від “навчання з учителем” – комбінування роботи ненавченого ігрового методу з навченим у попередньому інтервалі часу. Ефективність ігрового методу “навчання з учителем” у різних режимах функціонування мережі вимагає окремого дослідження.

1. Дэвис Д., Барбер Д., Прайс У., Соломонидес С. Вычислительные сети и сетевые протоколы. – М.: Мир,1982. 2. Богуславский Л.Б. Управление потоками данных в сетях ЭВМ. – М.: Наука, 1984. – 168 с. 3. Блэк Ю. Сети ЭВМ: протоколы, стандарты, интерфейсы. – М.: Мир, 1990. 4. Баранец С. Введение в сети с коммутацией пакетов // Компьютерное обозрение. – 1996. – № 15 (39). – С.15–18. 5. Найк Д. Стандарты и протоколы Интернета / Пер. с англ. – М: Издательский отдел “Русская Редакция” ТОО “Channel Trading Ltd.”, 1999. 6. Nelson Minar, Kwindla Hultman Kramer and Pattie Maes. Cooperating Mobile Agents for Dynamic Network Routing. In Alex Hayzelden, editor, Software Agents for Future Communications Systems, chapter 12. Springer–Verlag, 1999. 7. Allen B. MacKenzie and Stephen B. Wicker. Game Theory and the Design of Self–Configuring, Adaptive Wireless Networks. IEEE Communications Magazine. November 2001. PP. 126–131. 8. Назин А.В., Позняк А.С. Адаптивный выбор вариантов: Рекуррентные алгоритмы. – М., Наука, 1986.

УДК 681.3

Р.Б. Кравець, Ю.В. Скребець
Національний університет “Львівська політехніка”,
кафедра інформаційних систем та мереж

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПІДТРИМАННЯ ПРИЙНЯТТЯ РІШЕНЬ ДЛЯ СЛУЖБИ ШВИДКОЇ ДОПОМОГИ

Описується інтелектуальна система підтримання прийняття рішень для служби швидкої медичної допомоги (м. Львів). Система використовує базу правил, яка формується інтелектуальною компонентою. Компонента використовує модифікований метод побудови дерева рішень, та виведення на основі нього. Запропоновано опис та реалізацію нового способу зменшення кількості значущих правил у базі знань.

In this paper describe author decision support system (DSS) for ambulance service (Lviv). DSS uses a knowledge base (a base of some rules) that is with intelligent component built. The component make decision tree (new supposed technique) and uses it for prediction. Under consideration are new efficient technique for building a knowledge base and an achievement with small count of rules in base.

Постановка проблеми в загальному вигляді

Сьогодні проблеми швидкого, ефективного та зручного накопичення й збереження великої кількості даних значною мірою вирішені. Існує велика кількість інформаційних систем, які успішно працюють у багатьох галузях народного господарства. Значно складнішою задачею є реалізація систем аналітичного опрацювання інформації, тобто таких, які б допомагали користувачеві аналізувати нагромаджені дані, виявляти в них приховані, але надзвичайно корисні з практичної точки зору залежності.

Сучасний рівень розвитку апаратних та програмних засобів з деякого часу зробив можливим ведення баз даних оперативної інформації на всіх рівнях керування організацією. У процесі своєї діяльності промислові підприємства, корпорації, відомчі структури, органи державної влади накопичили значні обсяги даних. Вони містять великі потенційні можливості витягнення корисної аналітичної інформації, на основі якої можна виявляти приховані тенденції, будувати стратегії розвитку, знаходити нові рішення.

Описано інтелектуальну систему підтримання прийняття рішень, яка успішно експлуатується в приймальному відділенні лікарні швидкої допомоги м.Львова.

Базовою компонентою цієї системи є, звичайно, база даних. Проте ця база даних має виразні ознаки, які дозволяють зарахувати її до класу сховищ даних. База даних реалізована на основі швидкої промислової системи керування базами даних – Cache (v.5.0.1). Невід'ємною складовою системи є компонента отримання та опрацювання статистичних звітів, за допомогою якої можна отримувати різні агреговані характеристики даних, які містяться в базі даних.

Проте статистична звітність – далеко не вершина можливостей створеної системи підтримання прийняття рішень (далі СППР). Найцікавішими (та не менш корисними) її можливостями є отримання:

- звітів нерегламентованої структури про кількісні показники даних;
- правил-висновків на основі даних, які зберігаються в базі даних.

