

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
Навчально-науковий інститут економіки та менеджменту
Факультет економіки і підприємництва
Кафедра економічної кібернетики**

КОНСПЕКТ ЛЕКЦІЙ

з дисципліни «Інтелектуальні системи підтримки прийняття рішень в економіці та бізнесі»
за спеціальністю 051 «Економіка», спеціалізація «Економічна кібернетика»

Укладач(і):

доцент кафедри економічної кібернетики
к.е.н, доцент Кудрицька Ж.В.

доцент кафедри економічної кібернетики
к.е.н, доцент Овсяннікова Н.В.

Конспект лекцій розглянутий та схвалений
на засіданні кафедри

Економічної кібернетики

Протокол № 13 від «26» вересня 2016 р.

Завідувач кафедри _____ Т.І. Олешко

Лекція № 1

Тема лекції: ПРИЙНЯТТЯ РІШЕНЬ В ЕКОНОМІЦІ ТА БІЗНЕСІ

План лекції

1. Сутність процесу прийняття рішень. Аналіз ситуації прийняття рішення.
2. Вузькі місця та чинники, що визначають особливості процесу прийняття рішення
3. Інформаційні технології в економіці та управлінні. Структура інформаційної технології. Декомпозиція інформаційного процесу.
4. Інформаційна технологія автоматизації процесу аналізу інформації з використанням програмного забезпечення.
5. Принципи функціонування автоматичних засобів видобування знань.

Ключові слова

Управлінське рішення, альтернативи, умови процесу прийняття рішень, інформаційне забезпечення, інформаційні технології, принцип оптимальності, кластеризація, асоціація, регресія.

Література

1. Ситник В.Ф. Системи підтримки прийняття рішень: Навч.посібник. -К.:КНЕУ, 2004.-628 с.
2. Євланов Л.Г. Основи теорії прийняття рішень. / Євланов Л.Г. – М., 1979. – 210с.
3. Ларичев О.І. Теорія і методи прийняття рішень / Ларичев О.І. М., 2002. – 392с.
4. Рогоза М.Є. Системи підтримки і прийняття прийняття рішень/ Рогоза М.Є., Ємець О.О., Ємець Є.М. – Полтава, 2013. – 328с.
5. Галузинський Г. П. Перспективні технологічні засоби оброблення інформації : навч.-метод. посіб. для самост. вивчення дисципліни / Г. П. Галузинський, І. В. Гордієнко ; Київський національний економічний ун-т. - К. : КНЕУ, 2002. - 279 с.

Зміст лекції

1. Сутність процесу прийняття рішень. Аналіз ситуації прийняття рішення.

Управлінське рішення може розглядатися як організаційний акт, інтелектуальне завдання, як один з центральних складників процесу управління, найважливіша сполучна ланка між керуючою і керованою системами.

У процесі управління менеджер постає перед необхідністю вибору одного варіанта дій із декількох можливих. Результат такого вибору і є рішенням, отже, управлінське рішення — це альтернативне рішення.

Теорія і практика менеджменту пропонують класифікувати управлінські рішення за різними ознаками . В одних класифікаціях є тільки поділ за змістом завдань, що вирішуються (науково-технічні, технологічні, економічні, організаційні, соціальні, ідейно-

виховні). В окремих класифікаціях цей критерій називають "час дії", поділяючи рішення на стратегічні, тактичні й оперативні.

На кожному підприємстві, у кожній туристичній організації процес розробки і прийняття управлінських рішень має свої особливості. Вони визначаються специфікою діяльності організації, її структурою, організаційною культурою, досвідом керівників і управлінського персоналу тощо. Існує загальна технологія процесу прийняття рішень, яка передбачає сім етапів.

Діагностика проблеми найчастіше виявляє дві її категорії:

- проблему-ситуацію, коли поставлені цілі не вирішені;
- проблему — потенційну можливість.

Цей етап передбачає встановлення симптомів ускладнень; визначення новизни проблеми і ситуації; з'ясування причин і джерел виникнення проблеми; встановлення можливих взаємозв'язків з іншими проблемами; визначення повноти і достовірності інформації про проблему; встановлення можливості вирішення проблеми. Зрештою, визначення сутності проблеми дає змогу сформулювати конкретну ціль (цілі) прийняття рішення.

Другий етап характеризується визначенням обмежень для прийняття рішення (часових, фінансових, трудових, законодавчих, конкуренції тощо) і критеріїв для оцінки альтернатив (ефект від реалізації рішення, ступінь досяжності поставлених цілей, мінімізація витрат, максимізація прибутку, час виконання рішень тощо).

На третьому етапі визначаються альтернативи (варіанти рішення). Керівник, як правило, обмежує кількість варіантів вибору для розгляду лише декількома, які є найбільш прийнятними. При цьому великого значення набувають досвід та інтуїція керівника. У складних ситуаціях необхідно запрошувати експертів.

Четвертий етап призначений для аналізу й оцінювання обраних альтернативних варіантів рішення за встановленими критеріями і з урахуванням визначених раніше обмежень. При оцінюванні складних рішень керівник за допомогою експертів виявляє переваги та недоліки кожного з них, можливі загальні наслідки.

На п'ятому етапі на основі проведеного аналізу проблеми й оцінки альтернатив обирається остаточний варіант рішення (альтернатива з найбільш сприятливими наслідками).

На шостому стані реалізується (впроваджується) рішення. Завданням керівника на цій стадії є організація виконання рішення, яка включає складання плану реалізації рішення та доведення його, а також самого рішення до виконавців і тих, кого воно стосується. Менеджер повинен розподілити права і відповідальність, зацікавити людей у виконанні рішення, за необхідності провести роз'яснювальну роботу.

Завершальним етапом реалізації рішення є контроль. Головна мета цього стану — встановлення зворотного зв'язку, визначення причин відхилень, внесення (за необхідності) змін (коректив) у план реалізації рішення або в саме рішення.

Фактори, що впливають на процес прийняття рішень охоплюють:

- середовище прийняття рішення;
- особливості оцінки керівника, які визначаються його індивідуальною системою цінностей, що впливає на спосіб прийняття рішення та на його характер.

Прийнято розрізняти:

- врівноважені рішення приймають менеджери, які уважно і критично ставляться до власних дій, гіпотез, що висуваються, та їх перевірки. Зазвичай перед прийняттям рішення менеджери ретельно формулюють вихідну мсту і завдання;
- імпульсивні рішення приймають менеджери, які легко генерують ідеї, але не приділяють багато уваги їх перевірці, оцінці, тому вони виявляються недостатньо обґрунтованими і надійними;
- ризикові рішення відрізняються від імпульсивних тим, що їх автори не ретельно обґрунтовують власні гіпотези. Це потребує значної впевненості в собі, відсутності страху перед ризиком. Найчастіше такі рішення приймаються у надзвичайно критичній ситуації;
- інертні рішення є наслідком обережного пошуку, коли контрольні та уточнювальні дії переважають над генеруванням ідей, тому в таких рішеннях майже немає оригінальності або новаторства;
- обережні рішення приймаються після ретельної оцінки менеджером усіх варіантів.

Розглянуті види рішень приймаються здебільшого у процесі оперативного управління. Якість управлінського рішення є головним фактором ефективності системи менеджменту.

Умови, за яких досягається якісне управлінське рішення, такі:

- застосування до розробки управлінського рішення наукових підходів класичного менеджменту;
- урахування впливу економічних законів на ефективність управлінського рішення;
- своєчасне забезпечення особи, яка приймає рішення, достовірною інформацією;
- використання методів функціонально-системного аналізу, прогнозування, моделювання й економічного обґрунтування;
- структуризація проблеми й побудова дерева цілей;
- забезпечення багатоваріантності рішень (розробка щонайменше трьох організаційно-технічних варіантів виконання тієї самої функції з досягнення цілі);
- забезпечення можливості зіставляти варіанти рішення;
- правове обґрунтування рішення, що приймається;
- автоматизація процесів збирання й обробки інформації;
- формування і функціонування системи відповідальності та мотивації прийняття якісного й ефективного рішення;
- забезпечення механізмів реалізації рішення.

Забезпечення зіставності альтернативних варіантів управлінських рішень — обов'язкова умова при порівнянні варіантів — проводиться за такими критеріями: час (час здійснення проєктів або інвестицій), якість об'єкта, масштаб і місткість об'єкта, рівень освоєння технології, метод отримання інформації та виконання розрахунків, умови використання (експлуатації) об'єкта, рівень ризику і невизначеності.

У конкретних умовах альтернативні варіанти управлінського рішення можуть відрізнитися не за всіма факторами. Завдання спеціаліста, який приймає рішення, полягає у проведенні комплексного аналізу конкретних ситуацій з метою забезпечення зіставлення за максимальною кількістю факторів

2. Вузькі місця та чинники, що визначають особливості процесу прийняття рішення

1) В деяких завданнях самі варіанти проведення операцій являють собою невелику кількість альтернатив. Наприклад: будувати великий завод або будувати менше, а потім розширювати. Тому в теорії прийняття рішень (ТПР) постановка завдання визначається як пошук множини альтернатив, серед яких здійснюється пошук оптимального рішення.

2) Нерідко проблемна ситуація характеризується тим, що на етапі постановки завдання повний перелік альтернатив відсутній, наприклад альтернативи при грі в шахи. Тому первинним завданням прийняття рішень є генерація альтернатив. При грі в шахи доводиться генерувати альтернативи і оцінювати їх.

3) Оцінка цінності кожної альтернативи у багатьох випадках не зводиться до простому порівнянню. Цільова функція не є числовою і ТПР застосовує спеціальні методи вимірювання корисності альтернатив.

4) На практиці значна кількість завдань передбачає, що сама критеріальна величина не може бути єдиною, тобто критерій оцінки рішення являє собою не скаляр, а гілку. Тому навіть якщо за одним критерієм варіант є кращим, то за іншим він може виявитися гіршим, і тому їх порівняння не можливе.

5) У багатьох задачах автоматизації управління є нечіткі змінні, і нечіткі критерії.

6) Іноді зустрічаються задачі, в яких присутні кілька зацікавлених сторін (моделей, держав, фірм), які приймають рішення в одній і тій же системі, причому критерії цих сторін протилежні. Такі ситуації називаються конфліктними, при цьому принцип оптимальності є зовсім іншим, ніж у лінійному програмуванні. Цим займається теорія ігор.

7) У багатьох завданнях приймаються групові рішення, при цьому на основі експертизи виявляється бажання цієї групи, тобто дослідник відокремлюється від експерта.

В результаті обробки експертів, дослідник отримує критерій. Крім цього, прийняття рішення є багаторазовим повторюваним процесом.

Якщо певній задачі властиві усі сім проблем – вона нерозв'язна.

В теорії прийняття рішень Задачу прийняття рішення формалізовано як: вибір оптимального рішення з безлічи альтернатив.

В задачі прийняття рішення назвемо пару (Ω, P) ,

де Ω – множина варіантів (альтернатив),

P – принцип оптимальності.

Рішенням завдання є множина, яка утворюється у відповідності з принципом оптимальності P .

Відсутність хоча б одного з елементів (Ω, P) позбавляє завдання сенсу.

Математичним виразом принципу оптимальності P слугує функція вибору C_p , яка з усіма підмножинами являє його частину $C_p(x)$. Таким чином, рішенням вихідної задачі є $C_p(\Omega)$.

Задачі прийняття рішення розрізняються залежно від наявної інформації про множину Ω і принципі оптимальності P :

1) Загальна задача прийняття рішення: Ω, P – невідомі, необхідно отримати в процесі самого рішення.

2) Завдання з відомими Ω називається задачею вибору.

3) Завдання, в якій Ω, P – відомі називається спільним завданням оптимальності.

3. Інформаційні технології в економіці

Наш час – час розвитку технологій та час швидкості, коли необхідно приймати негайно рішення, реагувати на ситуації. Інформація відіграє при цьому важливу роль, адже саме вона дає підґрунтя для прийняття рішень та своєчасного реагування на ті чи інші події.

Інформаційні технології мають на меті забезпечують одержання інформаційного продукту для його аналізу людиною та прийняття на її основі рішень для виконання дій.

Інформаційний продукт (продукція) - документована інформація, яку підготовлено і призначено для задоволення потреб користувачів

Етапи розвитку:

1 - мова, усне відтворення інформації, передавання її за допомогою знаків, звуків - усе це можна назвати першим етапом розвитку інформаційних технологій (наскальних зображень, усної народної творчості, символів на предметах побуту).

2 - поява писемності є характерною ознакою другого етапу розвитку інформаційних технологій. Дерев'яні, вкриті воском або глиняні таблички, папірус, шкіра... перші сховища інформації – бібліотеки, книгодрукування.

3 – впровадження механічних засобів обробки, зберігання і передачі інформації, таких як друкарська машинка чи арифмометр.

4 – відкриття в галузі електрики – можливість передачі значних обсягів інформації на великі відстані з достатньо великою швидкістю (телефон, телетайп), зберігання їх на магнітних носіях.

5 – поява перших електронно-обчислювальних машин (ЕОМ) та переходом до електронно-цифрових технологій.

Можна відокремити такі етапи розвитку комп'ютерних інформаційних технологій:

- етап машинних ресурсів (впровадження ЕОМ, програмування в кодах машин);
- етап програмування (мови програмування, пакетна обробка);
- етап нових інформаційних технологій, який характеризується появою ПК - персональний комп'ютер, комп'ютерних мереж, АРМів (автоматизованих робочих місць), баз даних, OLAP-технології (динамічний аналіз даних), Інтернет-технологій, тощо.

Кожна інформаційна технологія орієнтована на обробку інформації певних видів, наприклад,

статистичної (електронні таблиці, СУБД),
текстової (текстові редактори),
графіки (графічні редактори).

Набори пакетів прикладних програм для математичних розрахунків і моделювання, експертні системи і бази знань використовуються в інформаційних системах для розв'язання формалізованих (чітко визначено алгоритм розв'язування) і неформалізованих задач.

