УДК 681.3

А. Пелещишин, Н. Шаховська
Національний університет "Львівська політехніка",
кафедра інформаційних систем та мереж

ЗАСТОСУВАННЯ АЛГОРИТМІВ КЛАСИФІКАЦІЇ ДЛЯ ЗМЕНШЕННЯ НЕВИЗНАЧЕНОСТІ

Описано методи побудови класифікаційних правил для усунення невизначеностей. Запропоновано алгоритми для вирішення основних задач.

The methods of uncertain eliminate are described. The main algorithms of design are described.

Вступ

У великій кількості предметних галузей потрібно опрацьовувати нечітку інформацію, причому результат аналізу даних повністю залежить від ступеня їх повноти у системі. Типовими предметними галузями появи нечіткості є соціологічна сфера (біржа праці, громадські фонди, маркетингові дослідження ринку тощо), історичні дослідження, планування господарської діяльності тощо.

Запропоновано алгоритми класифікації та класифікування об'єктів, інформацію про які зберігають у сховищі даних.

Основними проблемами, які виникають у задачах аналізу та структурування даних, є проблеми створення класів та віднесення до них об'єктів, інформація про які щойно надійшла до бази даних. Задача створення класу розбивається на дві підзадачі:

1) побудова класифікаційних функцій, за якими об'єкт класифікується як представник певного класу;

2) розбиття на класи та ідентифікація отриманих класів.

У цій статті розглянемо першу з цих підзадач.

1. Огляд сучасних досліджень та виділення невирішених проблем

На рівні кортежа у сховище даних уведено 8 типів невизначеностей [1–4].

1. Значення невідоме (відсутнє).
2. Неповнота інформації.
3. Нечіткість (використання розподілу для встановлення істинності знань).
4. Неточність (стосується числових даних).
5. Недетермінованість процедур виведення рішень (випадковість).
6. Ненадійність даних.
7. Багатозначність інтерпретацій.
8. Лінгвістична невизначність.

Розглянемо детальніше вказані типи невизначеностей та виявимо місця їх появи у відношенні.

Невизначеності типів 3 – 8, які класифікують у [1] як неоднозначність даних, переважно з'являються на рівні кортежа або підмножини значень атрибутів, з яких формується кортеж.

Відсутність інформації найчастіше зустрічається на рівні значення атрибута. Неповнота є станом кортежа, у якому відсутні значення. Нечіткість, неточність та випадковість можна віднести до фізичної невизначеності, одним із джерел якої є обмеженість у точності числових типів даних або втрата точності під час виконання математичних операцій (сюди ж відносять невизначеність,

яка виникає внаслідок роботи з інтервальнимим величинами). Ненадійність та багатозначність інтерпретацій виникає через неповне вивчення або неоднозначне відображення характеристик сутності у відношенні. У відношенні зображається за допомогою додаткового атрибута, значення якого характеризують міру довіри до цілого кортежа або підмножини значень атрибутів у кортежі. Багатозначність інтерпретації є одним із джерел виникнення протиріч. Лінгвістична невизначеність пов'язана із використнням природної мови для представлення знань, які мають якісний характер, і може виникати внаслідок нерозуміння (незнання) значення слова або нерозуміння змісту речення. Такий тип невизначеності зустрічається у системах обробки текстової інформації (системи автоматизованого перекладу, системи для самонавчання тощо).

Розглянуті типи невизначеностей можуть накладатись один на одного або бути джерелом появи один одного.

Сьогодні розроблено методи усунення відсутніх, неповних та нечітких даних [1–3], тому необхідно розробити методи, за якими можна працювати з усіма типами невизначеності.

2. Постановка задачі

Маємо деяке відношення r зі схемою R. З метою усунення невизначеності на основі відношення r необхідно побудувати множину класифікаційних правил вигляду $s(X \to Y)$, де $X, Y \subset R$, $X \cap Y = \varnothing$; X – підмножина атрибутів, на основі значень яких здійснюється віднесення до класу (усунення невизначеності по значеннях атрибута Y), Y – атрибут (підмножина атрибутів).

