $D=\{d^{(1)}, ..., d^{(N)}\}$ – множина об’єктів, $C=\{c_1, ..., c_M\}$ – множина категорій, Ф – цільова функція, яка по парі $\langle d^{(i)}, c_j \rangle$ визначає, чи відноситься об’єкт $d^{(i)}$ до категорії c_j (1 або True) або ні (0 або False). Задача класифікації полягає у побудові функції $\tilde{\Phi}$, яка максимально наближена до Ф.

Дана класифікація називається чіткою бінарною, тобто мається на увазі, що існують тільки дві категорії, які не перетинаються. До такої класифікації зводиться багато задач, наприклад, класифікація за множиною категорій $C=\{c_1, ..., c_M\}$ розбивається на M бінарних класифікацій за множинами $\{c_i, \bar{c}_i\}$.

Часто використовується ранжирування, при якому множина значень цільової функції – це відрізок [0, 1]. Об’єкт при ранжируванні може відноситися не тільки до однієї, а відразу до декількох категорій з різним ступенем приналежності, тобто категорії можуть перетинатися між собою.

Припустимо, що для кожної категорії c_i побудована функція $CSV^{(i)}$ (статус класифікації), що відображає множину об’єктів D на відрізок [0; 1], яка задає ступінь приналежності об’єкта категорії. Розглянемо задачу, яка полягає в тому, щоб від функції ранжирування перейти до точної класифікації. Найбільш простій спосіб – для кожної категорії c_i вибрати граничне значення (поріг) τ_i. Якщо $CSV^{(i)}(d) > \tau_i$, то об’єкт d відповідає категорії c_i. Можливий і інший підхід – для кожного об’єкта d вибирати k найближчих категорій, тобто k категорій, на яких $CSV^{(i)}(d)$ приймають найбільші значення.

Вибір порогового значення можливий, наприклад, таким чином. Учбова колекція розбивається на дві частини. Для кожної категорії c_i на одній частині учбової колекції обчислюється, яка частина об’єктів їй належить. Порогові значення вибирається так, щоб на іншій частині учбової колекції кількість об’єктів, віднесених c_i, була такою ж.

Розглянемо задачу лінійної класифікації. Нехай кожній категорії C_i відповідає вектор $\vec{c}^{(i)}=(c_1^{(i)}, ..., c_N^{(i)})$, де N – розмірність простору параметрів. У якості правила класифікатора об’єктів d використовується формула:

$$CSV^{(i)}(d)=\vec{d}\cdot\vec{c}^{(i)}.$$

Нормалізація проводиться зазвичай так, щоб підсумкова формула для $CSV^{(i)}(d)$ була нормованим скалярним добутком – косинус кута між вектором категорії C_i і вектором з вагових значень параметрів $\vec{d}=(d_1,...,d_N)$, що входять до об’єкту d:

$$CSV^{(i)}(d)=\frac{\vec{d}\cdot\vec{c}^{(i)}}{|\vec{d}\,\|\,\vec{c}^{(i)}|}.$$

Координати вектора $\vec{c}^{(i)}$ визначаються в ході навчання, яке проводиться за кожною категорією незалежно від інших.

Деякі класифікатори використовують так званий профайл (profile, прототип об’єкта) для визначення категорії. Профайл – це список зважених параметрів, наявність (або відсутність) яких дозволяє найбільш точно відрізнити конкретну категорію від інших категорій. Метод, запропонований Дж. Роччіо [8], відноситься до лінійних класифікаторів, в яких кожний об’єкт представляється у вигляді вектора вагових значень параметрів. Профайл категорії C_i розглядатимемо як вектор $\vec{c}^{(i)} = (c_1^{(i)}, \ ..., \ c_N^{(i)})$ (N – кількість масиву параметрів), значення елементів якого $c_k^{(i)}$ при навчанні класифікатора в рамках методу Роччіо розраховується за формулою:

$$c_k^{(i)} = \frac{\alpha}{|POS_i|} \sum_{d^{(j)} \in POS_i} w_k^{(j)} - \frac{\beta}{|NEG_i|} \sum_{d^{(j)} \in NEG_i} w_k^{(j)},$$

де: $w_k^{(j)}$ – це вага параметра t_k в об’єкті $d^{(j)}$;

POS_i – це позитивний приклад – множина об’єктів, що належать категорії $\vec{c}^{(i)}$, тобто $POS_i = \{d^{(j)} \mid \Phi(d^{(j)}, c_i) = 1\}$;

NEG_i – негативний приклад – множина об’єктів, що не належать категорії $\vec{c}^{(i)}$: $NEG_i = \{d^{(j)} \mid \Phi(d^{(j)}, c_i) = 0\}$.

У цій формулі, α та β – контрольні параметри, які характеризують вагомість позитивних і негативних прикладів. Наприклад, якщо α = 1 і β = 0, C_i буде “центром мас” всіх об’єктів, які відносяться до відповідної категорії.

Функція $CSV^{(i)}(d)$ в цьому випадку визначається за різними методиками – або як величина, зворотна відстані від вектора з вагових значень параметрів, що входять в об’єкт d, до профайла категорії C_i, або як скалярний добуток цих векторів.