Графічне і табличне подання даних часто застосовується як зручний інструмент економічного аналізу під час вивчення стану ринку (зокрема, за допомогою так званих графічних інформаційних систем), а також під час планування й прийняття рішень.

Гіпертекстові технології відкривають нові, якісно відмінні від традиційних, можливості засвоєння інформації. Вони передбачають переміщення від одних об'єктів інформації до інших з урахуванням їх змістової та семантичної взаємопов'язаності.

Інформаційна технологія автоматизації процесу аналізу інформації з використанням програмного забезпечення

Практика використання інформаційних технологій для моделювання та автоматизації підтримки прийняття рішень в управлінні соціально-економічними процесами тісно пов'язана із постійним розв'язанням задач аналізу значних обсягів інформації.

Особливої актуальності аналіз інформації набуває при потребі прийняття рішень для управління різними процесами.

Прикладом може служити аналіз значних обсягів економічної інформації.

Для його проведення можуть використовуватись методи, що ґрунтуються **на основі інструментальних засобів** та успішно застосовуються для складного аналізу статистичних даних шляхом прямого програмування інструментальними мовами. При цьому спочатку будуються економіко-математичні моделі, а потім – здійснюється їх комп'ютерна реалізація з використанням інструментальних засобів – мов програмування,

Інша група методів аналізу значних обсягів економічних даних ґрунтується на **використанні пакетів прикладного програмного забезпечення**.

Так, наприклад, для систематизованої обробки значних обсягів табличних даних на ПК під управлінням операційних систем сімейства Microsoft Windows використовуються процесори електронних таблиць (ЕТ) Microsoft Excel, спеціалізовані програмні пакети StatSoft Statistica, StatGraphics Plus, SPSS SigmaPlot та інші.

Зазначені програмні засоби мають потужні можливості обробки та аналізу даних, проте їх спільними недоліками є орієнтованість виключно на роботу під управлінням операційних систем сімейства MS Windows та порівняно висока вартість ліцензій на офіційне використання.

Для систематизованої обробки значних обсягів табличних даних на персональних комп'ютерах під управлінням операційних систем сімейства Linux використовуються наступні процесори електронних таблиць (ЕТ): OpenOffice.org Calc, ABS, Gnumeric, KSpread та ряд інших. Перевагами цих програмних засобів є відкрита ліцензія на використання і, відповідно, відсутність витрат на програмне забезпечення та широкі можливості.

Особливістю процесора ЕТ OpenOffice.org Calc є використання значної кількості вбудованих функцій, команд запису, створення і редагування макрозасобів; засобів для побудови і редагування графічних залежностей. Разом з цим, процесор ЕТ OpenOffice.org Calc підтримує імпорт, обробку і збереження даних у форматах файлів процесора ЕТ Microsoft Excel - файлах формату *.xls.

OpenOffice.org Calc обробляє дані з використанням вбудованих функцій різних категорій, включаючи статистичні та фінансові, які можна використовувати у формулах для складного аналізу даних. Засоби макропрограмування значно розширюють можливості обробки даних, автоматизують окремі процедури і полегшують роботу користувача. При цьому у програмі процесора ЕТ OpenOffice.org Calc підтримується версія мови програмування Basic, є вбудована можливість, окрім створення і застосування макросів, ще й написання власних підпрограм і модулів.

Програмний засіб Gnumeric має вбудовану значну кількість (понад 520) функцій, згрупованих у категорії: фінансові, дата/час, математичні, статистичні, інформаційні, бази даних, рядок, логічні, випадкові числа, теорія чисел, пошук, комплексні, операції з бітами, інженерні, Ерланг, Gnumeric. Всі вони можуть успішно використовуватися на практиці для автоматизації обчислень у електронних таблицях, в тому числі для аналізу інформації.

Практика використання розглянутих програмних засобів на ПК під управлінням операційних систем сімейства Linux показала їх високу ефективність, надійність і безпечність. Особливо потужними можливостями для аналізу даних шляхом використання вбудованих функцій та спеціальних інструментів аналізу даних володіє програмний засіб Gnumeric, який також є кросплатформним і може працювати як під управлінням операційних систем MS Windows, так і Linux.

Таким чином, інформаційна технологія на практиці може бути успішно реалізована на основі наведених програмних засобів для автоматизації процесів аналізу інформації.

5. Принципи функціонування автоматичних засобів видобування знань.

Для аналізу і розв'язання задач різного характеру, в тому числі і економічних, сучасні інформаційні технології пропонують широкий спектр засобів прийняття рішень – людино-машинні інтерактивні системи, які дозволяють особам, що приймають рішення, використовувати дані, знання, об'єктивні чи суб'єктивні моделі.

Вибір засобу для обробки інформації обумовлюється властивостями поставленої задачі.

Структуровані задачі містять кількісні та якісні змінні, підлягають формалізації, яка нескладно реалізується. Надалі для них можна розробити повністю структуровані процедури знаходження рішень.

Слабоструктуровані задачі містять як кількісні, так якісні змінні, для них можна частково розробити структуровані процедури знаходження рішень.

Неструктуровані задачі містять лише якісні описи, їх неможливо формалізувати, вимагають нестандартних процедур прийняття рішень, де використовуються досвід, кваліфікація та інтуїція людини.

Існує два способи отримання знань:

документальний (відомості містяться у різноманітних інформаційних джерелах (книги, документи, бази даних, інформаційні системи і т.п.).

і експертний припускає видобування і структуризацію знань з пам'яті людини – експерта, або фахівця в наочній області. Використовується для рішення неструктурованих задач.

Серед методів першої групи в економіці поширені методи математичної статистики, що вирішують спектр задач, проте не дозволяють знаходити і видобувати знання з масивів даних.

Серед другої групи поширені так звані експертні системи – спеціальні комп'ютерні програми, що моделюють процеси розмірковування та прийняття рішення людини.

Методи виявлення знань можна умовно розбити на п'ять груп:

1) класифікація;

2) кластеризація – групування об'єктів на основі даних, що описують сутність об'єкту. Об'єкти всередині кластера повинні бути "подібними" один на одного і відрізнятися від об'єктів, що увійшли до інших кластерів. Ступінь подібності об'єктів характеризує точність кластеризації. Для економічних задач використовують термін сегментація;

3) регресія, у тому числі і задача прогнозування. Це встановлення залежності вихідних змінних від вхідних. До цього ж типу задач відноситься і прогнозування часового ряду на основі хронологічних даних.

4) асоціація – виявлення закономірностей між пов'язаними подіями. Прикладом такої закономірності служить правило, яке вказує, що з події X впливає подія Y.

Такі правила називаються асоціативними. Вперше ця задача була розв'язана для знаходження типових шаблонів покупок в супермаркетах, тому іноді її ще називають аналізом споживчого кошика (market basket analysis).

5) послідовні шаблони – встановлення закономірностей між пов'язаними у часі подіями.

Розглянуті вище задачі знайшли широке застосування при видобуванні знань економічного характеру.

Розглянемо декілька наступних прикладів.

Класифікація використовується у випадку, коли класи об'єктів є наперед відомими. Наприклад, віднесення нового товару певної товарної групи (продовольчі, промислові), віднесення клієнта до визначеної категорії (постійний клієнт, новачок). При кредитуванні це може бути, наприклад, віднесення клієнта за певними ознаками до однієї з груп ризику.

Кластеризація може використовуватися для сегментації і побудови профілів клієнтів (покупців). При достатньо великій кількості клієнтів неможливо розробити для кожного індивідуальний підхід. Тому клієнтів зручно об'єднати в групи – сегменти з однорідними ознаками (групами ознак). Це можуть бути сегменти по сфері діяльності, по географічному розташуванню. Після сегментації можна отримати відомості, які саме сегменти є найактивнішими, які приносять найбільший прибуток, виділити характерні для них ознаки. Ефективність роботи з клієнтами підвищується за рахунок обліку їх персональних переваг.

Регресія використовується для встановлення залежності в чинниках. Наприклад, в задачі прогнозування залежною величиною є обсяги продажів, а чинниками, що впливають на цю величину, можуть бути попередні обсяги продажів, зміна курсу валют, активність конкурентів і т.д. Або, наприклад, при кредитуванні фізичних осіб вірогідність повернення кредиту залежить від особистих характеристик людини, сфери його діяльності, наявності майна, платоспроможності, тощо.

Асоціації допомагають виявляти товари, які люди купують одночасно. Це може бути корисно для більш зручного розміщення товару на прилавках, стимулювання продажів (наприклад: розміщення гірчиці чи кетчупу біля сосисок, чаю біля печива гарантує збільшення сукупних обсягів продажу цих продуктів).

Послідовні шаблони можуть використовуватись при плануванні продажів або наданні послуг. Наприклад: якщо людина придбала фотоплівку, то через деякий час віддасть її на проявлення і замовить друк фотографій.

Застосовуючи індуктивні методи до множини вхідних даних можна виявити нелінійні закономірності та видобути певні знання. Проте, незалежно від методу, їх якість та важливість насамперед залежить від якості, змістовності та повноти даних, що будуть проаналізовані.

Лекція № 2

Тема лекції : ОСНОВНІ ПОНЯТТЯ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ. ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ТА ЇХ ЗАСТОСУВАННЯ

План лекції

1. Зміст задачі СППР.
2. Поняття інтелектуальних систем та класифікація.
3. Бази даних як основа СППР.
4. Неєфективність використання OLTP-систем для аналізу даних.
5. Концепція сховища даних. Створення сховищ даних. Технології OLAP та Data Mining

Ключові слова

Сховище даних, прогнозування, оперативна база даних, аналіз даних, OLAP-технології, клієнт, сервер ресурсу, гіпотеза, чинник, експерт.

Література

1. Ситник В.Ф. Системи підтримки прийняття рішень: Навч. посібник. -К.:КНЕУ, 2004.-628 с.
2. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP / А. А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод. - 2-е изд., перераб. и доп. - СПб.:БХВ-Петербург, 2007. - 384 с.
3. Саати Т. Принятие решений. Метод анализа иерархий. // Пер. с англ. - М.: "Радио и связь", 1993. - 320 с.
4. Джексон П. Введение в экспертные системы: Пер. с англ.: Уч. пос. -М.: Издательский дом «Вильямс», 2001. - 624 с.

Зміст лекції

1. Зміст задачі систем підтримки прийняття рішень.

Кількість інформації у світі з кожним роком зростає. З великої кількості інформації дуже мала її частина буде коли-небудь побачена людським оком. Сьогодні можна без перебільшення сказати, що настала етап усвідомлення процесів, пов'язаних з інформацією. Обчислювальна техніка створювалась насамперед для обробки даних. В наш час обчислювальні машини та комп'ютерні мережі дозволяють накопичувати великі масиви даних для рішення задач обробки та аналізу. На жаль, сама по собі машинна форма надання даних *містить інформацію, необхідну для людини, с прихованому вигляді*, і для того, щоб її дістати необхідно використовувати *спеціальні методи аналізу даних*.

Великий обсяг інформації, з одного боку, дозволяє отримати більш точні розрахунки та аналіз, а з іншої – перетворює пошук рішень у складну задачу. Первинний аналіз даних був перекладений на комп'ютер. В результаті з'явився цілий клас програмних систем, які покликані полегшити роботу людей, що виконують аналіз (аналітиків). Такі системи прийнято називати *система підтримки рішень – СППР (DSS, Decision Support System)*.

Системи підтримки прийняття рішень почали розвиватися на ранніх стадіях ери розподіленого обчислення. Історія таких систем веде відлік приблизно з 1967 року і за цей час СППР пройшли значний шлях розвитку, включаючи орієнтовані на моделі СППР, інструментальні засоби запиту та звітування, системи бізнесової інформації

(Business Intelligence), оперативне аналітичне оброблення (OLAP), групові СППР та виконавчі інформаційні системи.

Історія систем підтримки прийняття рішень (див.табл.2.1) є відносно короткою, а концепції та технології все ще інтенсивно розвиваються.

Табл.2.1

Розвиток концепцій СППР у 60-90 рр. XX ст.			
Роки XX століття ЕВОЛЮЦІЯ КОНЦЕПЦІЙ СППР			
60-і	70-і	80-і	90-і
Структуровані звіти	BrandAid (Класи допоміжних засобів)	Ключові книги	Сховища і вітрини даних
Інтерактивні (діалогові) системи дослідження	MDS (Системи управлінських рішень)	GDSS (Групові СППР)	OLAP-системи
Розвиток теорії	RDBMS (Реляційні СКБД)	Виконавчі інформаційні системи (EIS) Експертні системи	Data mining (Добування даних)

Системи підтримки прийняття рішень (СППР) виникли завдяки розвитку управлінських інформаційних систем і являють собою системи, розроблені для підтримки процесів прийняття рішень менеджерами в складних і слабоструктурованих ситуаціях, зв'язаних з розробкою і прийняттям рішень.

Досі немає єдиного визначення СППР. Наприклад, деякі автори під СППР розуміють "інтерактивну прикладну систему, яка забезпечує кінцевим користувачам, які приймають рішення, легкий і зручний доступ до даних і моделей з метою прийняття рішень в напівструктурованих і неструктурованих ситуаціях з різних галузей людської діяльності".

Існує твердження, згідно з яким СППР являє собою специфічний і добре описуваний клас систем на базі персональних комп'ютерів.

Для виконання аналізу СППР повинна накопичувати інформацію, володіючи засобами для її введення та зберігання. Виділяють 3 основні задачі, що вирішують в СППР:

введення даних; зберігання даних; аналіз даних.

Основне завдання СППР – надати аналітикам інструмент для виконання аналізу даних. Для ефективного використання СППР користувач-аналітик має володіти відповідною класифікацією. Система не генерує вірні рішення, а тільки надає аналітику дані у відповідному вигляді (звіти, таблиці, графіки і т.п.) для вивчення та аналізу. Саме тому такі системи забезпечують виконання функції підтримки рішень.

Якість прийнятих рішень залежить від кваліфікації аналітика.