Вхідними даними для класифікування (віднесення до класу) є множина значень цільових атрибутів. *Цільовими атрибутами* (X) назвемо атрибути, які використовують для аналізу даних і згідно із значеннями яких здійснюють розбиття на класи. До цільових атрибутів, у першу чергу, належать усі атрибути, які входять до множини ключів. До цільових атрибутів віднесемо усі атрибути, що входять у множину лівих частин функціональних залежностей (крім первинних ключів), а також ті атрибути, які будуть впливати на ступінь довіри до отриманого результату аналізу. Крім того, для конкретної предметної галузі за допомогою експертного опитування визначають додаткову підмножину атрибутів, які вважатимуться цільовими для аналізу. Наприклад, для задачі соціологічного опитування такими атрибутами є вік, освіта, матеріальний стан тощо.

Атрибути, над якими виконують операції агрегації та порівняння, назвемо *критичними* Y (подають результати аналізу). Критичними атрибутами є атрибути, які містять числові дані, невизначеності, подані у довільному вигляді, та праві частини функціональних залежностей. До них також належать атрибути, що містять назви класів (*мітки*) [4].

Класом назвемо підмножину кортежів, для яких значення за множиною критичних атрибутів є однаковими

$$cl = \sigma_r(X = x, Y = y).$$

Для спрощення задачі вважатимемо, що класи є визначимі, а їх характеристики (тобто назви та правила, за якими об'єкт вважають представником цього класу) зберігаються у базі даних.

Віднесення до класу здійснюється на основі визначення підмножини значень цільових атрибутів. Наприклад, для класу "Студент" значення цільових атрибутів мають задовольняти умови: вік – [16, 23], освіта – {середня, середня професійна, незакінчена вища}, матеріальний стан – [50 грн., 150 грн.].

У зв'язку з тим, що важко отримати повну інформацію про об'єкти предметної галузі, то можна визначити не всі цільові атрибути. Тому для кожного класу визначають значення *межі* – величини у межах одиничного інтервалу, яка позначає мінімальний ступінь довіри до об'єкта, за яким об'єкт можна класифікувати як представника цього класу. Ступінь довіри s до об'єкта визначають як кількість цільових атрибутів із визначеними значеннями до усіх визначених цільових атрибутів цього класу (чим більше відомо про об'єкт, тим вищим буде ступінь довіри).

$$s = \sum \begin{cases} 0, cr_i \notin \text{cl} \\ 1, cr_i \in \text{cl} \end{cases},$$

де cr_i – значення за множиною цільових атрибутів.

Вважатимемо, що якщо значення атрибута визначене, то воно достовірне.

Розглянемо питання усунення невизначеності.

Віднесення до класу можна розглядати як один із способів усунення невизначеності, адже у процесі класифікування заповнюється порожнє значення атрибута, який містить значення назви класу. Крім того, класифікаційні правила можна вважати нечіткими (наближеними) функціональними залежностями.

У базі даних підтримується *нечітка функціональна залежність*

e(X→A),

якщо співвідношення кортежів, на яких виконується ця функціональна залежність, до кортежів, на яких вона не виконується, не менше, ніж s, де s – значення межі пропускання, визначене на основі експертного опитування [3]. Зрозуміло, значення s – не менше за значення межі класу .

$$e(X \to Y): \frac{COUNT(X = x, Y = y)}{COUNT(X = x, Y \neq y)} \geq s_{cl}$$

Значення межі пропускання позначатимемо ступенем багатозначної логіки Лукасевича (змінюється у межах [0, 1]).

Звідси випливає, що алгоритми усунення невизначеностей за допомогою функціональних залежностей можна застосувати для класифікування об'єктів.

Для розглянутого вище прикладу у базі даних існує класифікаційне правило *Вік, Освіта, Матеріальний стан→Соціальна група.*

3. Основний матеріал

Для того, щоби класифікувати об'єкти, необхідно побудувати функції класифікації. Взагалі у базі даних може зберігатися інформація про декілька типів класів, і для кожного типу класу є своя підмножина функцій. Одну й ту саму функцію можна застосовувати для визначення кількох типів класів.