Одним з методів лінійної класифікації є метод регресії, який використовується, коли ознаки категорій можуть бути виражені кількісно у вигляді деякої комбінації векторів вагових значень параметрів, що входять до об’єктів з учбової колекції. Отримана комбінація може використовуватися для визначення категорії, до якої буде відноситися новий об’єкт.

Метод регресії є варіантом лінійної класифікації. При застосуванні регресійного аналізу до класифікації текстів розглядається множина параметрів (F) і множина категорій (C). У цьому випадку учбовій колекції об’єктів ставиться у відповідність дві матриці:

- матриця об’єктів D в учбовій колекції, в якій кожен рядок – це об’єкт, а стовпець терм;

- матриця відповідей $O = \|o_{i,j}\|$, в якій рядок i відповідає об’єкту D_i ($i = 1, \ ..., \ N$), стовпець j – категорії ($j = 1, \ ..., \ M$), а $o_{i,j}$ – значенню $CSV^{(i)}(d^{(i)})$.

Метод регресії базується на алгоритмі знаходження матриці правил M, яка мінімізує значення норми матриці $\|MD - O\|_F$, що формально записується таким чином:

$$M = \arg \min_M \|MD - O\|_F .$$

Нагадаємо, що в лінійній алгебрі під нормою матриці розуміється функція, яка ставить у відповідність цій матриці деяку числову характеристику. В даному випадку

рекомендується використовувати норму Фробеніуса $\| . \|_F$, яка дорівнює кореню квадратному з суми квадратів всіх елементів відповідної матриці.

Елемент $m_{i,j}$ шуканої матриці M відображатиме ступінь приналежності i-го параметра j-ій категорій.

До інших, імовірнісних методів класифікації, належить насамперед наївна байєсова модель. Розглядається умовна ймовірність приналежності об’єкта до класу C при тому, що він має ознаки $F_1, \ ..., \ F_n$:

$$P(C \mid F_1, \ ..., \ F_n).$$

Відповідно до теореми Байєса:

$$P(C \mid F_1, \ ..., \ F_n) = \frac{P(C)P(F_1, \ ..., \ F_n \mid C)}{P(F_1, \ ..., \ F_n)}.$$

За визначенням умовної ймовірності (при $P(F_1, \ ..., \ F_n) \equiv 1$):

$$P(C \mid F_1, \ ..., \ F_n) = P(C)P(F_1, \ ..., \ F_n \mid C) = P(c)P(F_1 \mid C)P(F_2, \ ..., \ F_n \mid C, F_1) =$$

$$= P(c)P(F_1 \mid C)P(F_2 \mid C)P(F_3, \ ..., \ F_n \mid C, F_1, F_2).$$

Відповідно до “наївного” байєсовського підходу передбачається, що події F_i, F_j незалежні для будь-яких $i \neq j$:

$$P(F_i \mid C, \ F_j) = P(F_i \mid C).$$

Відповідно:

$$P(C \mid F_1, \ ..., \ F_n) = P(C)P(F_1 \mid C)P(F_2 \mid C) \cdot ... \cdot P(F_n \mid C) = P(C)\prod_{i=1}^{n} P(F_i \mid C).$$

Перейдемо до класифікації об’єктів. У разі бінарної класифікації “наївна” байєсовська ймовірність приналежності об’єкта класу визначається за формулою:

$$P(D \mid C) = \prod_i P(w_i \mid C).$$

Відповідно до теореми Байєса:

$$P(C \mid D) = \frac{P(C)}{P(D)} P(D \mid C).$$

Допустимо, класифікація відбувається тільки по двох класах – C і $\bar{C}$. Тоді відповідно до формули Байєса маємо:

$$P(C \mid D) = \frac{P(C)}{P(D)} \prod_i P(w_i \mid C);$$

$$P(\bar{C} \mid D) = \frac{P(\bar{C})}{P(D)} \prod_i P(w_i \mid \bar{C}).$$

Як критерій приналежності об’єкта до категорії розглядається наступне відношення ймовірності приналежності і неприналежності до класу C:

$$\frac{P(C \mid D)}{P(\bar{C} \mid D)} = \frac{P(C)}{P(\bar{C})} \prod_i \frac{P(w_i \mid C)}{P(w_i \mid \bar{C})}.$$

На практиці використовується логарифм відношення ймовірності:

$$\ln\frac{P(C\mid D)}{P(\bar{C}\mid D)}=\ln\frac{P(C)}{P(\bar{C})}+\sum_i\ln\frac{P(w_i\mid C)}{P(w_i\mid\bar{C})}.$$

Якщо виконується нерівність $\ln\frac{P(C\mid D)}{P(\bar{C}\mid D)}>0$, (тобто, просто $p(C\mid D)>p(\bar{C}\mid D)$), то вважається, що об’єкт D відноситься до категорії C.