Постійне зростання обсягів даних, що аналізуються, висока швидкість обробки та аналізу, а також складність використання машинної форми надання даних стимулює дослідження та розробку інтелектуальних СППР. Для таких СППР характерно наявність функції, яка реалізує окремі розумові можливості людини.

За ступенем «інтелектуальності» обробки даних при аналізі виділяють 3 класи задач аналізу:

- 1) *інформаційно-пошуковий* – СППР здійснює пошук необхідних даних, виконуються раніше визначені запити;
- 2) *оперативно-аналітичний* – СППР здійснює групування і обробку даних в будь-якому вигляді, необхідному аналітику. Тут неможливо наперед передбачити необхідні аналітику запити;

3) *інтелектуальний* – СППР здійснює пошук функціональних і логічних закономірностей в накопичених даних, побудову моделей і правил, які пояснюють знайдені закономірності і/або прогнозують розвиток деяких процесів (з певною ймовірністю).

Таким чином, узагальнена архітектура СППР може бути надана наступним чином (див.рис.2.1.)

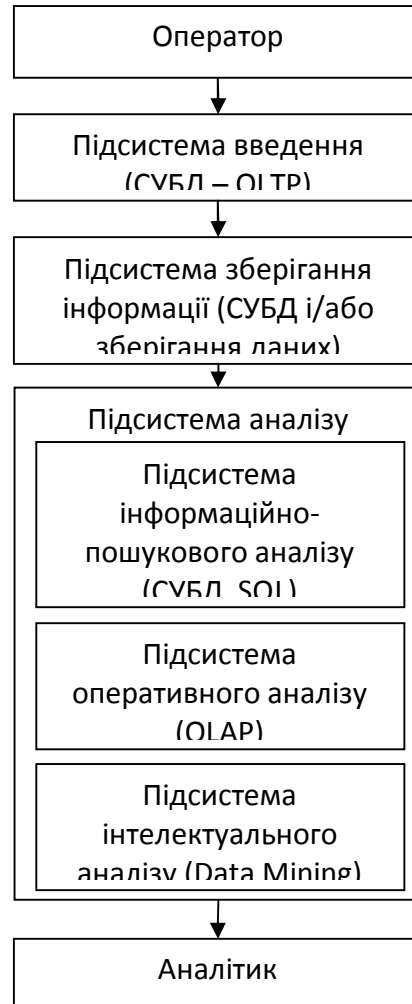


Рис.2.1. Узагальнена архітектура системи підтримки прийняття рішень

2. Поняття інтелектуальних систем та види.

Інтелектуальна система (ІС, [англ.](#) *intelligent system*) – це технічна або програмна система, яка може вирішувати задачі, що традиційно вважаються творчими та належать до конкретної предметної області, знання про яку зберігаються в пам'яті такої системи.

Структура інтелектуальної системи включає 3 блока:
базу знань, вирішувач та інтелектуальний інтерфейс.

ІС вивчаються групою наук, що об'єднуються під назвою *штучний інтелект*.

Види ІС:

- 1) Інтелектуальна інформаційна система
- 2) Експертна система
- 3) Розрахунково-логічна система
- 4) Гібридна інтелектуальна система
- 5) Рефлекторна інтелектуальна система

1) **Інтелектуальна інформаційна система (ІС)** — це один з видів автоматизованих інформаційних систем, інколи ІС називають системою, засновану на знаннях. ІС є комплексом програмних, лінгвістичних і логіко-математичних засобів для реалізації основного завдання: здійснення підтримки діяльності людини і пошуку інформації в режимі розширеного діалогу на природній мові.

Класифікація завдань, вирішуваних ІС:

- 1) інтерпретація даних – процес визначення змісту даних, результати якого мають бути погодженими і коректними. Зазвичай передбачається багатоваріантний аналіз даних.
- 2) діагностика – процес співвідношення об'єкта з деяким класом об'єктів і виявлення несправності в деякій системі. Несправність — це відхилення від норми.
- 3) моніторинг – безперервна інтерпретація даних у реальному масштабі часу і сигналізація про вихід тих або інших параметрів за допустимі межі.
- 4) проектування. Проектування полягає в підготовці специфікацій на створення «об'єктів» із заздалегідь визначеними властивостями. Під специфікацією розуміється весь набір необхідних документів — креслення, записка пояснення і так далі.
- 5) прогнозування – дозволяє передбачати наслідки деяких подій або явищ на підставі аналізу наявних даних. Прогнозуючі системи логічно виводять ймовірні наслідки із заданих ситуацій. У прогнозуючій системі зазвичай використовується параметрична динамічна модель, в якій значення параметрів «підганяються» під задану ситуацію. Висновки, що виводяться з цієї моделі, складають основу для прогнозів з ймовірними оцінками.
- 6) планування – знаходження планів дій, що відносяться до об'єктів, здатних виконувати деякі функції. У таких ІС використовуються моделі поведінки реальних об'єктів з тим, аби логічно вивести наслідки планованої діяльності.
- 7) навчання – використання комп'ютера для навчання деякої дисципліни або предмету. Системи навчання діагностують помилки при вивченні якої-небудь дисципліни за допомогою ЕОМ і підказують правильні рішення. Вони акумулюють знання про гіпотетичного «учня» і його характерні помилки, потім у роботі вони здатні діагностувати слабкості в знаннях учнів і знаходити відповідні засоби для їхньої ліквідації. Крім того, вони планують акт спілкування з учнем залежно від успіхів учня з метою передачі знань.
- 8) керування – функція організованої системи, що підтримує певний режим діяльності. Такого роду ІС здійснюють управління поведінкою складних систем відповідно до заданих специфікацій.
- 9) підтримка прийняття рішень. Підтримка прийняття рішень — це сукупність процедур, що забезпечує особу, що приймає рішення, необхідною інформацією і рекомендаціями, що полегшують процес ухвалення рішення. Ці ІС допомагають фахівцям вибрати і сформулювати потрібну альтернативу серед безлічі виборів при ухваленні відповідальних рішень.

У загальному випадку всі системи, засновані на знаннях, можна підрозділити на

1) системи, що вирішують завдання <u>аналізу</u>	Основна відмінність завдань аналізу від завдань синтезу полягає в тому, що в завданнях аналізу <u>безліч рішень може бути перераховане і включене в систему</u> ,	завданнями аналізу є: <u>інтерпретація</u> даних, діагностика, підтримка ухвалення рішення	Комбіновані завдання:
2) системи, які вирішують завдання <u>синтезу</u>	в завданнях синтезу <u>безліч рішень потенційно не обмежена і будується з вирішень компонент або проблем</u> .	завданнями синтезу є: <u>проектування</u> , <u>планування</u> <u>управління</u> .	<u>вчення</u> , <u>моніторинг</u> , <u>прогнозування</u> .

Типова схема функціонування інтелектуальної системи

Функціонування інтелектуальної системи – постійне прийняття рішень на основі аналізу поточних ситуацій для досягнення певної мети.

Етапи, які утворюють типову схему функціонування інтелектуальної системи:

1. Безпосереднє сприйняття зовнішньої ситуації; результатом є формування первинного опису ситуації.
2. Зіставлення первинного опису зі знаннями системи і поповнення цього опису; результатом є формування вторинного опису ситуації в термінах знань системи.
3. Планування цілеспрямованих дій та прийняття рішень, тобто аналіз можливих дій та їхніх наслідків і вибір тієї дії, яка найкраще узгоджується з метою системи. Це рішення, взагалі кажучи, формулюється деякою внутрішньою мовою (свідомо або підсвідомо).
4. Зворотна інтерпретація прийнятого рішення, тобто формування робочого алгоритму для здійснення реакції системи.
5. Реалізація реакції системи; наслідком є зміна зовнішньої ситуації і внутрішнього стану системи, і т. д.

2) Експертна система — це:

- методологія адаптації алгоритму успішних рішень одної сфери науково-практичної діяльності в іншу;
- (з поширенням комп'ютерних технологій) це тотожна (подібна, заснована на оптимізуючому алгоритмі) інтелектуальна комп'ютерна програма, що містить знання та аналітичні здібності одного або кількох експертів щодо деякої галузі застосування і здатна робити логічні висновки на основі цих знань, тим самим забезпечуючи вирішення специфічних завдань (консультування, навчання, діагностика, тестування, проектування тощо) без присутності експерта (спеціаліста в конкретній проблемній галузі);
- система, яка використовує базу знань для вирішення завдань (видачі рекомендацій) у деякій предметній галузі.

Структура експертних систем:

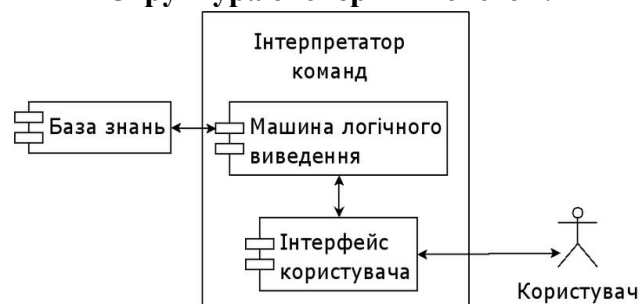


Рис. Структура типової експертної системи.

Відомі експертні системи:

- [CLIPS](#) — [мова програмування](#), використовується для створення експертних систем
- [Dendral](#) — аналіз даних [мас-спектрометрії](#)
- [Dipmeter Advisor](#) — аналіз даних, отриманих під час пошуку [нафти](#)
- [Jess](#) — від [англ. Java Expert System Shell](#), оболонка експертних систем на [Java](#). Рушій CLIPS реалізований на мові програмування Java, використовується для створення експертних систем
- [MQL 4](#) — MetaQuotes Language 4, спеціалізована мова програмування для опису фінансової стратегії
- [Mycin](#) — діагностика інфекційних хвороб крові та рекомендація антибіотиків
- [Prolog](#) — мова програмування, використовується для створення експертних систем
- [RI / XCON](#)(експертна система) — обробка замовлень
- [SHINE Real-time Expert System](#) — від [англ. Spacecraft Health INference Engine](#), рушій для отримання даних про стан і безпеку космічного корабля
- [STD Wizard](#) — експертна система для рекомендації та вибору медичних аналізів (діагностики)

Переваги експертних систем

над людьми-експертами при використанні:

- переважає можливості людини при вирішенні надзвичайно громіздких проблем;
- не має упереджених думок, тоді як експерт може користуватися побічними знаннями і легко піддається впливу зовнішніх факторів;
- не робить поспішних висновків, нехтуючи певними етапами знайдення рішення;
- забезпечує діалоговий режим роботи;
- дозволяє роботу з інформацією, що містить символічні змінні;
- забезпечує коректну роботу з інформацією, яка містить помилки, за рахунок використання ймовірнісних методів досліджень;
- дозволяє проводити одночасну обробку альтернативних версій;
- за вимогою пояснює хід кроків реалізації програми;
- забезпечує можливість обґрунтування рішення та відтворення шляху його прийняття.

Сфера застосування та перспективи розвитку

Експертні системи досить давно використовуються у діагностиці, зокрема у медичній та автомобільній.

Також експертні системи можна використовувати в [прогнозуванні](#), [плануванні](#), [контролі](#), [управлінні](#) та [навчанні](#). Наприклад, експертні системи вже застосовуються в банківській справі в таких напрямках:

- програмах аналізу інвестиційних проектів;
- програмах аналізу стану валютного, грошового та фондового ринку;
- програмах аналізу кредитоспроможності чи фінансового стану підприємств і банків.

3) Розрахунково-логічні системи – системи, які можуть *вирішувати управлінські і проектні завдання* по декларативним описам умов.

Дедуктивним (від латинського слова deductio — виведення) називається *умовивід*, у якому висновок про окремий предмет класу робиться на підставі класу в цілому.

Наприклад: Усі договори є угоди. Петренко уклав договір. Отже, Петренко уклав угоду. Тут висновок про те, що Петренко уклав угоду, зроблено на підставі загального положення про те, що «Всі договори є угоди».

У дедуктивному умовиводі думка рухається від загального до окремого, одиничного, часткового.

Семантична мережа — інформаційна модель предметної області, що має вигляд орієнтованого графа, вершини якого відповідають об'єктам предметної області (поняття, події, властивості, процеси), а ребра (дуги (причому направлені)) задають відносини між ними.

Є одним із способів представлення знань – семантика предметної області у вигляді понять і відносин між ними.

У комп'ютері вершинам, або вузлам, графа відповідають групи комірок пам'яті, а зв'язкам — вказівки, що містять коди адрес пам'яті, завдяки чому програма знаходить потрібні комірки.

Найважливіші зв'язки — типу «Це є»: вони дозволяють побудувати в мережі ієрархію понять, в якій вузли нижчих рівнів успадковують властивості вузлів вищих рівнів. Таким механізмом перенесення зумовлена ефективність семантичних мереж.

4) Гібридна інтелектуальна система (ГІС) – система, в якій для вирішення завдання використовується більш за один метод імітації інтелектуальної діяльності людини. Таким чином ГІС – це сукупність:

- аналітичних моделей
- експертних систем
- штучних нейронних мереж
- нечітких систем
- генетичних алгоритмів
- імітаційних статистичних моделей

ГІС – інструмент синергетичного штучного інтелекту, який призначений для *моделювання ефектів взаємодії, самоорганізації, адаптації, які спостерігаються в системах, де тісно переплітаються природа, людина і техніка.* Можливо, моделі і методи ГІС стануть релевантними складності задач, які вирішуються в системах підтримки прийняття рішень.

5) Рефлекторна система — це система, яка формує відповідні реакції на різні комбінації вхідних дій. Відповідні комбінації виробляються спеціальними алгоритмами, що забезпечують вибір найбільш ймовірних реакцій інтелектуальної системи на множину вхідних дій при відомих ймовірностях вибору реакції. Ця задача подібна до тієї, яку реалізують *нейромережі*.

При комбінації дій на рецептори формуються числові характеристики рефлекторів через проміжний шар (слої). Зв'язки між шарами (слоями) забезпечують передачу деякої величини – імпульсу від елементів одного шару (слою) до іншого. Якщо сумарний імпульс на вході деякого елемента перевищує його порогове значення, то він передає своє значення на елементи іншого шару. Таким чином кожен елемент є моделлю нейрона.