Розглянемо алгоритм породження класифікаційних функцій (правил).

Правила можна генерувати двома способами:

- на основі аналізу характеристик класів;
- на основі існуючих правил.

3.1. Породження класифікацйних правил на основі аналізу характеристик класу

За першим способом класифікаційні правила будують на основі функціональних залежностей, що підтримуються у відношенні. Ступінь довіри до такого правила буде максимальним (A=1).

Решту атрибутів, які входитимуть до правил, визначають на основі аналізу характеристик класів.

Послідовність кроків:

1. Кортежі відношення групують за назвами класів.
2. Всередині групи відбувається почергове групування за кожним цільовим атрибутом.

3. Якщо кількість елементів підгрупи разом із порожніми значеннями не дорівнює кількості кортежів у групі класу, то обираємо інший атрибут для перевірки та переходимо на крок 2.

4. Визначаємо значення e як відношення кількості кортежів з непорожнім значенням аналізованого атрибута до кількості усіх кортежів у групі (тобто визначаємо частотну характеристику).

5. До отриманих частотних характеристик застосовуємо багатозначне "або": $u \ \& \ v = \max \{0, u + v - 1\}$,

6. До класифікаційних правил як ліва частина входять усі атрибути, частотні характеристики яких більші або дорівнюють значенню, отриманому на кроці 6, а саму частотну характеристику вважатимемо ступенем довіри до правила.

Побудова класифікаційних правил методом прогонки

Розглянемо один із способів усунення невизначених значень. Виходячи із того, що класифікаційне правило вважають наближеною функціональною залежністю із визначеним ступенем довіри A, використаємо для цього метод, аналогічний до відомого методу прогонки [2]: рівність значень атрибутів у лівій частині правила зі ступенем довіри A означає і рівність значень атрибутів у правій частині.

Опишемо алгоритм застосування модифікованого методу прогонки.

Нехай у відношенні r підтримується наближена функціональна залежність $e(X_1, ..., X_n \rightarrow A)$. Символ $\downarrow$ позначає визначене значення, а $\perp$ - його відсутність; t_i – кортеж відношення r (послідовність кортежів значення не має)

1. Якщо $\{t_1(X_1)\downarrow, ..., t_1(X_n)\downarrow\}$ і $\{t_2(X_1)\downarrow, ..., t_2(X_n)\downarrow,\}$ і $\{t_1(X_1)\downarrow, ..., t_1(X_n)\downarrow = t_2(X_1)\downarrow, ..., t_2(X_n)\downarrow\}$ і $\{t_1(A)\downarrow\}$ і $\{t_2(A) = \perp\}$, то заміняємо кожне входження $\perp$ у r на $t_1(A)$.

2. Якщо $\{t_1(X_1)\downarrow, ..., t_1(X_n)\downarrow\}$ і {в t_2 m з n значень атрибутів – $\downarrow$, n - m значень атрибутів - $\perp$, $m \le n$} і $\{e \le m/n\}$ і {за визначеними значеннями $t_1(X^m)\downarrow = t_2(X^m)\downarrow$} і $\{t_1(A)\downarrow\}$ і $\{t_2(A) = \perp\}$ і, то заміняємо кожне входження $\perp$ у r на $t_1(A)$.

3. Якщо {в t_i m_i з n значень атрибутів - $\downarrow$, $m_i \le n$}і{в t_j m_j з n значень атрибутів - $\downarrow$, $m_j \le n$}і{за визначеними значеннями $t_i(X^m)\downarrow = t_2(X^m)\downarrow$} і {за визначеними значеннями $t_j(X^m)\downarrow = t_2(X^m)\downarrow$} і $\{m_i/n \le m_j/n\}$ і $\{t_i(A)\downarrow\}$ і $\{t_j(A)\downarrow\}$ і $\{t_2(A) = \perp\}$, то заміняємо кожне входження $\perp$ у r на $t_j(A)$.