Найпопулярніший на цей час метод класифікації документальної інформації, що відноситься до групи граничних методів класифікації, – це метод опорних векторів (Support Vector Mashine, SVM), запропонований В.Н. Вапником [9, 10]. Він визначає приналежність об’єктів до класів за допомогою меж областей. Розглядатимемо тільки бінарну класифікацію. Припустимо, що кожен об’єкт класифікації є вектором в N-вимірному просторі. Кожна координата вектора – це деяка ознака, кількісно тим більша, чим більше ця ознака виражена в даному об’єкті.

Передбачається, що існує учбова колекція – це множина векторів $\{\vec{x}_1, ..., \vec{x}_n\}\in R^N$ і чисел $\{y_1, ..., y_n\}\in\{-1, 1\}$. Число y_i рівне 1 у разі приналежності відповідного вектора x_i категорії C, та – 1 – інакше. Як було показано вище, лінійний класифікатор – це один з простих способів вирішення задачі класифікації. В цьому випадку шукається гіперплощина в N-вимірному просторі, що відокремлює всі точки одного класу від точок іншого класу. Якщо вдається знайти таку пряму, то завдання класифікації зводиться до визначення взаємного розташування точки і прямої: якщо нова точка лежить з одного боку прямої (гіперплощині), то вона належить класу C, якщо з іншого – класу $\bar{C}$.

Формалізуємо цю класифікацію: необхідно знайти вектор $\vec{w}$ такий, що для деякого значення b і нової точки $\vec{x}_i$ виконується:

$$y_i=\begin{cases}+1, & \vec{w}\cdot\vec{x}_i\ge b,\\ -1, & \vec{w}\cdot\vec{x}_i<b,\end{cases}$$

де: $\vec{w}\cdot\vec{x}_i$ – скалярний добуток векторів $\vec{w}$ і $\vec{x}_i$: $\vec{w}\cdot\vec{x}_i=\sum_{j=1}^{N}w_jx_{i,j}$.

$\vec{w}\cdot\vec{x}_i=b$ – рівняння гіперплощини, яка розділяє класи.

Тобто, якщо скалярний добуток вектора $\vec{w}$ на $\vec{x}_i$ не менше значення b, то нова точка належить до першого класу, якщо менше – до другого. Виникає питання, яка з гіперплощин розділяє класи краще за все? Метод SVM базується на такому постулаті: найкраща розділяюча пряма – це та, яка максимально далеко віддалена від найближчих до неї точок обох класів. Тобто завдання методу SVM полягає в тому, щоб знайти такі вектор $\vec{w}$ і число b, щоб для всіх векторів $\vec{x}_i$ з учбової колекції було справедливо:

$$y_i=\begin{cases}+1, & \vec{w}\cdot\vec{x}_i-b\ge+1,\\ -1, & \vec{w}\cdot\vec{x}_i-b\le-1.\end{cases}$$

Умова $-1<\vec{w}\cdot\vec{x}_i-b<1$ задає смугу, яка розділяє класи. Межами смуги є дві паралельні гіперплощини з направляючим вектором $\vec{w}$. Чим ширше смуга, тим точніше можна класифікувати об’єкти, відповідно, в методі SVM вважається, що найширша смуга є найкращою.

Сформулюємо умови завдання оптимальної розділяючої смуги, що визначається нерівністю: $y_i(\vec{w}\cdot\vec{x}_i - b)\geq 1$ (так переписується система рівнянь, виходячи з того, що $y_i \in \{-1,\ 1\}$). Жодна з точок учбової вибірки не може лежати усередині цієї розділяючої смуги. При цих обмеженнях $\vec{x}_i$ і y_i – постійні як елементи учбової колекції, а $\vec{w}$ і b – змінні.

Легко бачити, що ширина розділяючої смуги рівна $2/\|\vec{w}\|$. Тому необхідно знайти такі значення $\vec{w}$ і b, щоб виконувалися приведені лінійні обмеження, і при цьому якомога менше була норма вектора $\vec{w}$, тобто необхідно мінімізувати:

$$\|\vec{w}\|^2 = \vec{w}\cdot\vec{w}.$$

Якщо припустити, що на учбових об’єктах можливо були допущені помилки експертами при класифікації, то необхідно ввести набір додаткових змінних $\xi_i \geq 0,$ помилок, що характеризують величину, на об’єктах $\{\vec{x}_1,\ ...,\ \vec{x}_n\}$. Це дозволяє пом’якшити обмеження:

$$y_i(\vec{w}\cdot\vec{x}_i - b)\geq 1-\xi_i.$$

Передбачається, що якщо $\xi_i = 0,$ то на об’єкті $\vec{x}_i$ помилки немає. Якщо $\xi_i > 1,$ то на об’єкті $\vec{x}_i$ допускається помилка. Якщо $0<\xi_i<1,$ то об’єкт потрапляє всередину розділяючої смуги, але відноситься алгоритмом до свого класу.

Завдання пошуку оптимальної розділяючої смуги можна в цьому випадку переформулювати таким чином мінімізувати суму: $\|\vec{w}\|^2 + C\sum_i \xi_i$ при обмеженнях $y_i(\vec{w}\cdot\vec{x}_i - b)\geq 1-\xi_i$, де коефіцієнт C – параметр налаштування методу, який дозволяє регулювати співвідношення між максимізацією ширини розділяючої смуги і мінімізацією сумарної помилки. Приведене завдання є задачею квадратичного програмування. Доведено, що цільова функція цієї задачі залежить не від конкретних значень $\vec{x}_i$, а від скалярних добутків між ними. При цьому цільова функція є опуклою, тому будь-який її локальний мінімум є глобальним.