Рефлекторний алгоритм напряму розраховує адекватну вхідним діям реакцію інтелектуальної системи.

Штучні нейронні мережі (ШНМ) — математичні моделі, а також їхня програмна та апаратна реалізація, побудовані за принципом функціонування біологічних нейронних мереж — мереж нервових клітин живого організму.

Системи, архітектура і принцип дії базується на аналогії з мозком живих істот. Ключовим елементом цих систем виступає штучний нейрон як імітаційна модель нервової клітини мозку — біологічного нейрона.

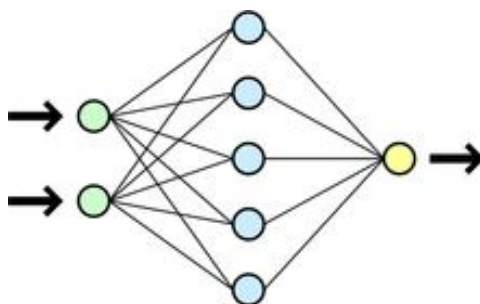


Рис. Схема простої нейронної мережі.
Зеленим кольором позначенні *вхідні* нейрони, голубим *приховані* нейрони, жовтим — *вихідний* нейрон

ШНМ представляють собою систему з'єднаних і взаємодіючих між собою простих процесорів (штучних нейронів). Такі процесори зазвичай достатньо прості, особливо в порівнянні з процесорами, що використовуються в персональних комп'ютерах.

Кожен процесор схожої мережі має справу тільки з сигналами, які він періодично отримує, і сигналами, які він періодично посилає іншим процесорам. І тим не менш, будучи з'єднаними в досить велику мережу з керованою взаємодією, такі локально прості процесори разом здатні виконувати достатньо складні завдання.

Нейронні мережі не програмуються в звичайному розумінні цього слова, вони навчаються. Можливість навчання — одна з головних переваг нейронних мереж перед традиційними алгоритмами.

Технічно навчання полягає в знаходженні коефіцієнтів зв'язків між нейронами. У процесі навчання нейронна мережа здатна виявляти складні залежності між вхідними даними і вихідними, а також виконувати узагальнення. Це означає, що у разі успішного навчання мережа зможе повернути вірний результат на підставі даних, які були відсутні в навчальній вибірці, а також неповних та / або «зашумлених», частково перевернутих даних.

Штучний нейрон – *Перцептрон*, или *персептрон* (англ. *perceptron* от лат. *perceptio* — восприятие; нем. *perzeptron*) — математична і комп'ютерна модель сприйняття інформації мозком (кібернетична модель мозку) – схема устроювання, яке моделює процес людського сприйняття.

Він здатен навчатись і вирішувати доволі складні задачі. Основна математична задача, з якою він справляється – лінійний розподіл будь-яких нелінійних множин.

Рефлекторні системи *застосовуються для вирішення задач:*

оцінки інвестиційних пропозицій,

оцінки та прогнозування впливів негативних факторів на здоров'я населення,

прогнозування результатів та спортивних ігор.

3. Бази даних як основи СППР.

База даних являє собою модель деякої предметної області, яка складається з пов'язаних між собою даних про об'єкти, їх властивості і характеристики.

Засоби для роботи з БД являють собою СУБД. СУБД є інструментом для розробки прикладних програм, що використовують БД.

Для того, щоб зберігати дані згідно якої-небудь моделі предметної області, структура БД повинна максимально відповідати цій моделі. Першою структурою, що використовувалась в СУБД, була *ієрархічна структура* (60-ті роки минулого століття). Вона передбачала зберігання даних у вигляді *дерева*. Це значно спрощувало створення та підтримку таких БД. Однак неможливість надати багато об'єктів реального світу у вигляді ієрархії призвела до використання таких БД у спеціалізованих областях.

Наступна спроба покращити ієрархічну структуру створення *мережевої структури БД*. Але робота з мережевими БД була настільки складним процесом, що така структура не знайшла широкого застосування.

Найбільш поширені в наш час – *реляційні БД* (від лат. – відношення). Їх структура побудована на взаємовідношенні складових її частин.

Особливості:

- 1) дані надаються у вигляді таблиць
- 2) дані доступні логічно
- 3) БД включає в себе метадані (опис таблиць: назва, тип, розмір колонок; індекси, процедури тощо)
- 4) Використовується єдина мова для взаємодії (в наш час SQL)
- 5) СУБД повинна забезпечувати альтернативний вигляд відображення даних (будувати віртуальні таблиці, поєднувати тощо)
- 6) Повинні підтримувати операції реляційної алгебри
- 7) Повинна забезпечувати розподіл операцій – користувач повинен мати можливість пов'язувати дані, що знаходяться в таблицях на різних вузлах комп'ютерної мережі.
- 8) Повинна забезпечувати можливість одночасно працювати декільком користувачам з декількох комп'ютерів, гарантуючи при цьому цілісність даних. Для виконання цих правил в СУБД використовується механізм управління транзакціями. *Транзакція* – послідовність операцій над БД, що розглядаються СУБД як єдине ціле. Транзакція переводить БД із одного цілісного стану в інший.

4. Сховище даних.

Прагнення об'єднати в одній архітектурі СППР можливості OLTP-підсистем та систем аналізу призвело до появи концепції сховищ даних.

Сховище даних (*англ.* data warehouse) — предметно орієнтований, інтегрований, незмінний набір даних, що підтримує хронологію і здатний бути комплексним джерелом достовірної інформації для оперативного аналізу та прийняття рішень.

В основі концепції сховища даних (СД) лежить розподіл інформації, що використовують в системах оперативної обробки даних (**OLTP**) і в системах підтримки прийняття рішень (**СППР**). Такий розподіл дозволяє оптимізувати як структури даних оперативного зберігання для виконання операцій введення, модифікації, знищення та пошуку, так і структури даних, що використовуються для аналізу. В СППР ці два типи даних називаються відповідно *оперативними джерелами даних (ОДД) та сховищем даних*.

Структура сховища даних та оптимізація його обсягів

Методи інтелектуального аналізу інформації часто розглядають як природний розвиток концепції сховищ даних.

Головна *відмінність* сховища від бази даних полягає в тому, що їх створення і експлуатація переслідують різну мету.

База даних відіграє роль помічника в оперативному управлінні організацією. Це щоденні задачі отримання актуальної інформації: бухгалтерські звітності, облік договорів, тощо.

Сховище даних накопичує всі необхідні дані для здійснення задач стратегічного управління в середньостроковому і довгостроковому періоді. Наприклад, продаж товару і генерація рахунку проводяться з використанням бази даних, а аналіз динаміки продажів за декілька років, що дозволяє спланувати роботу з постачальниками – за допомогою сховища даних.

Сховище даних (Data Warehouse) – це систематизована інформація з різномірних джерел, яка є необхідною для обробки з метою ухвалення стратегічно важливих рішень

Сховище будується на основі клієнт-серверної архітектури, СУБД і утиліт підтримки прийняття рішень. Дані, що надходять у сховище, стають доступні тільки для читання.

Властивості сховища даних:

о предметна орієнтація (інформацію організовано відповідно до основних аспектів діяльності);

о інтегрованість даних (дані в сховище надходять з різних джерел і відповідно агрегуються);

о стабільність, інваріантність у часі (записи в DW ніколи не змінюються, являючи собою відбитки даних, зроблені у певний час);

о мінімізація збитковості інформації (перед завантаженням у сховища дані фільтруються, зберігаються у певній послідовності, а також формується деяка підсумкова інформація).

В сховищах даних надмірність даних є мінімальною (приблизно 1%), оскільки:

о при завантаженні у сховище дані сортуються і фільтруються;

о інформація у сховищах зберігається в хронологічному порядку, що майже повністю виключає перекриття даних;

о при завантаженні у сховище дані зводяться до єдиного формату, включаючи обчислення підсумкових (агрегованих) показників.

Сервери багатовимірних баз даних можуть зберігати дані по-різному, крім агрегованих показників формується ще й додаткова інформація: поля часу, дати; адресні посилання, таблиці метаданих тощо. Це приводить до значного збільшення інформації. Вхідний масив розміром 200 Mb може розростись до об'єму 5 Gb. Сховище даних повинне бути оптимально організованою базою даних, яка забезпечує максимально швидкий і оперативний пошук інформації.

Вітрина даних – це спрощений варіант сховища даних, що містить лише тематично орієнтовані, агреговані дані

Глобальне сховище даних складається з трьох рівнів:

- 1) сховище агрегованих даних;
- 2) вітрини даних, які базуються на інформації зі сховища даних;
- 3) клієнтські робочі місця, на яких встановлено засоби оперативного аналізу даних.

Всі дані в Схов.Д поділяються на:

- 1) *детальні дані* -переносяться безпосередньо з ОДД, вони відповідають елементарних подіям, що фіксуються OLTP-системами (наприклад, продаж, експерименти тощо). Їх поділяють на *вимірювання* (набір даних, необхідних для опису події – міста, товару, людей) та *факти* – відображають сутність події (кількість проданого товару, результати експериментів).
- 2) *агреговані дані* отримуються на основі детальних даних шляхом сумування числових фактичних даних по всім вимірам (адитивні (можуть бути просумовані), напівадитивні (можуть бути прогумовані тільки по деяким вимірам), неадитивні).
- 3) *метадані* – це дані про дані, які визначають джерело, приймач та алгоритм трансформації даних під час перенесення їх від джерела до приймача.

Метадані містять:

- описи структур даних та їхніх взаємозв'язків;
- інформацію про джерела даних і про ступінь їх вірогідності;
- інформацію про власників даних, права доступу;
- схему перетворення стовпців вхідних таблиць у стовпці кінцевих таблиць;
- правила підсумовування, консолідації та агрегування даних;
- інформацію про періодичність оновлення даних;

- каталог використаних таблиць, стовпців та ключів;
- фізичні атрибути стовпців;
- кількість табличних рядків та обсяг даних;
- часові ярлики (дата та час створення/модифікації записів);
- статистичні оцінки часу виконання запитів.

Контроль модифікації (versioning) полягає у властивості метаданих відслідковувати зміни в структурі даних та їх значення в часі.

Функціональна архітектура сховища даних містить наступні компоненти:

- 1) сховище даних;
- 2) клієнтська частина системи (дизайнери сховища, засоби розробки додатків, засоби адміністрування, інструменти аналізу даних, завантаження словника метаданих з XML-файлу у сховище і експорт його зі сховища в XML-файл);
- 3) сервер обміну даними (Data Exchange Server) - набір програм імпорту/експорту даних зі сховища й каталогів для організації обміну даними із зовнішніми OLTP-системами;
- 4) бібліотеки прикладних класів: ACL (Application Class Library), VCL (Visual Component Library), Win Lite.

Наповнення інформаційних сховищ відбувається в декілька етапів:

- 1) екстракція (витяг) – імпорт даних у сховище з інформаційних підсистем, виробничих відділів та інших джерел;
- 2) трансформація – консолідування, агрегування даних, розбиття їх на фракції, коригування та трансформування у відповідні формати;
- 3) завантаження – у сховище, синхронізація з датою або зовнішніми подіями.

Обслуговування інформаційних сховищ полягає в:

копіюванні баз даних, налаштуванні, тиражуванні, надсиланні застарілих баз даних до архіву, управлінні правами користувачів, створенні та редагуванні графічних діаграм баз даних, тощо.

Для ефективної роботи зі сховищем даних, необхідно зібрати максимум інформації про процес. Наприклад, для прогнозування обсягів продажів можуть бути використані бази даних облікових систем компанії, маркетингові дані, відгуки клієнтів, дослідження конкурентів і т.п.

Необхідною для прогнозу є наступна інформація:

- о хронологія продажів;
- о стан складу на кожний день - якщо спад продажів буде пов'язаний із відсутністю товару на складі, а не через відсутність попиту;
- о відомості про ціни конкурентів;
- о зміни у законодавстві;
- о загальний стан ринку;
- о курс долара, інфляція;
- о відомості про рекламу;
- о відомості про відношення до продукції клієнтів;
- о різного роду специфічну інформацію. Наприклад, для продавців морозива - температуру, а для фармакологічних складів -санітарно-епідеміологічний стан, тощо.

Проблема полягає в тому, що зазвичай в системах оперативного обліку більша частина цієї інформації відсутня, а наявна - неповна або спотворена. Кращим варіантом в цьому випадку буде створення сховища даних, куди б з певною заданою періодичністю надходила вся необхідна інформація, задалегідь систематизована і очищена (рис.5.1).



Рис.5.1. Приклад сховища даних

Ефективна архітектура сховища даних організовується таким чином, щоб бути складовою частиною інформаційної системи управління підприємством.

Найбільш поширений випадок, коли сховище організовано за типом "зірка", де в центрі розміщуються факти і агрегатні дані, а "проміннями" є виміри. Кожна "зірка" описує певну дію, наприклад, продаж товару, його відвантаження, надходження коштів й інше:

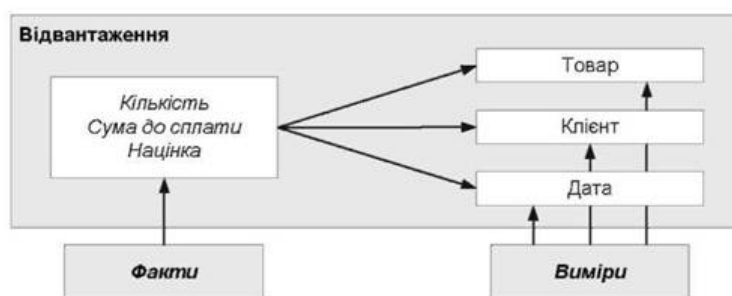


Рис.5.2. Схема організації сховища даних за типом "зірка"

Як правило, дані копіюються в сховище з оперативних баз даних і інших джерел відповідно до певних правил.