Отже, система має виразні ознаки OLAP- та Data Mining-систем. Актуальність такого підходу важко переоцінити. По-перше, на зміну великій кількості журналів обліку хворих, які потрапляють до лікарні, приходить спеціалізована програма для персонального комп'ютера, яка дозволяє зручним способом вносити нові, модифікувати та видаляти наявні записи в базі даних; по-друге, старша медична сестра, головний лікар чи завідувач приймального відділення в будь-який момент швидко можуть отримати звіт про поточний стан справ у відділенні: кількість хворих, які поступили, кількість госпіталізованих осіб, інформацію про них, про їх місцеперебування, встановлені діагнози тощо (цю інформацію швидко можуть отримати родичі та близькі пацієнтів від оператора); по-третє, керівний персонал має можливість оперативно отримувати результати нерегламентованих запитів, сформованих за допомогою OLAP-компоненти системи (без необхідності перегляду величезної кількості архівних матеріалів), а також використовувати Data Mining-компоненту для видобування корисних закономірностей, що можуть бути отримані на основі даних, накопичених за тривалий проміжок часу, які можна використовувати при вирішенні багатьох актуальних питань, пов'язаних із покращанням роботи лікарні загалом, підвищенням якості обслуговування пацієнтів, ефективним використанням трудових та матеріальних ресурсів та ін.

Тобто при введені системи в експлуатацію виводяться на якісно новий рівень оперативність пошуку інформації в архівах (виключно паперові носії), складанні добових, квартальних та річних звітів, а також надається додаткова можливість отримувати практично корисну інформацію про тенденції захворювань пацієнтів, які потрапляють до лікарні, їх вікові, соціальні та економічні ознаки поєднано з діагнозами госпіталізації та ін.

Аналіз останніх досліджень

У галузі інформаційних технологій завжди співіснували два класи систем [1]:

1) системи, орієнтовані на операційне (транзакційне) опрацювання даних; їх називають терміном OLTP (On-Line Transaction processing), на відміну від OLAP – аналітичного опрацювання даних;

2) системи, орієнтовані на аналітичне опрацювання даних – системи підтримання прийняття рішень (СППР), або Decision Support Systems (DSS).

На перших стадіях інформатизації завжди потрібно навести порядок саме в процесах повсякденного рутинного опрацювання даних, на що й орієнтовані традиційні системи опрацювання даних (СОД), тому випереджальний розвиток цього класу систем легко пояснити [1, 2].

Системи другого класу – СППР – вторинні порівняно з першими. Часто виникають ситуації, коли дані в організації накопичуються в кількох незв'язаних СОД, багато в чому дублюючи одні одних, але зовсім не узгоджені. У такому випадку достовірну комплексну інформацію отримати практично неможливо, незважаючи на її надлишок.

Метою побудови корпоративного сховища даних є інтеграція, актуалізація та узгодження оперативних даних з різнорідних джерел для формування єдиного та несуперечливого погляду на об'єкт керування загалом. При цьому в основу концепції сховищ даних покладено визнання

необхідності розділення наборів даних, які використовуються для транзакційного опрацювання, та наборів даних, які застосовуються для СППР [1,4].

Для аналітичного опрацювання інформації часто необхідно агрегувати великі масиви даних для подання інформації на високому концептуальному рівні. Такий вид аналітичного опрацювання інформації називається узагальненням даних.

Узагальнення даних – це процес абстрагування великих наборів відповідних даних в БД від низького концептуального рівня до високого та подання інформації на різних рівнях ієрархії класів даних. Відповідні методи узагальнення даних поділяють на дві категорії: атрибутно-орієнтована індукція та багатовимірний підхід [5].

Для вирішення задач аналітичного опрацювання даних з метою отримання корисної для прийняття важливих рішень інформації застосовуються технології Data Mining. Data Mining перекладається як “видобування” або “розкопка даних” [1]. Нерідко поруч з Data Mining зустрічаються слова “виявлення знань у базах даних” (knowledge discovery in databases, KDD) й “інтелектуальний аналіз даних”. Виникнення всіх зазначених термінів пов’язане з новим напрямком у розвитку засобів і методів опрацювання даних.