Метод класифікації розділяючою смугою має два недоліки:

- при пошуку розділяючої смуги важливе значення мають тільки граничні точки;
- у багатьох випадках знайти оптимальну розділяючу смугу неможливо.

Для поліпшення методу застосовується ідея розширеного простору, для чого:

1. Вибирається відображення $\phi(\vec{x})$ векторів $\vec{x}$ в новий, розширений простір.

2. Автоматично застосовується нова функція скалярного добутку, яку використовують при рішенні задачі квадратичного програмування, так звана функція ядра (kernel function): $K(\vec{x},\vec{z}) = \phi(\vec{x})\cdot\phi(\vec{z})$. На практиці зазвичай вибирають не відображення $\phi(\vec{x})$, а відразу функцію $K(\vec{x},\vec{z})$, яка могла б бути скалярним добутком при деякому відображенні $\phi(\vec{x})$. Функція ядра – головний параметр настроювання машини опорних векторів.

3. Визначається розділяюча гіперплощина в новому просторі: за допомогою функції $K(\vec{x},\vec{z})$ встановлюється нова матриця коефіцієнтів для завдання оптимізації. При цьому замість скалярного твору $\vec{x}_i\cdot\vec{x}_j$ береться значення $K(\vec{x}_i,\vec{x}_j)$ і вирішується нове завдання оптимізації.

4. Знайшовши $\vec{w}$ і b, отримуємо поверхню, яка класифікує $\vec{w}\cdot\phi(\vec{x})-b$ в новому,

розширеному просторі.

Ядром може бути не всяка функція, проте клас допустимих ядер достатньо широкий. Наприклад, в системі класифікації контенту новин із застосуванням відомого пакету LibSVM (http://www.csie.ntu.edu.tw/~cjlin/libsvm) [11] як функцію ядра рекомендується використовувати радіальну базисну функцію:

$$K(\vec{x},\vec{z}) = \exp(-\gamma \left\| \vec{x} - \vec{z} \right\|^2),$$

де: γ – параметр, що настроюється.

Метод SVM має певні переваги:

- на тестах з документальними масивами перевершує інші методи;
- при виборах різних ядер дозволяє реалізувати інші підходи. Наприклад, великий клас нейронних мереж можна представити за допомогою SVM із спеціальними ядрами;
- підсумкове правило вибирається не за допомогою експертних евристик, а шляхом оптимізації деякої цільової функції.

До недоліків метода можна віднести:

- іноді дуже мала кількість параметрів для настройки: після того, як фіксується ядро, єдиним змінним параметром залишається коефіцієнт помилки C;
- немає чітких критеріїв вибору ядра;
- повільне навчання системи класифікації.

Існують ситуації, коли необхідно групування об’єктів за відсутністю класифікатора. Один із шляхів вирішення цієї проблеми – застосування кластерного аналізу, тобто методики автоматичного групування даних у класи. При цьому об’єкти, які потрапляють в один клас, в деякому розумінні повинні бути ближче один до одного (по відповідності параметрів), чим до об'єктів з інших класів.

Різні методології використовують різні алгоритми подібності об’єктів за наявності великої кількості ознак (разом з тим у разі роботи з HTML-документами виникають можливості урахування гіпертекстової розмітки для виявлення текстових блоків, тегів розмітки, імен доменів, URL-адрес, адресних підрядків і тому подібне). З іншого боку, як тільки методами кластерного аналізу визначаються класи, виникає необхідність їх супроводу, оскільки простір об’єктів може зростати. В цьому випадку на допомогу приходить класифікація.

Кластерний аналіз може бути ефективно використано в аналітичних задачах у різних наочних областях, де використовуються білінги. Він дозволить істотно понизити трудовитрати аналітиків в процесі обробки значних масивів інформації і вирішити задачу класифікації і виділення схожих груп об’єктів.

Механізм класифікації зазвичай навчається на відібраних об’єктах тільки після того, як закінчується стадія навчання шляхом автоматичної кластеризації – розбиття множини об’єктів на класи (кластери), смислові параметри яких заздалегідь невідомі. Кількість кластерів може бути довільною або фіксованою. Якщо класифікація допускає приписування об’єктам визначених, відомих заздалегідь ознак, то кластеризація – складніший процес, який допускає не тільки приписування об’єктам деяких ознак, а й виявлення самих цих ознак як основ формування класів. Мета методів кластеризації масивів об’єктів полягає в тому, щоб подібність об’єктів, які потрапляють в кластер, була максимальною. Тому методи кластерного аналізу базуються на таких визначеннях кластера, як множина об’єктів, значення семантичної близькості між будь-якими двома елементами яких (або значення близькості між будь-яким об’єктом цієї множини і центром кластера) не менше певного порогу.