Лекція № 3

Тема лекції: СХОВИЩА ДАНИХ ТА OLAP ТЕХНОЛОГІЇ В СППР

План лекції

1. Багатомірна модель даних.
2. Архітектура OLAP.

Ключові слова

інформаційні системи; інтелектуальний аналіз даних, класифікація ІСППР, бази даних, реляційний підхід до побудови баз даних.

Література

1. Галузинський Г. П. Перспективні технологічні засоби оброблення інформації : навч.-метод. посіб. для самост. вивчення дисципліни / Г. П. Галузинський, І. В. Гордієнко ; Київський національний економічний ун-т. - К. : КНЕУ, 2002. - 279 с.
2. Введение в OLAP: часть 1. Основы OLAP - http://www.olap.ru/basic/OLAP_intro1.asp
3. Системи підтримки прийняття рішень [Текст] : навчальний посібник для самостійного вивчення дисципліни / [уклад.: С. М. Братушка, С. М. Новак, С. О. Хайлук] ; Державний вищий навчальний заклад "Українська академія банківської справи Національного банку України". – Суми : ДВНЗ "УАБС НБУ", 2010. – 265 с.
4. Інтелектуальна обробка інформації до виконання лабораторних робіт(OLAP) для студентів напряму підготовки "Інформаційні мережі зв'язку". - К.:НТУУ "КПІ", 2009. – 54с.

Зміст лекції

1. Багатомірна модель даних.

Сховища даних не вирішують задач, пов'язаних з організацією ефективного аналізу даних і надання доступу до них. Ці задачі вирішуються підсистемами аналізу.

В процесі аналізу дуже часто виникає необхідність в побудові залежностей між різними параметрами. Крім того, число таких параметрів може коливатись в широких межах.

Цю проблему вирішує можливість вимірення.

Вимірення – це послідовність значення одного з параметрів, що аналізуються. Наприклад, для параметру «час» це послідовність календарних днів, для параметру «регіон» - список міст.

Множина вимірень передбачає представлення даних у вигляді багатомірної моделі, тобто у множинній перспективі, яка складається з декількох незалежних вимірень, згідно яких можуть бути проаналізовані певні сукупності даних. Одночасно аналіз по декількох вимірнях визначається як *багатомірний аналіз*.

Кожне вимірня може бути подане у вигляді ієрархічної структури. Н., вимірня «виконавець: підприємство – підрозділ – відділ – службовець». Деякі вимірня можуть мати декілька видів ієрархічного подання. Н., «час»: рік – квартал – місяць – день, або тиждень – день.

Не перетині осей вимірень розташовуються дані, які кількісно характеризують факти, що підлягають аналізу – *міра* (Н., обсяги продаж в одиницях продукції, в грошову вимірі, залишки ан складі, витрати тощо).

Таким чином, багатомірну модель даних можна уявити у вигляді кубу. Ребра такого куба не є рівними в даному понятті. Ребра – це вимірня, ячейка – міра.

OLAP (On-Line Analytical Processing) є ключовим компонентом організації сховищ даних. Ця технологія заснована на побудові і візуалізації багатомірних кубів даних з можливістю довільного маніпулювання даними, що містяться в кубі (рис.1). Це дозволяє представити дані для аналізу в будь-якому розрізі.

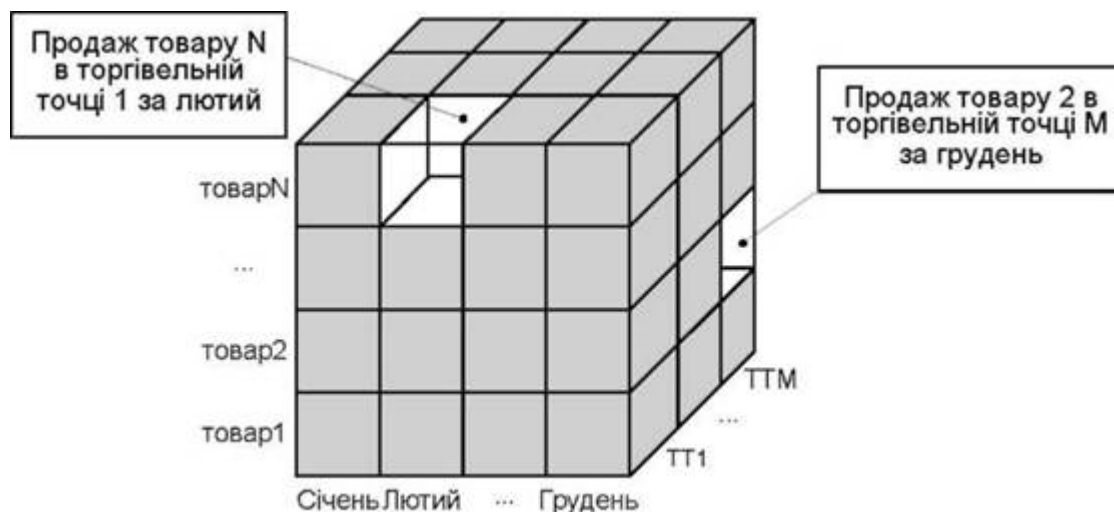


Рис.1. Схема організації розміщення даних для технології OLAP

Наприклад, аналіз продажів. Припустімо, що керівнику необхідно знати обсяги продажів за деякий період, (наприклад, за місяць), що нещодавно завершився. При цьому, компанія продає не один, а множину товарів і має велику кількість торговельних точок (ТТ) -магазинів.

Перші два найпростіші питання, на які потрібно мати відповіді, - це обсяги продажів товарів в кожній торгівельній точці та за кожний місяць.

Відповідь на ці питання оформляється у вигляді двовимірної таблиці. В першому випадку рядками і стовпцями цієї таблиці відповідно будуть назви товарів, місяці і суми, а в другому - назви ТТ і суми:

Таблиця 1. Приклад багатовимірного звіту

Місяць	ТТ	Товар 1	Товар 2	Товар 3	Товар 4	Разом
Січень	ТТ1	913,45	861,28	1056,94	1345,52	4177,19
	ТТ5	970,14	1789,58	1865,78	2299,86	6925,36
	Разом	1883,59	2650,86	2922,72	3645,38	11102,55
Лютий	ТТ1	911,16	843,40	1653,50	1171,23	4579,29
	ТТ5	1291,62	1655,36	2528,30	2741,24	8216,52
	Разом	2202,78	2498,76	4181,8	3912,47	12795,81
Березень	ТТ1	790,07	954,59	1093,96	583,46	3422,08
	ТТ5	1599,17	1451,85	2041,93	2326,35	7419,30
	Разом	2389,24	2406,44	3135,89	2909,81	10841,38

Проте, аналіз даних в такому представленні є ускладненим. Виникає потреба об'єднання даних декількох таблиць. В результаті у звіті фігуруватиме три аналітичні виміри (місто, товар і торгівельна точка), і замість двовимірних таблиць буде використовуватись тривимірна модель представлення даних.

Технологія комплексного багатовимірного аналізу даних OLAP представляє собою головний компонент організації сховищ даних, а саме: збору, очищення й попередньої обробки інформації. Часто OLAP характеризують як швидкий аналіз багатовимірної розподіленої інформації.

Автор реляційної моделі даних Е.Ф. Кодд сформулював 18 правил OLAP, які розділив на 4 групи:

Основні властивості:

1. багатовимірне представлення даних;
2. інтуїтивне оперування даними (без застосування меню);
3. доступність;
4. пакетне вилучення замість інтерпретації;

5. моделі аналізу OLAP (за категоріями, тлумачний, абстрактний і стереотипний);

6. архітектура "клієнт-сервер";

7. прозорість;

8. можливість одночасного обслуговування багатьох користувачів.

Спеціальні властивості:

9. обробка ненормалізованих даних;

10. збереження результатів OLAP;

11. вилучення значень, яких немає (відрізняються від нульових значень);

12. обробка значень, яких немає (ігноруються OLAP-аналізатором без врахування їх джерел).

Особливості представлення звітів:

13. гнучкість формування звітів;

14. стандартна продуктивність звітів (не знижується із зростанням кількості вимірів і об'єму бази даних);

15. автоматичне налаштування фізичного рівня.

Управління вимірами:

16. універсальність вимірів;

17. необмежена кількість вимірів і рівнів агрегації;

18. необмежені операції над розмірностями.

2. Архітектура OLAP-систем

OLAP-система включає в себе 2 основних компоненти:

1) *OLAP-сервер* – забезпечує зберігання даних, виконання над ними необхідних операцій та формування багатомірної моделі на концептуальному рівні. В даний час OLAP-сервери об'єднуються з сховищами даними або вітринами даних;

2) *OLAP-клієнт* – представляє користувачу інтерфейс до багатомірної моделі даних, забезпечуючи його можливістю зручно маніпулювати даними для виконання задач аналізу.

OLAP-сервери ховають від кінцевого користувача спосіб реалізації багатомірної моделі. Вони формують гіперкуб, з яким користувач через OLAP-клієнта виконує всі необхідні маніпуляції, аналізуючи дані. Між тим, спосіб реалізації дуже важливий, т.як. від нього залежать такі характеристики як продуктивність і задіяні ресурси.

Виділяють три основні способи реалізації (Корпоративне сховище даних може функціонувати в трьох архітектурах) -:

MOLAP – багатовимірній OLAP – для реалізації багатомірної моделі використовують багатомірні БД;

ROLAP – реляційній OLAP – для реалізації багатомірної моделі використовують реляційні БД.

У ROLAP (Relation OLAP) – архітектурі дані зберігаються в реляційній базі даних, а агреговані – у спеціальних службових таблицях. Реляційні таблиці і зв'язки між ними генеруються автоматично. Головні функції системи розподіляються між трьома логічними рівнями:

- масштабована паралельна реляційна база даних забезпечує зберігання і швидкий доступ;

- середній рівень аналізу підтримує багатовимірне представлення даних і розширені функціональні можливості, які є недоступними на базовому реляційному сервері;

- рівень представлення відповідає за донесення результатів до користувачів.

Реляційна архітектура забезпечує високу швидкість роботи зі сховищем при невеликих обсягах даних.

HOLAP – гібридній або змішаній OLAP – для реалізації багатомірної моделі використовують і багатомірні, і реляційні БД.

HOLAP (Hybrid OLAP) – передбачає збереження основних даних в реляційному сховищі, а агрегованих - в багатовимірній базі MOLAP (Multidimensional OLAP) у зручному для користувача вигляді.

Часто в літературі зустрічаються:

DOLAP – настільна OLAP. Є недорогою та простою у використанні OLAP-системою, для локального аналізу і представлення даних.

JOLAP – нова, заснована на Java, для створення і управління даними.

Комп'ютер, що керує певним ресурсом, називають сервером цього ресурсу, комп'ютер, який ним користується - клієнтом (файл-сервер, сервер баз даних). Той самий комп'ютер може виконувати як роль сервера, так і клієнта.

Цей принцип поширюється і на взаємодію програм. Якщо одна з них виконує деякі функції, надаючи іншим відповідний набір послуг, то вона називається сервером. Програми, що користуються цими послугами, називаються клієнтами (SQL-сервер і SQL-клієнт).

Підтримка інтелектуального аналізу є одним із базових компонентів сучасних корпоративних інформаційних систем. В межах OLAP-технологій вона отримала самостійний розвиток в сучасних технологіях.

Лекція № 4

Тема лекції: Нейрокомп'ютерні технології та мережі.

План лекції

1. Поняття добуття даних Data Mining.
2. Задачі Data Mining.
3. Практичне застосування Data Mining
4. Моделі Data Mining.
5. Методи Data Mining.
6. Процес знаходження даних.

Ключові слова

Ключові поняття: нейронні мережі; дерева рішень; карти, що самоорганізуються; коннекціонізм; управління в реальному часі; розпізнання зображень.

Література

1. ↑ *Дунин-Барковский В. Л.*, Нейрокибернетика, Нейроинформатика, Нейрокомпьютеры, В кн.: Нейроинформатика / *А. Н. Горбань, В. Л. Дунин-Барковский, А. Н. Курдин* и др. — Новосибирск: Наука. Сибирское предприятие РАН, 1998. — 296 с
2. ↑ *Уидроу Б., Стирнс С.*, Адаптивная обработка сигналов. — М.: Радио и связь, 1989. — 440 с.
3. ↑ *Савельев А. В.*, Нейрокомпьютеры в изобретениях // «Нейрокомпьютеры: разработка, применение», М. Радиотехника, 2004, № 2-3, с. 33-49.
4. ↑ *Терехов В. А., Ефимов Д. В., Тюкин И. Ю.* Нейросетевые системы управления. — М.: Высшая школа, 2002. — С. 184.
5. ↑ *Галушкин А. И.*, Применения нейрокомпьютеров в финансовой деятельности.
6. ↑ *Миркес Е. М.*, Логически прозрачные нейронные сети и производство явных знаний из данных, В кн.: Нейроинформатика / *А. Н. Горбань, В. Л. Дунин-Барковский, А. Н. Курдин* и др. — Новосибирск: Наука. Сибирское предприятие РАН, 1998. — 296 с

Зміст лекції

1. Поняття добуття даних Data Mining.

OLAP-системи являють собою аналітику засобу перевірки гіпотез при аналізі даних. При цьому основною задачею аналітика є генерація гіпотез. Він вирішує її, базуючись на власних знаннях та досвіді. Однак, знання є не

тільки у людини, а й у накопичених даних, які підлягають аналізу. Такі знання називають «схованими», так вони знаходяться в гігабайтах і терабайтах інформації, які людина не в змозі дослідити самотійно. У зв'язку з цим, існує ймовірність пропустити гіпотези, які можуть принести значну вигоду.

Очевидно, що для знаходження необхідних даних необхідно застосовувати спеціальні методи автоматичного аналізу, за допомогою яких практично добуваються знання із завалів інформації. За цим направленням закріпився термін *Data Mining*.