Основою сучасної технології Data Mining (discovery-driven data mining) є концепція шаблонів, що відбивають фрагменти багатоаспектних взаємозалежностей між даними. Ці шаблони являють собою закономірності, властиві підвибіркам даних, що можуть бути компактно виражені в зрозумілій для людини формі. Пошук шаблонів проводиться методами, не обмеженими апріорними припущеннями про структуру вибірки і вид розподілів значень аналізованих показників.

Фактично всі автори публікацій на теми, пов’язані з інтелектуальним аналізом даних, зазначають такі задачі Data Mining [1, 2, 5]:

- класифікація;
- кластеризація;
- виявлення послідовностей;
- виявлення асоціацій;
- прогнозування.

Інтелектуальні засоби аналізу даних використовують такі основні методи [1, 2, 5]:

- нейронні мережі;
- дерева рішень;
- індукція правил.

На базі цих методів розроблено доволі багато комбінованих методів інтелектуального аналізу даних.

У багатьох практичних задачах необхідно класифіккувати об’єкти. Класифікація передбачає наявність у даних певних класифікаційних ознак, що належать відповідні методи класифікації до методів навчання з вчителем. Такі ознаки мають назву атрибутів прийняття рішень, а таблиці з такими атрибутами називають таблицями прийняття рішень. Якщо результат класифікації відомий, то його значення задається значенням спеціального атрибуту прийняття рішень, а системи такого типу називаються системами прийняття рішень. До основних методів класифікації даних належать методи, основані на побудові дерев рішень, k-найближчих сусідів, міркування на основі попередніх випадків, наближених множин тощо. Іншим підходом до побудови систем прийняття рішень, що відрізняється від класифікації, є кластеризація або навчання без вчителя, при якому тренувальні кортежі не розбиті попередньо на класи, а класифікаційні ознаки цих класів невідомі і визначаються у процесі навчання.

У загальному випадку розв’язування задачі класифікації даних складається з двох етапів. На першому етапі будується модель, яка описує відому множину класів даних (кажуть про навчання системи). Така модель створюється аналізом кортежів бази даних про досліджувану предметну область. Для кожного кортежа встановлюється клас, до якого цей кортеж належить, і цей кортеж позначається певною міткою класу.

На другому етапі модель використовується для класифікації. Оцінюється точність передбачення моделлю. Якщо точність моделі вважається прийнятною, то вона може використовуватися для класифікування майбутніх даних, для яких атрибут прийняття рішення невідомий.

У результаті застосування методу дерев рішень до навчальної вибірки даних створюється ієрархічна структура класифікаційних правил типу "ЯКЩО... ТО...", що має вигляд дерева. Для того, щоб вирішити, до якого класу належить деякий об'єкт, алгоритм передбачає проходження дерева від кореня до одного з його листків. Процедура є ітераційною, і на кожній ітерації обирається нова гілка залежно від значення відповідного параметра об'єкта класифікації. Кожен кінцевий вузол (листок) містить вказівку про належність об'єкта до певного класу. Цей метод є наочним, і його результати легко інтерпретуються.

У [6] грунтовно описано етапи базового алгоритму ID3.

Складність дерева прийняття рішень (точніше, його висота й кількість вузлів) залежить від послідовності, у якій вибираються атрибути при формуванні нового вузла. Цілком очевидним є те, що працювати з менш складним деревом є значно простіше, ніж із складнішим. Тому паралельно виникає задача не просто побудувати дерево рішень, а побудувати оптимальне щодо кількості вершин та його висоти. Для цього треба визначити, який з атрибутів є кращим, щоб ним можна було позначити чергову кореневу вершину. Відповідь на це запитання дає алгоритм C4.5, який, по суті, є продовженням та вдосконаленням алгоритму ID3.

Найпростішим деревом можна вважати таке дерево, що правильно класифікує всі приклади з таблиці і при цьому оцінює те, які атрибути є важливішими при прийнятті рішення. Такі атрибути є більш інформативними і їх важливість оцінюється значенням певної функції якості процесу класифікації, який виконується деревом прийняття рішень.

Математичною основою для вимірювання інформаційного вмісту повідомлення є теорія інформації Шеннона. Приріст інформації, отриманий в результаті перевірки кореня кожного піддерева, дорівнює загальній інформації в піддереві мінус кількість інформації, необхідної для завершення класифікації після виконання перевірки. Використовуючи апарат теорії інформації, можна будувати оптимальні дерева.