Логічні методи розпізнавання. Серед методів розпізнавання образів, що застосовуються в інформаційно-аналітичних системах, можна виділити логічні методи, що базуються на принципі виведення логічних закономірностей або індукції правил (rule induction).

Нехай $\varphi: X \to \{0,1\}$ – деякий предикат, визначений на множині об’єктів X. Говорять, що предикат φ виділяє або покриває (cover) об’єкт x, якщо $\varphi(x) = 1$. Предикат називають закономірністю, якщо він виділяє достатньо багато об’єктів якогось одного класу c і практично не виділяє об’єкти інших класів (більш конкретне визначення буде дано нижче).

Особливу цінність представляють закономірності, які описуються простими логічними формулами. Процес пошуку правил по вибірці називають витяганням знань з даних (knowledge discovery). До знань пред’являється особлива вимога – вони повинні бути інтерпретовані, тобто зрозумілі людям.

Простим прикладом коректного алгоритму є наступний [12]. Розпізнаваний об’єкт S порівнюється з кожним з об’єктів навчання $S_1,...,S_m$. У разі якщо опис об’єкта S збігається з описом учбового об’єкта S_i, об’єкт S відноситься до того класу, якому належить об’єкт S_i, інакше алгоритм відмовляється від розпізнавання. Неважко бачити, що описаний алгоритм є коректним, проте він не зможе розпізнати жодного об’єкта, опис якого не збігається з описом жодного з учбових об’єктів.

Очевидно, що вимога повного збігу описів розпізнаваного об’єкта і одного з учбових об’єктів є дуже обережним. Аналіз прикладних завдань свідчить про те, що питання про близькість об’єктів і їх приналежності одному класу можна вирішувати на основі порівняння деякої множини їх підписів. Тому виникає питання, як вибирати набори ознак, що породжують такі підписи, по яких порівнюватимуться об’єкти. Один з варіантів відповіді на дане питання використовується в моделі алгоритмів обчислення оцінок (АОО) [1].

Початковою сировиною для побудови логічних алгоритмів класифікації служать інформативні закономірності. Множину предикатів, для яких слід шукати інформативні закономірності, називають ще простором пошуку.

Найбільш простий той випадок, коли всі початкові ознаки є бінарними $f_j: X \to \{0,1\}, \quad j = 1,...,n$. Тоді простір пошуку утворюється самими ознаками і всілякими булевими функціями, які з цих ознак можна побудувати.

Нижче розглядаються найбільш поширені методи побудови і відбору бінарних ознак.

Довільну ознаку $f_j: X \to D_f$ породжує сімейство предикатів, що перевіряє попадання значення $f(x)$ в певні підмножини множини D_f. Нижче перераховуються найбільш типові конструкції такого вигляду.

Якщо f – номінальна ознака:

$$\beta(x) = [\, f(x) = d \,], \quad d \in D_f;$$

$$\beta(x) = [\, f(x) \in D' \,], \quad D' \subset D_f.$$

Якщо f – порядкова або кількісна ознака:

$$\beta(x) = [\, f(x) \le d \,], \quad d \in D_f;$$

$$\beta(x) = [\, d \le f(x) \le d' \,], \quad d, d' \in D_f, d < d'.$$

У разі кількісних ознак $f : X \to \square$ має сенс брати тільки такі значення порогів d, які по-різному розділяють вибірку X^l. Наприклад, можна узяти пороги вигляду:

$$d_i = \frac{f^{(i)} + f^{(i+1)}}{2}, \quad f^{(i)} \neq f^{(i+1)}, \quad i = 1,...,l-1,$$

де: $f^{(1)} \leq ... \leq f^{(l)}$ – послідовність значень ознаки f на об’єктах вибірки $f(x_1),..., f(x_l)$, впорядкована за збільшенням (варіаційний ряд). Якщо виключити тривіальне розбиття, що обертає $\beta(x)$ в 0 або 1 на всій вибірці, то таких значень опиниться не більше l -1.

Описані способи дозволяють отримати величезну кількість предикатів. Якщо надалі вони використовуватимуться для синтезу кон’юнкцій, то для скорочення перебору має сенс відразу відібрати з них найбільш інформативні.

У разі порядкових і кількісних ознак дане завдання вирішується шляхом оптимального розбиття діапазону значень ознаки на зони.

Нехай $f : X \to \square$ – числова ознака, $d_1,...,d_r$ – зростаюча послідовність порогів. Зонами значень ознаки f називатимемо предикати вигляду:

$$\varsigma_0(x) = [f(x) < d_1];$$

$$\varsigma_s(x) = [d_s \leq f(x) < d_{s+1}], \quad s = 1,...,r-1;$$

$$\varsigma_r(x) = [d_r \leq f(x)].$$

Відповідно до алгоритму починається розбиття на “дрібні зони”. Пороги визначаються за наведеною вище формулою і проходять між всіма парами крапок x_{i-1}, x_i, рівно одна з яких належить класу c. Неважко показати, що розстановка порогів між точками класу c або між точками не класу c призведе тільки до зменшення інформативності зон.