Data Mining (добування знань, даних) – технологія аналізу сховищ даних, що ґрунтується на методах штучного інтелекту та інструментах підтримки прийняття рішень. Зокрема сюди входить знаходження трендів і комерційно корисних залежностей. Деколи використовують термін "knowledge discovery" (виявлення знань) – виявлення прихованих структур (patterns) у сховищах даних, щоб перетворити їх на знання або термін "інтелектуальний аналіз даних". Всі ці терміни є синонімами.

Класичне визначення технології "видобування даних" (Data Mining) звучить таким чином: це виявлення в початкових ("сирих") даних – раніше невідомих, нетривіальних, практично корисних і доступних інтерпретації знань.

Тобто інформація, знайдена в процесі застосування методів Data Mining, повинна бути нетривіальною і раніше невідомою, наприклад, отриманий показник середніх продажів не є таким. Знання повинні описувати нові зв'язки між властивостями, передбачати значення та характеристики одних параметрів на основі інших.

Основна мета *Data Mining* полягає у виявленні прихованих правил і закономірностей у великих масивах даних. На відміну від оперативної аналітичної обробки даних (OLAP), у *Data Mining* основну задачу формулювання гіпотез і виявлення незвичайних шаблонів (закономірностей) перекладено з людини на комп'ютер.

Фірми Oracle, Microsoft, IBM т. ін. випустили ряд продуктів (DarWin, Microsoft SQL Server 200, IBM Intelligent for Data, відповідно), що реалізують алгоритми *Data Mining* і дають змогу автоматизувати процес аналізу даних.

В *Data Mining* для представлення отриманих знань служать моделі. Види моделей залежать від їх створення. Найбільш розповсюджені: правила, дерева рішень, кластери та математичні функції.

2. Задачі Data Mining.

Задача класифікації та регресії.

При аналізі дуже часто необхідно визначити до якого з відомих класів належать об'єкти, що досліджуються, тобто класифікувати їх.

Н., коли в банк звертається людина за кредитом, банк.службовець повинен визначити: кредитоздатний потенційний клієнт чи ні. Очевидно, таке рішення приймається на сонові даних про людину: місце роботи, розмір зарплати, вік, склад сім'ї тощо. В результаті аналізу банк.службовець відносить людину до однієї з категорій: кредитоздатний, некредитоздатний.

Н., фільтрація електронної пошти (спам чи лист) на основі найбільш вживаних слів: імені отримувача чи безособового звернення, слів типу: купити, заробить, вигідна пропозиція тощо.

Кількість класів може бути більше двох (Н., в задачі розпізнавання образу цифр в десятковій системі обчислення їх може бути 10). В такій задачі об'єктом класифікації є матриця пік селів, яка представляє образ цифри, що розпізнається.

В Data Mining задачу класифікації розглядають як визначення значення одного параметра з параметрів об'єкта, що аналізується, на основі параметрів інших об'єктів. Параметр, який визначається називають залежною перемінною, а параметри, що беруть участь у його визначенні – незалежними.

Якщо значення незалежних та залежних змінних є дійсними числами, то задача називається *задачею регресії*. Н., визначення суми кредиту, яка може бути видана банком клієнту.

Задачі класифікації та регресії вирішуються в 2 етапи:

1) виділяється навчаюча вибірка, в яку входять всі об'єкти, для яких відомі значення залежних та незалежних змінних. Н.,:

- інформація про клієнтів, яким раніше надавались кредити на різні суми і інформація про їх погашення;
- повідомлення, що класифікуються вручну як спам і як лист;
- розпізнані раніше матриці образів цифр.

На основі вибірки будується модель визначення значення залежної змінної. Її часто називають функцією класифікації чи регресії

2) побудовану модель застосовують на об'єктах, що аналізуються (до об'єктів, що з невизначеним значенням залежної змінної).

Задачі класифікації та регресії мають геометричну інтерпретацію, що дозволяє розглядати їх у двомірному просторі.

Задача пошуку асоціативних правил.

Найбільш популярний додаток *Data Mining* пошук асоціативних правил. Суть задачі полягає в пошуку наборів об'єктів, що зустрічаються

найчастіше у великій множині таких наборів. Це частковий випадок класифікації. Спочатку вона вирішувалась при аналізі тенденції поведінки покупців ц супермаркеті (аналізувались дані про покупки, які вони здійснюють: які товари купуються разом, в якій послідовності, які категорії покупців яким товарам віддають перевагу і т.д.) .

Актуальна не тільки в торгівлі, а й сфері обслуговування (якими послугами клієнти хочуть користуватись в сукупності).

При аналізі дуже часто викликає інтерес послідовність, з якою відбуваються події. При знаходженні закономірності в таких подіях з деякою ймовірністю можна передбачати події в майбутньому, що дозволяє приймати вірні рішення. Така задача є різновидом асоціативних правил та називається *синквенціальним аналізом*.

Шик орок використовується в телекомунікаційних компаніях, для аналізу даних про аварії на різних вузлах мережі.

Задача кластеризації.

Полягає в розподілі множини об'єктів на групи «схожих» об'єктів (кластери). Рішення задачі розподілу елементів на кластери називають кластерним аналізом.

Застосовується в будь-якій області. Де необхідно дослідження експериментальних або статистичних даних.

В маркетингу – сегментація.

Приклади: таблиця Менделєєва.

Кластеризація відрізняється класифікації від тим, що для проведення аналізу не потрібно мати виділену залежну змінну. Тут характерна відсутність яких-небудь відмінностей як між перемінними, так і між об'єктами. Тут шукають групи найбільш близьких, схожих об'єктів. Методи кластерного аналізу рідко використовується самі по собі. Після визначення кластерів зазвичай використовуються інші методи *Data Mining*.

Дозволяє розглядати достатньо великий обсяг інформації і різко скорочувати, ужимати великі масиви інформації, робити їх компактними та наглядними.

4. Моделі Data Mining.

Ціль технології Data Mining – заходження в даних таких моделей, які не можуть бути знайдені іншими звичними методами.

Два види моделей:

1) *передбачувальні моделі* (предсказательний) – будуються на сонові набору даних з відомими результатами. Використовуються для передбачення

результатів на основі інших наборів даних.. при цьому вимагається, щоб модель працювала достатньо точно і була статистично значима і оправдана. До них відносяться:

- *моделі класифікації* – описують правила та набір правил, у відповідності з яким можна віднести опис нового об'єкта до одного з класів. Такі правила будуються на основі інформації про існуючі об'єкти шляхом розподілу їх на класи.

- *моделі послідовностей* – описують функції, які дозволяють прогнозувати зміни неоперених числових параметрів. Будуються на основі інформації про зміни деякого параметра за минулий період часу.

2) *описові моделі* – досліджують сутність залежностей в наборі даних, взаємний вплив різних факторів, тобто побудові емпіричних моделей різних систем. Легкі та прозорі для сприйняття людиною. До них відносяться:

- *регресійні* – описують функціональні залежності між залежними та незалежними показниками в зрозумілій для людини формі.

- *моделі кластерів* – описують групи кластерів на які можна розділити об'єкти, дані про які підлягають аналізу. Чим більше схожі між собою об'єкти всередині кластера, тим точніша кластеризація;

- *моделі виключень* – описують виключні ситуації в записях (н., окремих пацієнтів), які різко відрізняються чим-небудь від основної множини записів.

- *підсумкові моделі* – виявляють обмеження для даних масиву, що аналізується (Н., при виборці пацієнтів не старше 30 років, що перенесли інфаркт міокарда, з'ясовано, що всі пацієнти, які описані в даній вибірці, або курять більше 5 пачок цигарок в день, або мають вагу не нижче 95 кг.);

- *асоціативні моделі* – виявляють закономірності між зв'язаними подіями. Н., такою закономірністю служить правило, що із події X слідує подія Y. Такі правила називають асоціативними.

Для побудови розглянутих моделей використовуються різні методи та алгоритми Data Mining, більшість з яких були розроблені на основі різноманітних технологій та концепцій, пов'язаних зі статистикою, теорією інформації тощо.

5. Методи Data Mining

Методи Data Mining дозволяють виявляти стандартні закономірності:

- *асоціація* (кілька подій пов'язані одна з одною, наприклад при купівлі пива дуже часто купують і чіпси чи горішки);
- *послідовність* (ланцюжок пов'язаних у часі подій, наприклад: нова квартира - нові меблі);
- *кластеризація* (відрізняється від класифікації тим, що групи заздалегідь не створені. Використовується для сегментації ринку і замовників);
- *прогнозування* (базою служить історична інформація. Ґрунтується на побудові математичних моделей).

Базові методи.

Сюди насамперед прийнято відносити алгоритми, що базуються на переборі. Простий перебір досліджуваних об'єктів вимагає $O(2^N)$ операцій, де N – кількість об'єктів. зі збільшенням кількості даних обсяг обчислюваних операцій зростає експотенціально (в геометричній прогресії), що при великому обсягу робить рішення будь-якої задачі практично неможливим.

Для скорочення обчислювальної складності в таких алгоритмах використовують різного виду евристики, що призводять до скорочення перебору. оптимізація подібних алгоритмів зводиться до приведенню залежності кількості операцій від кількості досліджуваних даних до лінійної функції.

Сюди ж відносять ще підходи, що використовують елементи теорії статистики, їх достатньо багато. Основна ідея зводиться до кореляційного, регресійного та інших видів стат.аналіза.

Нечітка логіка.

Основним способом дослідження задач аналізу даних є їх відображення на формалізовану мову і наступний аналіз отриманої моделі. Невизначеність за обсягом відсутньої інформації у системного аналітика можна розділити на 3 великі групи:

1. Невідомість.
2. Неповнота (недостатність, неадекватність).
3. Недостовірність. Буває *фізичною* (джерелом її є зовнішнє середовище) і *лінгвістичною* (виникає в результаті мовного спілкування і зумовлена необхідністю описання нескінченного числа ситуацій обмеженим числом слів за обмежений час).

Виділяють 2 вида фізичної невизначеності:

1. Неточність (неточність вимірювань значень визначеної величини, які виконуються фізичними приборами).

2. Випадковість (або наявність у зовнішньому середовищі декількох можливостей, кожна з яких випадковим образом може стати дійсністю; припускається знання відповідного закону розподілу ймовірностей).

Виділяють 2 види лінгвістичної невизначеності:

1. Невизначеність значень слів (багатозначність, розмитість, неясність, нечіткість). Виникає, коли відображаються одним і тим словом об'єкти задачі управління, які є за своєю суттю різними.

2. Неоднозначність змісту фраз (виділяють синтаксичну і семантичну).

Основна сфера застосування нечіткої логіки – управління.

Із збільшенням розмірів і складності системи суттєво ускладнюється її моделювання за допомогою відомих математичних виразів. Це пов'язано з збільшенням числа змінних та параметрів, підвищення складності виміру окремих параметрів. В результаті створення адекватної моделі стає практично неможливим. Тому було прийнято застосовувати лінгвістичну модель, яка використовує не математичні вирази, а слова, які відображають якість. Такі моделі хоча й забезпечують точність, аналогічну мат.моделюванню, та все ж неможливо створити якісну модель в такий спосіб. В такому випадку предметом обговорень є нечіткість слів мови опису системи.

Людині в процесі управління складними об'єктами властиво оперувати поняттями і відношеннями в розмитих кордонах. Вихідні поняття, що використовуються – нечітка множина і лінгвістичні змінна – суттєво розширюють можливості формалізації опису подібних складних систем. Такі моделі називають лінгвістичними.

Мовна модель «якщо...., то.....».

Методи управління на основі нечіткої логіки можна вважати в багато чому евристичними.

Генетичні алгоритми.

Відносяться до числа універсальних методів оптимізації, що дозволяють вирішувати задачі різних типів і ступня складності. Мають можливість як однокритеріального, так і багатокритеріального пошуку.

Інтеграція ГА та нейронних мереж, нечіткої логіки дозволяє вирішувати проблеми пошуку оптимальних значень ваги входів нейронів, оптимізувати систему продукційних правил, які можуть бути використані для управління операторами ГА.

Нейронні мережі.

Це клас моделей, що базуються на біологічній аналогії з мозком людини і призначені для проходження етапу навчання на існуючих даних для рішення різнобічних задач аналізу даних. При застосування цих методів необхідно обрати конкретну архітектуру мережі (число «слоїв», кількість «нейронів» в кожному з них). Розмір та структура мають відповідати змісту досліджуваного явища.

Потім побудована мережа піддається навчанню. На цьому етапі нейрони ітеративно обробляють вхідну інформацію і коригують свої ваги так, щоб мережа найбільш точно прогнозувала дані.

Нейронна мережа може видавати дуже точні прогнози, однак залежності в моделях не можуть бути описані в явному вигляді.

6. Основні етапи аналізу процесу знаходження знань.

1. Розуміння і формулювання задачі аналізу.
2. Підготовка даних для автоматизованого аналізу.
3. Застосування методів Data Mining і побудова моделей.
4. Перевірка побудованих моделей.
5. Інтерпретація моделей людиною.

Засоби Data Mining.

В даний час технологія Data Mining представлена цілим рядом комерційних та вільно розповсюджуваних продуктів (www.kdnuggets.com). Класифікувати програмні продукти можна за тим же принципом, що й саму технологію.

Умовно продукти Data Mining можна розділити на 3 великі категорії:

- вхідні як невід'ємна частина в системи управління базами даних;
- бібліотеки алгоритмів Data Mining з супровідною інфраструктурою;
- коробкові або настільні рішення ("чорні ящики").

Продукти перших двох категорій надають найбільші можливості для інтеграції та дозволяють реалізовувати аналітичний потенціал практично в будь-якому додатку (приложенні) в будь-якій області.

Коробкові надають деякі унікальні досягнення і можуть бути спеціалізованими в якійсь конкретній сфері застосування. однак в більшості випадків їх складно інтегрувати в більш широке застосування.

Функціональність Data Mining в даний час реалізована в наступних комерційних базах даних:

Oracle;

Microsoft SQL Server;

IBM DB2.

Існують два способи впровадження нової інформаційної технології в локальні інформаційні структури:

1. пристосування її до організаційної структури підприємства;
2. модернізування організаційної структури з метою найбільш ефективного використання нової інформаційної технології.