На основі отриманого дерева рішень будується множина правил. Множина правил, отримана з оптимального дерева рішень, буде містити меншу кількість правил, а останні, в свою чергу, будуть коротшими. Відповідно меншу кількість ресурсів доведеться затрачати на класифікацію об'єктів.

Цілі статті

Фактично OLAP базується на багатомірному поданні даних. З його допомогою можна отримувати нерегламентованої структури звіти про кількісні (статистичні) показники даних, які є в базі або сховищі даних. Існує кілька підходів до фізичного зберігання даних, які підлягають аналітичному опрацюванню: формування багатовимірних структур (MOLAP), використання реляційного подання – ROLAP (Relational OLAP) тощо. OLAP-компонента розробленої СППР побудована на основі технології ROLAP. Перевага ROLAP над MOLAP полягає у значно вищій швидкості формування звітів за рахунок ненадлишковості даних.

Багатовимірні структури, що отримуються у процесі проектування, описують функції багатьох змінних, де значення полів таблиць бази даних є аргументами функцій, а числові показники – значення функцій. У загальному випадку для подання значень функцій при всіх наборах даних, які є в базі даних, використовується реляційна таблиця, що має розмірність (кількість полів) на одиницю більшу за кількість аргументів функції. Додатковий вимір використовується для зберігання агрегованих показників (значень функції). Тобто, коли маємо функцію $F(X_1, X_2, \ldots, X_n)$ від n змінних, її аргументи та значення задаються у вигляді $n+1$-вимірної таблиці, в рядках якої розміщено значення змінних $X_1, X_2, \ldots, X_{n-1}$, а останній рядок містить значення функції (фактично – це кількість кортежів, які описані відповідним рядком реляційної таблиці). Таку структуру називатимемо реляційним кубом даних.

У багатовимірному підході подання даних структури, у яких зберігаються значення функцій, прийнято називати гіперкубами даних, а змінні-аргументи – вимірами [3]. Дамо формальне означення поняттям гіперкуба даних та виміру. Нехай A_1, A_2,..., A_m – атрибути з універсуму U, визначені на доменах $dom(A_1)$, $dom(A_2)$,..., $dom(A_m)$ відповідно.

Виміром D називається підмножина декартового добутку $dom(A_1)\times dom(A_2)\times \times K \times dom(A_m)$. Значення виміру d називається кортеж виміру D. Кількість всіх кортежів виміру D називається обсягом виміру D і позначається як $|D|$.

Схемою виміру D називається сукупність атрибутів цього виміру $\mathbf{D} = \{A_1, A_2, \ldots, A_m\}$.

Нехай $D_1, D_2, \ldots, D_n$ – виміри зі схемами $\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_n$ відповідно.

Гіперкубом даних $H(D_1, D_2, \ldots, D_n)$ називається відображення $H : D_1 \times D_2 \times \ldots \times D_n \to \mathbf{V}$, де $\mathbf{V}$ – деяка множина, на якій задана функція агрегації Agg елементів множини. Координатою гіперкуба даних $H(D_1, D_2, \ldots, D_n)$ називається сукупність значень вимірів $(d_1, d_2, \ldots, d_n)$. Значенням гіперкуба даних $H(D_1, D_2, \ldots, D_n)$ в координаті $(d_1, d_2, \ldots, d_n)$ називається елемент $h \in \mathbf{V}$, у який відображається координата $(d_1, d_2, \ldots, d_n)$.

Отже, гіперкуб даних визначається як множина

$$H(D_1, D_2, \ldots, D_n) = \{(d_1, d_2, \ldots, d_n, h) \mid \forall i, 1 \le i \le n, d_i \in D_i, h \in \mathbf{V}, h = h(d_1, d_2, \ldots, d_n)\}.$$

Схемою гіперкуба даних H називається сукупність схем вимірів та множина значень гіперкуба даних $H = \{D_1, D_2, \ldots, D_n, V\}$.

Кількість вимірів, від яких залежить гіперкуб даних $H(D_1, D_2, \ldots, D_n)$, визначає вимірність гіперкуба даних. Отже, гіперкуб даних $H(D_1, D_2, \ldots, D_n)$ має вимірність n, або, інакше кажучи, є n-вимірним.