3.2. Породження класифікаційних правил на основі існуючих

У [2] обгрунтовано використання ступенів багатозначної логіки для подання довіри до правила. За такого подання праву та ліву частини правила можна вважати дискретними, і працювати з їх частинами як з окремими елементами. Оскільки у попередньому розділі показано, що класифікаційне правило вважають наближеною функціональною залежністю, то до них можна застосувати основні аксіоми виведення [3]. .Використовуючи логічні операції багатозначної логіки [1] "і" для нащадків та "або" для предків, можна генерувати нові правила на основі існуючих та автоматично визначати до них ступені довіри (які можна перевірити експериментально). Звідси випливає, що у базі даних потрібно зберігати лише мінімальне покриття наближених функціональних залежностей (тобто класифікаційних правил), а усі решта можна виводити на основі їх комбінацій з використанням операцій багатозначної логіки [1] та аксіом виведення.

Наведемо приклад генерації правил (таблиця):

Приклад генерації правил на основі існуючих

Існуючі правила	Породжені правила
$\overset{0,8}{\text{Вік, Освіта}} \rightarrow$ Соціальна група $\overset{0,4}{\text{Матеріальний стан}} \rightarrow$ Соціальна група	$\overset{0,4}{\text{Вік, Освіта, Матеріальний стан}} \rightarrow$ Соціальна група
$\overset{0,8}{\text{Вік, Освіта}} \rightarrow$ Соціальна група $\overset{0,4}{\text{Освіта}} \rightarrow$ Рівень матеріального забезпечення	$\overset{0,2}{\text{Вік, Освіта}} \rightarrow$ Рівень матеріального забезпечення, Соціальна група

Класифікаційні правила (або нечіткі функціональні залежності) доцільно зберігати в окремому відношенні (словнику), можливий варіант схеми якого показано на рисунку.

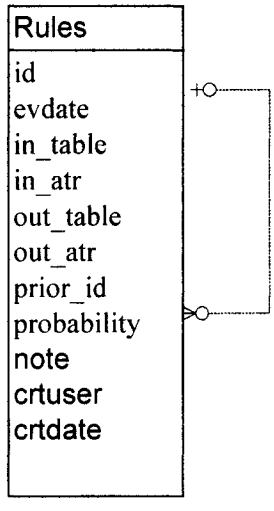

Схема відношення Rules

Схема відношення: ID – код, EVDATE – дата актуальності правила, IN_TABLE – назва таблиці-предка, IN_ATR – атрибут-предок, OUT_ATR атрибут-нащадок, OUT_TABLE – таблиця-нащадок, PRIOR_ID – зовнішній ключ таблиці Rules (для формування правил зі складеними частинами предків чи нащадків), PROBABILITY – довіра до правила.

За модифікованим методом прогонки почергово перебирають усі правила з відношення *Rules* та застосовують його до кортежів відношень, вказаних у відповідному кортежі за правилом, що застосовується.

Висновки

Опрацювання невизначеності є ключовим моментом для багатьох методів видобування даних. Існуючі сьогодні методи усунення невизначеностей опрацьовують лише відсутність, неповноту та нечіткість.

Наукова новизна. У статті запропоновано модель класу та класифікаційних правил як нечітких функціональних залежностей. Пропонуються методи визначення міри довіри до об'єктів класів.

Практична цінність. Наукові результати, отримані в статті, дозволяють провадити подальші практичні дослідження за методами класифікації з метою усунення невизначеності.

1. Panti, G. Multi-valued logics, in: D. Gabbay, P. Smets (eds.) Handbook of Defeasible Reasoning and Uncertainty Management Systems. vol. 1: P. Smets (ed.) Quantified Representation of Uncertainty and Imprecision. Kluwer Acad. Publ., Dordrecht. – 1998. – P. 25-74 . 2. Мейер Д. Теория реляционных баз данных: Пер. с англ. – М.: Мир, 1987. – 608 с., ил. 3. Huhtala Y., Karkainen J. Tane: An Efficient Algoritm for discovering Functional and Approximate Dependencies // The Computer Journal. 1999. – Vol. 42. – № 2.