Перший спосіб є дешевшим і не вимагає великих змін в організації діяльності підприємства. Проте ефект від його впровадження може бути незначним. Другий спосіб вимагає більших капіталовкладень, але забезпечує якісно новий рівень діяльності підприємства чи організації.

Наведені нижче приклади з різних областей економіки демонструють основну перевагу методів Data Mining - здатність виявлення нових знань, які неможливо отримати методами статистичного, регресивного аналізу або економетрики.

1. Клієнти компанії за допомогою одного з інструментів Data Mining були об'єднані в сегменти з схожими ознаками. Це дозволило проводити різну маркетингову політику і будувати окремі моделі поведінки для кожного сегменту. Найважливішими чинниками для розподілу були: віддаленість регіону клієнта, сфера діяльності, середньорічні суми операцій, кількість операцій за тиждень.

2. Автоматичний аналіз банківської бази даних кредитних операцій фізичних осіб виявив правила, за якими позичальникам відмовляли у видачі кредиту. Вирішальними чинниками, виявились: термін кредиту, середньомісячний дохід і витрати позичальника. Надалі це враховувалося при експрес-кредитуванні.

3. При аналізі бази даних клієнтів страхової компанії був встановлений соціальний портрет людини, що страхує життя - це виявився чоловік 35-50 років, що має двох і більше дітей і середньомісячний дохід вище \$2000.

Висунення гіпотез. Під гіпотезою в даному випадку будемо розуміти припущення про вплив певних чинників на досліджувану задачу. При цьому форма цієї залежності в значення не має. Тобто можна припустити, що на продаж впливає відхилення ціни на товар від середньоринкової, але при цьому не зазначати, як саме цей чинник впливає на продажі. Для вирішення цієї задачі і використовується Data Mining. Наприклад, для обробки даних про обсяги продажів певного товару висувається гіпотеза про вплив чинника його відсутності у торговій точці.

Автоматизувати процес висунення гіпотез не представляється можливим, принаймні, на сьогоднішньому рівні розвитку технологій. Цю задачу повинні вирішувати експерти - фахівці в даній області. З

використанням їх знань про предмет, методом опитування накопичується максимальна кількість гіпотез/припущень.

Результатом цього кроку буде список з описом всіх чинників. Наприклад, для задачі прогнозування попиту це може бути список наступного вигляду: сезон, день тижня, обсяги продажів за попередні тижні, обсяги продажів за аналогічний період минулого року, рекламна компанія, маркетингові заходи, якість продукції, бренд, відхилення ціни від середньоринкової, наявність даного товару у конкурентів, тощо.

При розв'язку певної задачі необхідно створювати спеціалізований набір даних, причому їх велика кількість не обумовлює якість рішення.

Після підготовки таблиці з описом чинників експертно оцінюється значущість кожного з чинників. Ця оцінка не є остаточною, вона служить відправним пунктом. В процесі аналізу може виявитися, що чинник, який експерти вважали вкрай важливим, таким по суті не є і, навпаки, незначущий із їхньої точки зору чинник може мати значний вплив. У будь-якому випадку, всі варіанти проаналізувати відразу неможливо, потрібно від чогось відштовхуватися, цією крапкою і є оцінка експертів. До того ж, досить часто реальні дані підтверджують їх оцінку. Результатом цього кроку може бути таблиця наступного вигляду (табл.5.2).

Коли гіпотезу висуває один експерт, задача оцінки значущості істотно спрощується. Проте, із зростанням складності системи, зростає і складність отримання адекватної оцінки експертів.

Таблиця 5.2. Приклад експертної оцінки значущості чинників

Чинник	Оцінка значущості (< 100)
Сезон	100
День тижня	80
Обсяги продажів за попередні тижні	100
Рекламна кампанія	60
Маркетингові заходи	40
Якість продукції	50
Відхилення ціни від середньоринкової	60
Наявність даного товару у конкурентів	15

Часто думки експертів можуть розходитись - виникає питання одержання середніх показників з десятків думок. Для цього існують спеціальні математичні методи - методи проведення складних експертиз: ранжування, парне порівняння та інші.

Лекція № 5

Тема лекції: Асоціативні правила та дерева рішень в ІСППР

План лекції

1. Асоціативні правила в ІСППР.
2. Дерева рішень в ІСППР

Ключові слова

дерева рішень; карти, що самоорганізуються; асоціативні правила; аналіз “що – якщо?; попередня обробка; візуалізатор; підтримка; вірогідність

Література

1. Галузинський Г. П. Перспективні технологічні засоби оброблення інформації : навч.-метод. посіб. для самост. вивчення дисципліни / Г. П. Галузинський, І. В. Гордієнко ; Київський національний економічний ун-т. - К. : КНЕУ, 2002. - 279 с.
2. Введение в OLAP: часть 1. Основы OLAP - http://www.olap.ru/basic/OLAP_intro1.asp
3. Системи підтримки прийняття рішень [Текст] : навчальний посібник для самостійного вивчення дисципліни / [уклад.: С. М. Братушка, С. М. Новак, С. О. Хайлук] ; Державний вищий навчальний заклад “Українська академія банківської справи Національного банку України”. – Суми : ДВНЗ “УАБС НБУ”, 2010. – 265 с.
4. Інтелектуальна обробка інформації до виконання лабораторних робіт(OLAP) для студентів напряму підготовки "Інформаційні мережі зв'язку". - К.:НТУУ "КПІ", 2009. – 54с.
5. Рогоза М.Є. Системи підтримки і прийняття рішень/ М.Є.Рогоза, ОО. Ємець, Є.М. Ємець. – Полтава,2013. – С. 30-66 – [Дерево рішень]

Зміст лекції

1. Асоціативні правила в ІСППР.

Останнім часом задачі пошуку нових знань у великих базах сирих даних стають все більш популярними та актуальними. Одним із популярних методів виявлення знань став алгоритм пошуку так званих асоціативних правил (Association Rules). Суть задачі полягає в знаходженні наборів об'єктів, які зустрічаються найчастіше серед всієї множини ймовірних наборів об'єктів. Першим застосуванням такої задачі був аналіз тенденцій в поведінці покупців у супермаркетах. При цьому аналізувались дані про всі здійснені покупки, які кожен покупець кладе у свій кошик, та одержувалась інформація про те, які товари переважно купуються разом, в якій послідовності, якими категоріями покупців, в які періоди часу, тощо. Такого роду знання дозволяють ефективно планувати закупку товарів у магазин, розробляти ефективні рекламні кампанії та розкладати товар таким чином, щоб провокувати покупців на різноманітні покупки.

Наприклад, з набору товарів, які купуються в магазинах, можна виділити такі набори товарів, що переважно купуються одночасно:

- {чіпси, пиво};
- {вода, горіхи};
- {чай, печиво};
- Тощо.

Таким чином, можна зробити висновок про те, що якщо купуються чіпси чи горіхи, то, як правило, купуються, пиво чи вода, відповідно. Отже, можна розмістити ці товари поруч на прилавках, об'єднати їх в один пакет зі знижкою чи здійснити інші дії.

Задача пошуку асоціативних правил є актуальною не лише у сфері торгівлі. Наприклад, в сфері обслуговування цікавою є інформація про те, якими послугами клієнти користуються в сукупності. Для одержання цієї інформації вирішується задача аналізу даних про послуги, якими користується один клієнт протягом певного часу. Це допомагає визначити, наприклад, як найбільш вигідно сформувати пакети послуг для клієнтів.

В медицині аналізуватись можуть симптоми та хвороби пацієнтів. В цьому випадку знання про те, які поєднання хворів та симптомів зустрічаються найчастіше, дозволяють в майбутньому ставити правильні діагнози.

Визначення

Щоб дати означення асоціативного правила, будемо вважати, що існує база даних, якій містяться записи про всі здійснені покупки в супермаркеті. Кожен запис називається транзакцією і включає дані про набір товарів, куплених одним покупцем за один візит. Таку транзакцію ще називаю ринковим кошиком.

Нехай $I = \{i_1, i_2, i_3, \dots, i_n\}$ – це вся множина товарів з супермаркету, що називаються елементами.

Приклад 11 :

Ідентифікатор	Найменування товару	Ціна
0	Шоколад	30.00
1	Чіпси	12.00
2	Кокоси	10.00
3	Вода	4.00
4	Пиво	14.00
5	Горіхи	15.00

Тобто вся множина елементів (їх загальна кількість рівна n) I буде:

$$I = \{\text{Шоколад, Чіпси, Кокоси, Вода, Пиво, Горіхи}\}$$

Кожна транзакція T описується як: $T = \{i_k \mid i_k \in I\}$. Приклади транзакцій:

$$T_1 = \{\text{Чіпси, Вода, Пиво}\}$$

$$T_2 = \{\text{Кокоси, Вода, Горіхи}\}$$

Набір усіх відомих транзакцій (загальна їх кількість нехай рівна m) позначимо як D : $D = \{T_1, T_2, \dots, T_m\}$.

Нехай для нашого прикладу:

$$D = \left\{ \{Чіпси, Вода, Пиво\}, \{Кокоси, Вода, Горіхи\}, \{Горіхи, Кокоси, Чіпси, Кокоси, Вода\}, \{Кокоси, Горіхи, Кокоси\} \right\}$$

Тоді множину D можемо представити у вигляді:

№ транзакції	Ідентифікатор товару	Найменування товару	Ціна
0	1	Чіпси	12.00
0	3	Вода	4.00
0	4	Пиво	14.00
1	2	Кокоси	10.00
1	3	Вода	4.00
1	5	Горіхи	15.00
2	5	Горіхи	15.00
2	2	Кокоси	10.00
2	1	Чіпси	12.00
2	2	Кокоси	10.00
2	3	Вода	4.00
3	2	Кокоси	10.00
3	5	Горіхи	15.00
3	2	Кокоси	10.00

Множину транзакцій, в яку входить об'єкт i_j позначимо як:

$$D_{i_j} = \{T_r \mid i_j \in T_r, j = 1..n, r = 1..m\} \subseteq D$$

Наприклад, множина транзакцій, в які входить елемент «вода»:

$$D_{\text{вода}} = \left\{ \{Чіпси, Вода, Пиво\}, \{Кокоси, Вода, Горіхи\}, \{Горіхи, Кокоси, Чіпси, Кокоси, Вода\} \right\}$$

Деякий довільний набір елементів позначимо так: $F = \{i_j \mid i_j \in I, j = 1..n\}$. Набір, що складається з k об'єктів називається k -елементним набором. Приклад 2-елементного набору: $F = \{Кокоси, Вода\}$.

Множину транзакцій, в яку входить набір F , позначимо D_F

$$D_F = \{T_r \mid F \subseteq T_r, r = 1..m\} \subseteq D$$

В даному прикладі:

$$D_{\{Кокоси, Вода\}} = \left\{ \{Кокоси, Вода, Горіхи\}, \{Горіхи, Кокоси, Чіпси, Кокоси, Вода\} \right\}$$

Відношення кількості транзакцій, в які входить F , до загальної кількості транзакцій називається *підтримкою (support)* набору F та позначається $\text{Supp}(F)$:

$$\text{Supp}(F) = \frac{|D_F|}{|D|}$$

Можна підтримку рахувати у відсотках (тоді треба помножити на 100%).

Для набору $F = \{\text{Кокоси}, \text{Вода}\}$ підтримка рівна 0.5 або 50%, так як цей набір входить у дві транзакції (з номерами 1 та 2), а всього транзакцій є 4.

При пошуку аналітик може вказати мінімальне значення підтримки для наборів, що його цікавлять – Supp_{\min} .

Набір називається частим, якщо значення його підтримки є більшим за вказане мінімальне значення, задане користувачем: $\text{Supp}(F) > \text{Supp}_{\min}$.

Таким чином, при пошуку асоціативних правил необхідно знайти множину всіх частих наборів:

$$L = \{F \mid \text{Supp}(F) > \text{Supp}_{\min}\}$$

В даному прикладі частими наборами при $\text{Supp}_{\min} = 0.5$ є такі:

$$\begin{aligned} \{\text{Чинси}\} \text{Supp}_{\min} &= 0.5; \\ \{\text{Чинси}, \text{Вода}\} \text{Supp}_{\min} &= 0.5; \\ \{\text{Кокоси}\} \text{Supp}_{\min} &= 0.75; \\ \{\text{Кокоси}, \text{Вода}\} \text{Supp}_{\min} &= 0.5; \\ \{\text{Кокоси}, \text{Вода}, \text{Горіхи}\} \text{Supp}_{\min} &= 0.5; \\ \{\text{Кокоси}, \text{Горіхи}\} \text{Supp}_{\min} &= 0.75; \\ \{\text{Вода}\} \text{Supp}_{\min} &= 0.75; \\ \{\text{Вода}, \text{Горіхи}\} \text{Supp}_{\min} &= 0.5; \\ \{\text{Горіхи}\} \text{Supp}_{\min} &= 0.75. \end{aligned}$$

З іншого боку, важливо не лише знайти часті набори, але виявити правила «якщо....., то...». Наприклад, в даному прикладі можна досліджувати, наскільки правдивим є правило: *якщо «кокоси», то «вода»*. Тобто важливо не просто знати. Що ці ва елементи часто знаходяться в одному наборі, але й вміти прогнозувати, що при покупці «кокосів» ймовірно буде покупка «води» або навпаки.