Отже, гіперкуб даних утворюється як багатовимірна структура, вимірами якої є інформаційні відношення. Атрибут гіперкуба, який описує показник, що аналізується, набуває значень на множині, на якій визначена функція агрегації її елементів. Наявність такої функції дає можливість агрегувати дані гіперкуба, що є необхідним для аналітичного опрацювання інформації [4].

Важливою задачею є формулювання операцій, які дозволяють оперувати кубами даних та їх частинами.

Наступним завданням системи є інтелектуальний аналіз даних, формалізація якого наводиться нижче.

Задача побудови дерева рішень формулюється так: значення атрибута прийняття рішень d визначає певний клас, до якого належить об'єкт з множини W. Множина значень атрибута d розбиває множину W на класи еквівалентності, кожний з яких задає прийняття рішення. Таблиці прийняття рішень поставимо у відповідність дерево прийняття рішень, яке у даному випадку є деревом, внутрішнім вершинам якого відповідають тести над заданою множиною об'єктів W та атрибутів A. Листки такого дерева відповідають класам еквівалентності, а дугам приписані значення атрибутів, які мають вершини дерева, з яких ці дуги виходять.

Наведемо означення дерева рішень [5]. Нехай B – деяка непорожня множина та F – деяка множина функцій, які визначені на A та такі, що набувають значення з множини $E_k = \{0,1,...,k-1\}$. Функції з F називатимемо перевірками, а пару $U = (B, F)$ – системою перевірок. Система U називається скінченною, якщо множина перевірок F є скінченною, та нескінченною у протилежному випадку. Задача над U – це набір вигляду $z = (v, f_1, ..., f_m)$, де $f_1, ..., f_m \in F$, $v : E_k^m \to \omega$ та $\omega = \{0,1,2,...\}$. Задача z полягає у визначенні за довільним елементом $b \in B$ значення $z(b) = v(f_1(b), ..., f_m(b))$. Дерево рішень над U – це кореневе дерево,

кожному листку якого відповідає число з ω – результат роботи дерева рішень, з кожною внутрішньою вершиною пов'язана перевірка з *F*, з кожною дугою пов'язане число з E_k – значення перевірки, при якій перехід здійснюється за цією дугою. Інцидентним кожній вершині дугам відповідають попарно різні значення.

Побудова оптимального дерева не є тривіальною задачею і потребує застосування спеціального математичного апарата.

Основний матеріал

Для ефективного аналізу даних, які збережені в кубі, потрібно визначити операції для їх агрегування. Ці операції повинні зменшувати кількість вимірів куба, певним чином агрегуючи показники (значення багатовимірних функцій).

Операція зрізу зводиться до фіксування значення одного з показників. Текст запиту мовою SQL, що реалізує операцію зрізу, наведено нижче:

SELECT $f_1, f_2, ..., f_{i-1}, f_{i+1}, ..., f_n, SUM(f_{n+1})$

FROM T

WHERE $f_i = f^*$

GROUP BY $f_1, f_2, ..., f_{i-1}, f_{i+1}, ..., f_n$,

де f_j – імена стовпців таблиці T бази даних, f^* – значення зафіксованого і-го поля.

Операцію зрізу можна поширити на будь-яку кількість полів, включивши їх значення до розділу WHERE директиви SELECT.

Операція проекції на множину полів реалізується включенням в результат запиту лише тих полів, які цікавлять аналітика. SQL-запит, який повертає відношення, що не містить поля f_i:

SELECT $f_1, f_2, ..., f_{i-1}, f_{i+1}, ..., f_n, SUM(f_{n+1})$

FROM T

GROUP BY $f_1, f_2, ..., f_{i-1}, f_{i+1}, ..., f_n$.

Можна застосовувати операцію згортки за відношенням. Складність її застосування полягає у необхідності додаткового відношення/відношень, яке містить правила агрегування значень виміру/вимірів. Операція не використовувалась в реалізації OLAP-компоненти СППР.

Застосовуючи послідовно до отримуваних наборів даних різні комбінації операцій зрізу та проекції, аналітик може отримати куб з двома (або трьома) вимірами, значення яких можна подати за допомогою графіків (для неперервних значень атрибутів) та діаграм (для дискретних).

При побудові дерева рішень як базовий використовується алгоритм ID3. Для отримання оптимального дерева використовується модифікація алгоритму.

Введемо поняття значущості залежності та правила як величини, що характеризує ступінь (кількісна величина) залежності між атрибутами.