Далі зони укрупнюються шляхом злиття трійок сусідніх зон. Саме трійок – злиття пар приводить до порушення чергування “c – не c”, в результаті деякі “дрібні зони” можуть так і залишитися такими, що не зливаються. Зони зливаються до тих пір, поки інформативність деякої зони $\varsigma_{i-1} \vee \varsigma_i \vee \varsigma_{i+1}$, що зливаються, перевищує інформативність початкових зон ς_{i-1}, ς_i і ς_{i+1}, або поки не буде отримано задану кількість зон r.

Кожного разу вибирається та трійка, при злитті якої досягається максимальний виграш інформативності.

Цей алгоритм має трудомісткість $O(l^2)$. Його можна помітно прискорити, якщо на кожній ітерації зливати не одну трійку зон, а τl трійок з достатньо великим виграшем δI_i, за умови, що вони не перекриваються. Число τ – ще один параметр алгоритму $0 < \tau < 0.5$. В цьому випадку трудомісткість становить $O(l/\sqrt{\tau})$.

Для вирішення завдання класифікації на базі теорії розпізнавання образів розроблено алгоритм обчислення оцінок (далі – АОО), запропонований академіком РАН Ю.І. Журавльовим [13] на початку 1970-х років. Він суміщає метричні і логічні принципи класифікації, узагальнюючи широкий клас алгоритмів розпізнавання в рамках єдиної конструкції. Від метричних алгоритмів АОО успадковує принцип оцінювання схожості об’єктів. Поняття схожості часто допускає декілька альтернативних формалізацій.

У деяких практичних завданнях простіше вимірювати відстані між об’єктами, ніж формувати їх описи ознак. Наприклад, до таких завдань відноситься ідентифікація підписів або класифікація за біометричними ознаками. Причому, як правило, є декілька альтернативних способів визначення функції відстані.

Основним в алгоритмі є завдання системи опорних множин Ω і обчислення оцінок $\Gamma_j(S)$ по класах $K_j, \quad j=1,...,l$.

Обчислення оцінок $\Gamma_j(S)$ безпосередньо по їх визначеннях практично неможливе, оскільки проводиться велике число підсумовувань. В основному воно йде по опорній множині. У загальному випадку припускатимемо, що заданий набір метрик $\rho_s(x,x'), \quad s=1,...,n$, причому неможливо апріорі сказати, яка з них “найправильніша”.

Від логічних алгоритмів АОО успадковує принцип пошуку кон’юнктивних закономірностей. При цьому кон’юнкції будуються над бінарними функціями близькості вигляду $\beta(x,x')=\left[\rho_s(x,x')<\varepsilon_s\right]$. В цьому випадку кожній закономірності відповідає не підмножина ознак, а підмножина метрик, так звана опорна множина. Як правило, однієї опорної множини недостатньо для надійної класифікації, тому в АОО застосовується так зване зважене голосування за системою опорних множин.

Опорна множина має високу інформативність у тому випадку, коли об’єкти, близькі по всіх метриках опорної множини, істотно частіше виявляються лежачими в одному класі, ніж у різних. Таким чином, в АОО гіпотеза про існування інформативних закономірностей виявляється еквівалентній гіпотезі компактності, властивій метричним алгоритмам.

Принцип класифікації, вживаний в АОО, називається також принципом часткової прецедентності – об’єкт x відноситься до того класу, в якому є більше учбових об’єктів, близьких до x по інформативних частинах описів ознак (опорних множинах).

Перейдемо до формального опису моделі АОО.

1. Задаються функції відстані $\rho_s : X \times X \to \mathbb{R}_+, \quad s=1,...,n$, в загальному випадку не обов’язково визначеного вище вигляду і навіть не обов’язково метрики.

2. Задається система опорних множин:

$$\Omega=\left\{\omega \mid \omega \subseteq \{1,...,n\}\right\}.$$

3. Вводиться бінарна порогова функція близькості, що оцінює схожість пари об'єктів $x,x' \in X$ по опорній множині:

$$B_\omega(x,x')=\underset{s\in\omega}{\wedge}\left[\rho_s(x,x')<\varepsilon_s\right],$$

де: ε_s – ненегативні числа, пороги $s=1,...,n$. Поріг ε_s називають точністю вимірювання ознаки f_s.

4. Вводиться оцінка близькості об’єкта $x \in X$ до класу $c \in Y$ як результат зваженого голосування близькості об’єкта x до всіх учбових об’єктів класу c за всіма опорними множинами:

$$\Gamma_c(x)=\sum_{i:x_i=c}\sum_{\omega\in\Omega}\alpha_{\omega i}B_\omega(x_i,x'),$$

де: ваги $\alpha_{\omega i}$ передбачаються нормованими на одиницю: $\sum_{i:x_i=c}\sum_{\omega\in\Omega}\alpha_{\omega i}=1$.

5. Алгоритм класифікації $\alpha(x)$ відносить об’єкт x до того класу, який набрав найбільшу суму голосів: $\alpha(x)=\arg\max_{c\in Y}\Gamma_c(x)$.