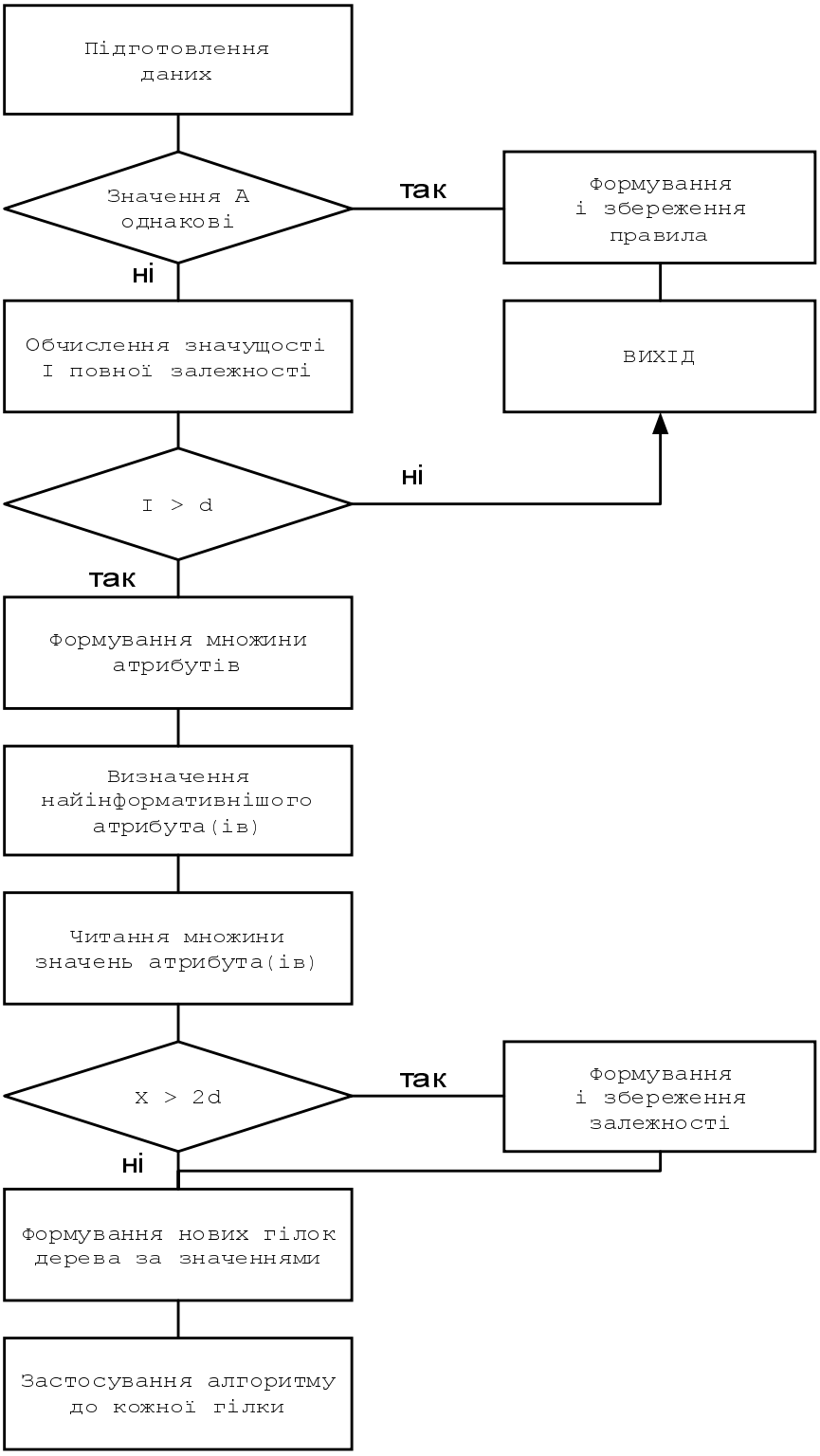
При побудові нової вершини дерева рішень визначається значення значущості (в конкретнму випадку – інформації) всіх атрибутів відносно атрибута прийняття рішення. Якщо це значення не перевищує певне порогове, то недоцільно будувати відповідну гілку дерева рішень. Значення інформації визначається за формулою:

$$I_r(X \to A) = n_r\left(\mathrm{H}_r(X) + \mathrm{H}_r(A) - \mathrm{H}_r\left(X \cup \{A\}\right)\right) ,$$

де X – множина умовних атрибутів, A – множина атрибутів (атрибута в конкретному випадку) прийняття рішення, n_r – загальна кількість кортежів у відношенні, H_r – ентропія.

Отже, визначення значущості залежності зводиться до обчислення ентропій інформаційного відношення r на підмножинах атрибутів X, A та $X \cup \{A\}$.

На наступному кроці вибирається той атрибут, значення ентропії для якого є максимальним і перевищує те ж порогове значення. Якщо ж значення ентропії є меншим за порогове, то обирається пара (трійка і т.д.) атрибутів.



Блок-схема алгоритму

Оцінка ентропії Шеннона, визначена на основі інформаційного відношення r, задається формулою:

$$\mathrm{H}_r(X) = -\sum_{x_i \in dom(X)} \frac{n_r(X = x_i)}{n_r} \ln \frac{n_r(X = x_i)}{n_r},$$

де $n_r(X = x_i)$ – кількість кортежів у відношенні r, які задовольняють умову $X = x_i$, n_r – загальна кількість кортежів у відношенні r.

При побудові вершин дерева рішень доцільно сформувати множину правил, причому до множини правил повинні входити лише значущі правила. Для визначення значущості правил в алгоритмах використана функція значущості χ_r^2, яка задається формулою:

$$\chi_r^2\left((X=x)\to(A=a)\right)=\frac{\left(n_r\left(X=x\wedge A=a\right)n_r-n_r\left(X=x\right)n_r\left(A=a\right)\right)^2 n_r}{n_r\left(X=x\right)n_r\left(A=a\right)\left(n_r-n_r\left(X=x\right)\right)\left(n_r-n_r\left(A=a\right)\right)}.$$

Отже, значущість правила визначається через величини $n_r\left(X=x\wedge A=a\right)$, $n_r\left(X=x\right)$ та $n_r\left(A=a\right)$, які можна отримати із таблиць кількостей кортежів, побудованих для підмножин атрибутів $X\cup\{A\}$, X та A відповідно.

Для опису стохастичної природи правил використаємо ступені підтримки та достовірності правил [5].

На рисунку наведено блок-схему модифікованого алгоритму ID3 для побудови дерев рішень.

Кажуть, що правило $(X=x)\to(A=a)$ має підтримку (support) *s*, якщо *s* % кортежів з n_r задовольняють умову множину $X=x\wedge A=a$, тобто

$$supp\left((X=x)\to(A=a)\right)=supp(X=x\wedge A=a)=\frac{n_r(X=x\wedge A=a)}{n_r}\cdot 100\%.$$

Достовірність (confidence) правила показує, яка імовірність того, що з умови $(X=x)$ випливає також умова $(A=a)$. Кажуть, що правило $(X=x)\to(A=a)$ справедливе з достовірністю *c*, якщо *c* % кортежів з n_r, що задовольняють умову $(X=x)$, задовольняють умову $(X=x\wedge A=a)$, тобто

$$conf\left((X=x)\to(A=a)\right)=\frac{supp(X=x\wedge A=a)}{supp(X=x)}\cdot 100\%.$$

Висновки

Система підтримання прийняття рішень реалізує всі вищевикладені теоретичні результати та використовує описані операції для кубів даних та алгоритм побудови дерева рішень та формування множини залежностей. Проблеми швидкого, ефективного та зручного накопичення і збереження великої кількості даних вирішені технологічно вже довший час, але жодна сучасна організація не може використовувати в своїй роботі на всю потужність жодне зі стандартних вирішень, які присутні на ринку. Існує велика кількість інформаційних систем, які успішно працюють у багатьох галузях народного господарства і створені для конкретного замовника.

Значно складнішою задачею порівняно із задачею реалізації інформаційної складової системи є реалізація системи аналітичного опрацювання даних, тобто такої, яка б допомагала користувачеві аналізувати нагромаджені дані, виконувати вибірки отриманих результатів у всіх можливих зрізах та проекціях всієї множини атрибутів, якими оперує система. Застосований багатовимірний підхід до аналітичного опрацювання даних виявився потужним інструментом аналітиків, а в термінах приймального відділення лікарні швидкої дпомоги – керівного персоналу. Отримані аналітичні звіти після детального аналізу можна використовувати як підгрунтя для прийняття важливих керівних рішень, які неможливо було б прийняти без застосування реалізованих в системі методів агрегування кількісних показників даних.