Отже, алгоритм обчислення оцінок задається системою опорних множин, порогами ε_s, $s=1,...,n$ і вагами $\alpha_{\omega i}$, $\omega\in\Omega$, $i=1,...,l$. Ці параметри оптимізуються по критерію мінімуму числа помилок на учбовій вибірці.

Описаний алгоритм збігається з більш загальним алгоритмом зваженого голосування, якщо як правила R_c узяти функції близькості:

$$R_c=\{\varphi_c=B_\omega(x_i,x)\,|\,\omega\in\Omega,\ \ y_i=c,\ \ i=1,...,l\},\ \ c\in Y.$$

Оскільки функції близькості є кон’юнкціями, то для пошуку інформативних кон’юнкцій підходять градієнтні методи. Для настройки ваги $\alpha_{\omega i}$ можна застосувати бустинг або будь-який лінійний класифікатор, наприклад, SVM.

Для застосування алгоритмів пошуку інформативних кон’юнкцій потрібно задати сімейство елементарних предикатів B. В даному випадку воно має вигляд:

$$B=\{\beta(x)=[\rho_s(x_i,x)\le\varepsilon_s]\,|\,s=1,...,n,\ \ y_i=c,\ \ i=1,...,l\}.$$

Для скорочення перебору при побудові кон’юнкцій має сенс заздалегідь відібрати лише деякі значення порогів $\{\varepsilon_s\}$, наприклад, за допомогою виділення інформативних зон. Відмітимо, що такий спосіб вибору порогів не дає гарантії їх оптимальності. Оптимізація порогів є складним комбінаторним завданням, для практичного вирішення якого розроблені численні евристичні алгоритми [14].

Ще один спосіб скорочення перебору – заздалегідь відібрати як еталони не всі учбові об’єкти, а тільки найбільш представницькі. Це еквівалентно обнулінню ваги $\alpha_{\omega i}$ для деяких об’єктів x_i, $i=1,...,l$. Зрозуміло, ці об’єкти як і раніше залишаються у вибірці і використовуються при настройці параметрів ε_s, $\alpha_{\omega i}$ і оцінюванні інформативності закономірностей.

Нарешті, необхідно врахувати, що на будову АОО накладено специфічне обмеження на процедуру нарощування кон’юнкцій – кон’юнкція не повинна містити елементарних предикатів, що відносяться до різних еталонних об’єктів. Тобто повинні бути заборонені конструкції вигляду: $\varphi_c=[\rho_s(x_i,x)\le\varepsilon_s]\wedge[\rho_t(x_j,x)\le\varepsilon_t]$, у яких $i\ne j$, оскільки вони не є функціями пари об’єктів (x_i,x). Цю обставину нескладно врахувати при реалізації циклу перебору по множині елементарних предикатів B.

Таким чином, розглянуті раніше логічні алгоритми легко пристосувати для навчання АОО, що є нескладною технічною вправою.

У роботі [15] наведене найпростіше застосування АОО для завдань розпізнавання.

Введемо ще один ряд позначень. Нехай H – деякий набір з r, $r\le n$ різних цілочисельних ознак вигляду $\{x_{j_1},\ ...,\ x_{j_r}\}$. Близькість об’єктів $S'=(a'_1,a'_2,...,a'_n)$ і $S''=(a''_1,a''_2,...,a''_n)$ з M по набору ознак H оцінюватимемо величиною:

$$B(S',S'',H)=\begin{cases}1, & if\ \ a'_{j_t}=a''_{j_t},\ t=1,...,r;\\ 0, & if\ \ a'_{j_t}\ne a''_{j_t}.\end{cases}$$

Принципова схема побудови цього алгоритму АОО наступна.

У системі ознак $\{x_1,\ ...,\ x_n\}$ виділяється сукупність різних підмножин вигляду $H=\{x_{j_1},\ ...,\ x_{j_r}\}$, $r\le n$, не обов’язково однакової потужності. Надалі виділені підмножини називаються опорною множиною алгоритму, а вся їх сукупність

позначається, як і раніше, через Ω. Задаються параметри: γ_i – параметр, який характеризує показність об’єкта S_i, $i=1,2,...,m$; P_H – параметр, який характеризує показність опорної множини H, $H \in \Omega$. Далі проводиться процедура голосування або обчислення оцінок. Розпізнаваний об’єкт S порівнюється з кожним учбовим об’єктом S_i по кожній опорній множині.

Вважається, що об’єкт S отримує голос за приналежність класу K, якщо і $S_i \in K$ описи об’єктів S і S_i збігаються по опорній множині H (в цьому випадку $B(S,S_i,H)=1$).

Для кожного класу K, $K \in \{K_1,...,K_l\}$ обчислюється оцінка приналежності $\Gamma(S,K)$ об’єкта S до класу K, яка має вигляд:

$$\Gamma(S,K)=\frac{1}{|W_K|}\sum_{S_i \in K}\sum_{H \in \Omega}\gamma_i \cdot P_H \cdot B(S,S_i,H),$$

де: $|W_K|=|K \cap \{S_1,...,S_m\}|$.

Об’єкт S відноситься до того класу, який має найбільшу оцінку. Якщо класів з найбільшою оцінкою декілька, то відбувається відмова від розпізнавання.