Проте неповноцінною можна було б вважати реалізовану систему підтримання прийняття рішень без інтелектуальної компоненти. Велику множину характеристик хворих, що надходять в приймальне відділення, отриману внаслідок функціонування системи тривалий період часу, можна використати не лише для формування аналітичних звітів, що містять агреговані показники, але й з метою виявлення в них прихованих, але надзвичайно корисних з практичної точки зору, залежностей. Фактично на основі даних зі створеного сховища можна отримати базу знань. Для

отримання бази знань (множини значущих правил) використовується в реалізованій системі ефективний метод дерев прийняття рішень. Отримані на основі нього правила мають структуру, що подібна до синтаксичних конструкцій природної мови, тому ці правила легко осмислити.

1. Щавльов Л.В. Способи аналітичного опрацювання даних для підтримання прийняття рішень. СУБД, 4–5/98. 2. http://www.olap.ru/basic/olap_intro6.asp, 10.09.2003 р. 3. Кравець Р.Б. Багатовимірна модель даних у системах аналітичної обробки інформації. – Вісник Національного університету "Львівська політехніка". – 1998. – №330. – С. 147–153. 4. Codd E.F., Codd S.B., Salley C.T. Providing OLAP (on-line analytical processing) to user-analysts: an IT mandate. – E.F. Codd & Associates, 1993. 5. Han J., Kamber M. Data mining: methods and technique. – Morgan Kaufman, 2000. 6. Нікольсь кий Ю., Щербина Ю., Якимечко Р. Дерева прийняття рішень та їхнє застосування для прогнозування діагнозу у медицині // Вісник Львівського університету імені Івана Франка. – 2003. – Вип. 6. – С. 195–196.

УДК 004.896

Р.Б. Кравець, Н.Б. Шаховська, А.Р. Продан
Національний університет "Львівська політехніка",
кафедра інформаційних систем та мереж

СИСТЕМА ВИБОРУ ОПТИМАЛЬНИХ ДАНИХ ДО РЕГУЛЮВАННЯ ПЕРЕТОКУ ЕЛЕКТРОЕНЕРГІЇ НА ЕКСПОРТ

Розглянуто проблеми вибору оптимального телевиміру ТЕС на основі методу МГУА.

Described main problems of optimel teletype choise based on MGCA.

Вступ

Людство невідворотно вступає в інформаційну епоху, де вага інформації постійно зростає. За підрахунками науковців, з початку нашої ери для подвоєння знань треба було 1750 років, друге подвоєння відбулося в 1900 році, а третє – до 1950 року, тобто вже за 50 років, при зростанні обсягу інформації за ці півстоліття у 8–10 разів. Причому ця тенденція усе підсилюється, тому що обсяг знань у світі до кінця XX століття зріс удвічі, а обсяг інформації збільшиться більше ніж у 30 разів. Це явище одержало назву "інформаційний вибух" та вказується серед ознак початку століття інформації.

Відсутність, неповнота або спотворення інформації призводять до викривлення дійсності, втрати управління, призупинення суспільного розвитку. Збирання, передавання, перетворення, обробка інформації та інші інформаційні процедури передують прийняттю будь-яких рішень, тому на сучасному етапі розвитку цивілізації велику увагу приділяють розробці прогресивних інформаційних технологій, підвищенню інформаційної культури суспільства.

Сьогодні є велика потреба саме в інформації для прийняття рішень при великій кількості даних. Коли кількість змінних є більшою ніж кількість випадків, найбільш відомі алгоритми обробки даних зіштовхуються з деякими проблемами. Навіть якщо дані чіткі, велика кількість змінних створює проблему розмірності. Тому прийняття рішення, базоване на аналізі даних – діалоговий та ітераційний процес різних частин робочих задач і рішень – називається отриманням знань з даних.

Є багато різних інструментів видобування даних, описаних у [3, 4]. Важливо обмежити залучення користувача до процесу отримання даних для складних систем при включенні відомого апріорного знання. Це зробить процес більш автоматизованим і об'єктивним. Більшість користувачів насамперед цікавить можливість отримання корисних та зразкових кінцевих результатів без застосування складних математичних, кібернетичних і статистичних методів чи без витрат часу на