Очевидно, що побудований алгоритм не завжди є коректним. Для коректності алгоритму потрібне виконання системи лінійних нерівностей вказаного нижче вигляду.

Для простоти нехай $l=2$, $S_i \in K_1$ при $1 \le i \le m_1$, $S_i \in K_2$ при $m_1+1 \le i \le m$, $1 \le m_1 \le m-1$. Тоді система нерівностей має вигляд:

$$\Gamma(S_1,K_1) > \Gamma(S_1,K_2),$$
$$\cdots\cdots\cdots$$
$$\Gamma(S_{m_1},K_1) > \Gamma(S_{m_1},K_2),$$
$$\Gamma(S_{m_1+1},K_2) > \Gamma(S_{m_1+1},K_1),$$
$$\cdots\cdots\cdots$$
$$\Gamma(S_m,K_2) > \Gamma(S_m,K_1).$$

Вирішення системи зводиться до вибору параметрів γ_i, $i=1,2,...,m$, і P_H, $H \in \Omega$. У разі якщо система несумісна, знаходиться її максимальна сумісна підсистема і з вирішення цієї підсистеми визначаються значення параметрів γ_i і P_H.

Інший спосіб добитися коректності алгоритму – вибрати “хорошу” систему опорних множин. Зокрема, вибрати її так, щоб для будь-якого учбового об’єкта $S' \notin K$ було виконано $\Gamma(S',K) = 0$ і для будь-якого учбового об’єкта $S'' \in K$ було виконано $\Gamma(S'',K) > 0$. Для вирішення цієї задачі переходять до системи тестів.

Модель АОО допускає різні варіанти завдання системи опорних множин, функцій близькості і вирішального правила.

Висновки.

Якщо раніше, ще декілька років тому розпізнавання образів велося на базі відбору та впорядкування ознак об’єктів і подальшого ранжирування об’єктів на основі ідентифікаційної значимості, то зараз застосовується широкий аналіз інформаційного простору, мережних можливостей, виявлення мережних структур, прихованих зв’язків.

За рахунок підвищення потужності комп’ютерної техніки з’явилися можливості багатоаспектної класифікації та кластеризації даних, застосування біометричних підходів, пошуку в масивах мультимедійної інформації.

Використана література

1. Журавлев Ю.И. Об алгебраическом подходе к решению задач распознавания или классификации // Проблемы кибернетики. – М. : Наука, 1978. – [Вып. 33]. – С. 5 – 68.

2. Баскакова Л.В., Журавлев Ю.И. Модель распознающих алгоритмов с представительными наборами и системами опорных множеств // Журн. вычисл. матем. и матем. физ. – Т. 21. – 1981. – № 5. – С. 1264 – 1275.

3. Вайнцвайг М.Н. Алгоритм обучения распознаванию образов // Алгоритмы обучения распознаванию образов. – М. : Сов. радио, 1973. – С. 82 – 91.

4. Дмитриев А.И., Журавлев Ю.И., Кренделев Ф.П. О математических принципах классификации предметов или явлений // Дискретный анализ. – Новосибирск : ИМ СО АН СССР, 1966. – С. 1 – 17. – [Вып. 7].

5. Чегис И.А., Яблонский С.В. Логические способы контроля электрических схем // Труды Матем. ин-та им. В.А. Стеклова АН СССР. – Т. 51. – 1958. – С. 270 – 360.

6. Лившиц Ю. Курс лекций “Алгоритмы для Интернета”. – Режим доступу : // www.logic.pdmi.ras.ru/~yura/internet.html

7. Sebastiani F. Machine Learning in Automated Text Categorization. – Режим доступу : // www.nmis.isti.cnr.it/sebastiani/Publications/ACMCS02.pdf

8. Rocchio, J. Relevance feedback in information retrieval // In G. Salton ed., The SMART Retrieval System: Experiments in Automatic Document Processing, Englewood Cliffs, New Jersey, Prentice-Hall, 1971. – P. 313 – 323.

9. Vapnik V.N. Statistical Learning Theory. NY : John Wiley, 1998. – 760 p.

10. CJC Burges. A Tutorial on Support Vector Machines for Pattern Recognition. //www.music.mcgill.ca/_rfergu/adamTex/references/Burges98.pdf

11. Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines, 2001. – Режим доступу : //www.csie.ntu.edu.tw/~cjlin/papers/ libsvm.pdf.

12. Воронцов К.В. Лекции по логическим алгоритмам классификации. – Режим доступу : //www.ccas.ru/voron/download/LogicAlgs.pdf

13. Журавлев Ю.И. Непараметрические задачи распознавания образов // Кибернетика. – 1976. – № 6.

14. Рязанов В.В., Сенько О.В. О некоторых моделях голосования и методах их оптимизации // Распознавание, классификация, прогноз. – Т. 3. – 1990. – С. 106 – 145.

15. Иофина Г.В. Эффективные оценки в алгоритмах вычисления оценок // “Штучний інтелект”. – № 2’2006. – С. 155 – 159.

~~~~~~~~~~~~~ * * * ~~~~~~~~~~~~~
~~~~~~~~~~~~~