Розіб'ємо наш досліджуваний набір на два піднабори: F_1 та F_2 . Наприклад, набір $F = \{\text{Кокоси}, \text{Вода}\}$ будемо розглядати як: $F_1 = \{\text{Кокоси}\}$ та $F_2 = \{\text{Вода}\}$, тобто $F = F_1 \cup F_2$. Тоді асоціативним правилом можна назвати імплікацію[2]: $F_1 \Rightarrow F_2$, де $F_1 \subset I, F_2 \subset I, F_1 \cap F_2 = \emptyset$. Правило $F_1 \Rightarrow F_2$ має підтримку:

$$\text{Supp}(F_1 \Rightarrow F_2) = \frac{|D_{F_1 \cup F_2}|}{|D|} \cdot 100\% = \frac{|D_F|}{|D|} \cdot 100\%$$

$$\text{Supp}(F_1 \Rightarrow F_2) = \text{Supp}(F_1 \cup F_2)$$

тобто $\text{Supp}(F_1 \Rightarrow F_2)$ – це відсоток зі всіх транзакцій D , що містять і набір F_1 , і набір F_2 (тобто містять набір $F = F_1 \cup F_2$).

$$\text{Supp}(\{\text{Кокоси}\} \Rightarrow \{\text{Вода}\}) = \frac{2}{4} \cdot 100\% = 50\%$$

Бо, як було вже згадано вище, з чотирьох транзакцій дві містять і «Кокоси» і «Воду».

Достовірністю правила називається ймовірність того, що саме з F_1 випливає F_2 . Правило $F_1 \Rightarrow F_2$ має достовірність (confidence):

$$\text{Conf}(F_1 \Rightarrow F_2) = \frac{\text{Supp}(F_1 \cup F_2)}{\text{Supp}(F_1)}$$

що показує, який відсоток з усіх транзакцій D , що містить F_1 , також містить і F_2 .

$$\text{Conf}(\{\text{Кокоси}\} \Rightarrow \{\text{Вода}\}) = \frac{\text{Supp}(\{\text{Кокоси}\} \cup \{\text{Вода}\})}{\text{Supp}(\{\text{Кокоси}\})} \cdot 100\% = \frac{50}{75} \cdot 100\% \approx 66.7\%$$

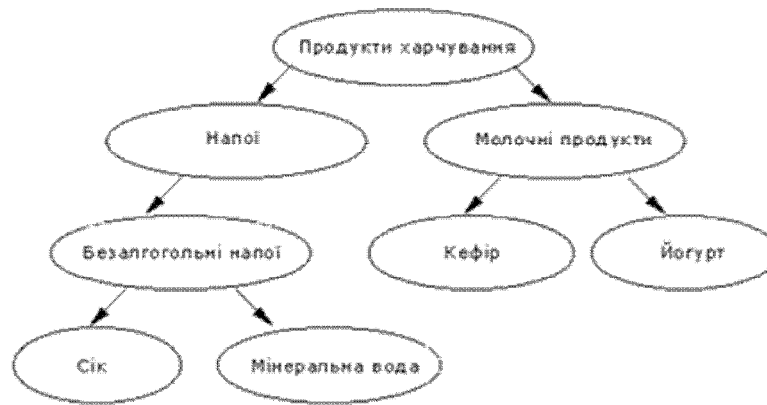
Отже, підтримка правила $\{\text{Кокоси}\} \Rightarrow \{\text{Вода}\}$ рівна 50% (50% зі всіх транзакцій містять і «Кокоси», і «Воду»), а достовірність цього правила рівна 66.7% (66.7% зі всіх транзакцій, що містять «Кокоси», також містять і «Воду»).

Іншими словами, метою аналізу є встановлення наступних залежностей: якщо в транзакції зустрівся деякий набір елементів F_1 , то на підставі цього можна зробити висновок про те, що інший набір елементів F_2 також повинен з'явитися в цій транзакції. Алгоритми пошуку асоціативних правил призначені для знаходження всіх правил $F_1 \Rightarrow F_2$, причому підтримка і достовірність цих правил повинні бути вищими за деякі наперед задані пороги, що називаються відповідно мінімальною підтримкою (Supp_{\min}) та мінімальною достовірністю (Conf_{\min}).

Деякі видозміни асоціативних правил

Узагальнені асоціативні правила (Generalized Association Rules)

При пошуку асоціативних правил вище припускалось, що всі аналізовані елементи є однорідними. Проте, повертаючись до аналізу ринкової корзини, не складе великих труднощів доповнити транзакцію інформацією про те, до якої товарної групи входить товар і побудувати ієрархію товарів. Приведемо приклад такого групування (таксономії) у вигляді ієрархічної моделі.



Нехай дана база транзакцій та відомо, в які групи (таксони) входять елементи. Тоді з даних можна одержувати правила, що пов'язують групи з групами, окремі елементи з групами і т.д. Наприклад, якщо покупець купив товар з групи «Безалкогольні напої», то він купить і товар з групи «Молочні продукти»: правило *Сік ⇒ Молочні продукти*. Ці правила носять назву узагальнених асоціативних правил.

Введення додаткової інформації про угруповання елементів у вигляді ієрархії має свої переваги, зокрема, допомагає встановити асоціативні правила не тільки між окремими елементами, але й між різними рівнями ієрархії (групами). Проте з додаванням до транзакції понять груп збільшується кількість атрибутів і, відповідно, розмірність вхідного простору. Це ускладнює завдання, а також призводить до генерації більшої кількості правил. Для знаходження узагальнених асоціативних правил бажано використання спеціалізованого алгоритму, який усуває вищеписані проблеми.

Групувати елементи можна не тільки по входу до певної товарної групи, але й за іншими характеристиками, наприклад за ціною (дешево, дорого), брендом і т.д.

Чисельні асоціативні правила (Quantitative Association Rules)

При пошуку асоціативних правил все зводилося до того, чи присутній в транзакції елемент чи ні. Тобто, якщо розглядати випадок ринкової корзини, то розглядаємо два стани: куплено товар чи ні. При цьому ігнорується, наприклад, інформація про те, скільки чого було куплено, хто саме купив, тощо. Тобто було розглянуто "булеві" асоціативні правила. Проте можна аналізувати дані різних типів: числові, категоріальні і т.д.

Приклад чисельного асоціативного правила:

Якщо «[Вік: 30-35]» і «[Сімейний стан: одружений]», то «[Місячний дохід: 1000-1500 гривень]».

2. Древа рішень в ІСППР

Метод дерева рішень - це один з методів автоматичного аналізу величезних масивів даних. Перші ідеї створення "дерев рішень" починаються з робіт П.Ховленда і Е.Ханта кінця 50-х років ХХ століття. Проте основоположною роботою, що дала імпульс для розвитку цього напрямку, стала книга Е.Ханта, Дж.Мерина і П.Стоуна "Experiments in Induction", яку було опубліковано в 1966 р.

Область використання методу "дерева рішень" можна об'єднати в три класи:

опис даних: застосування "дерева рішень" дозволяє зберігати інформацію про вибірку даних в компактній і зручній для обробки формі, що містить в собі точні описи об'єктів;

класифікація: застосування "дерева рішень" дозволяє справитися із завданнями класифікації, тобто відношення об'єктів до одного з описаних класів;

регресія: якщо змінна має недостовірні значення, то застосування "дерева рішень" дозволяє визначити залежність цієї цільової змінної від незалежних (вхідних) змінних.

Для прийняття рішення за допомогою "дерева рішень" необхідно виконати такі кроки:

1) оцінити стан ринку вектором чинників $X = (x_1, x_2, x_3, x_4)$ (цей крок виконується користувачем системи);

2) визначити клас зростання прибули шляхом руху вектора $X = (x_1, x_2, x_3, x_4)$ по дереву рішень з верхніх рівнів до нижніх (цей крок виконується системою).

У методиці використовується ієрархічна структурна схема. Для її побудови прийняті відповідні позначення елементів (подій) і логічних операцій.

В основу методу "дерева цілей" покладено підпорядкованість, розгортаємість і ранжування цілей. Дерево цілей з кількісними показниками, що використовуються в якості одного із засобів при прийнятті рішень, і носить назву "дерева рішень".

Головна перевага "дерева рішень" перед іншими методами - можливість пов'язати ставлення цілі з діями, що підлягають реалізації в сьогодні. При побудові багаторівневого "дерева рішень" досягнення мети кожного з рівнів моделі забезпечується комплексом заходів попереднього рівня. Кожен рівень "дерева рішень" повинен займати певне місце в ієрархічній послідовності, складеної на основі дотримання причинно-наслідкових зв'язків.

Етапи побудови "дерева рішень"

Дерево рішень - це графічне зображення послідовності рішень і станів середовища з указівкою відповідних ймовірностей і виграшів для будь-яких комбінацій альтернатив і станів середовища.

Побудова "дерева рішень" виконується "зверху вниз" - від задач більш складних, більш важливих - до завдань менш складним, менш важливим, що вимагає менше часу (коштів, сил, ресурсів) для їх здійснення.

На схемі "дерева рішень" саме верхнє положення займає кінцева мета розв'язання проблеми (кінцевий результат).

Чим складніше можна вирішити завдання, тим більше має бути число рівнів розгляду проблеми і тим більше число завдань, що вирішуються на кожному рівні.

Для кожного "дерева рішень" будується матриця. Часто вводяться коефіцієнти взаємної корисності рішень, одержувані опитуванням експертів. Вони показують вплив ступеня важливості одних рішень на інші.

Застосування методу "дерева рішень" дозволяє:

визначати шляхи досягнення мети з виконанням кількісної оцінки складності виникають завдань та оцінкою труднощі здійснення того чи іншого варіанту;

поліпшувати якість рішень в умовах невизначеності.

Процес прийняття управлінських рішень за допомогою дерева рішень у загальному випадку припускає виконання п'яти етапів:

Етап 1. Формулювання завдання.

Насамперед необхідно відкинути всі фактори, що не стосуються проблеми, а серед безлічі тих, що залишилися, виділити суттєві і несуттєві. Це дозволить привести опис завдання щодо прийняття управлінського рішення у форму, що піддається аналізу. Повинні бути виконані такі основні процедури: визначення можливостей збору інформації для експериментування і реальних дій;

складання переліку подій, що з певною імовірністю можуть відбутися;

установлення часового порядку розміщення подій, у наслідках яких міститься корисна і доступна інформація, і тих послідовних дій, які можна розпочати.

Етап 2. Побудова "дерева рішень".

Етап 3. Оцінка ймовірностей станів середовища, тобто зіставлення шансів виникнення кожної конкретної події. Слід зазначити, що вказані ймовірності визначаються або на підставі наявної статистики, або експертним шляхом.

Етап 4. Установлення вигравів (чи програшів, як вигравів зі знаком мінус) для кожної можливої комбінації альтернатив (дій) і станів середовища.

Етап 5. Вирішення завдання.

Перш ніж продемонструвати процедуру застосування дерева рішень, введемо ряд визначень. У залежності від ставлення до ризику розв'язання задачі може виконуватися з позицій так званих "об'єктивістів" і "суб'єктивістів".

Безумовним грошовим еквівалентом (БГЕ) гри називається максимальна сума грошей, які гравець готовий заплатити за участь у грі (лотереї), або, що те саме, та мінімальна сума грошей, за яку він готовий відмовитися від гри.

Кожен гравець має свій БГЕ. Гравця, для якого БГЕ збігається з очікуваною грошовою оцінкою (ОГО) гри (з середнім виграшем у грі), тобто виконується рівність

$$БГЕ = ОГО$$

то гравця умовно називають об'єктивістом; якщо рівність (9.1) не виконується, то гравця, для якого БГЕ менше або більше ОГО, називають суб'єктивістом.

Очікувана грошова оцінка розраховується як сума добутків розмірів вигравів на ймовірності цих вигравів, тобто ОГО співпадає з математичним очікуванням випадкової величини X - величини вигравів. Якщо x_1, x_2, \dots, x_n - можливі значення випадкової величини X , а p_1, p_2, \dots, p_n - ймовірності, з якими випадкова величина X приймає можливі значення, тоді ОГО можна знайти за формулою

$$ОГО = \sum_{i=1}^n x_i p_i . \quad (9.2)$$

Слід відмітити, якщо для суб'єктивіста виконується нерівність

$$БГЕ > ОГО , \quad (9.3)$$

то суб'єктивіста вважають схильним до ризику; якщо виконується нерівність

$$БГЕ < ОГО , \quad (9.4)$$

то суб'єктивіста вважають несхильним до ризику

Приклад 1. Проводиться розіграш лотереї, в якій розігрується 100 білетів. З них 5 мають виграш 50 грн., 10 - 10 грн., інші виграшу не мають. Вартість одного лотерейного білету 1 грн. Розрахувати ОГО для виграшу гравця та проаналізувати гравця з позиції БГЕ.

Розв'язання. Для того, щоб розрахувати ОГО побудуємо випадкову величину X - величину виграшу в лотереї. Наприклад, якщо гравець придбає білет, і він не виграє, то гравець витратить 1 грн., тому його виграш буде дорівнювати втраченій 1 грн., тобто $x_1 = -1$. З аналогічних міркувань отримаємо, що випадкова величина X буде мати такі можливі значення: $x_1 = -1$, $x_2 = 9$, $x_3 = 49$. За класичним означенням ймовірності знайдемо ймовірності можливих значень: $p_1 = 0,85$, $p_2 = 0,1$, $p_3 = 0,05$. Таким чином маємо закон розподілу випадкової величини X , що наведено в табл.

Таблиця -. Закон розподілу виграшу в лотереї

X	-1	9	49
p_i	0,85	0,1	0,05

За формулою (9.2) розрахуємо ОГО

$$OGO = -1 \cdot 0,85 + 9 \cdot 0,1 + 49 \cdot 0,05 = 2,5 \text{ (грн.)}$$

Один індивід пошкодує і 1 грн. за право участі в такій лотереї, тобто просто не купить лотерейний білет, інший готовий заплатити за лотерейні білети 5 грн., а третій заплатить навіть 30 грн. за можливість одержати 50 грн. (наприклад, коли ситуація складається так, що тільки маючи 50 грн., гравець може досягти своєї мети, тому можлива втрата останніх коштів, а в нього їх рівно 30 грн., не змінює для нього ситуації).

Якщо БГЕ гри дорівнює 2,5 грн., то гравця можна назвати об'єктивістом; якщо БГЕ гри менше ніж 2,5 грн., то гравець є суб'єктивістом, який не схильний до ризику; якщо БГЕ гри більше ніж 2,5 грн., то гравець є суб'єктивістом, який схильний до ризику.

9.3.2. Процедура прийняття господарського рішення за допомогою "дерева рішень"

Розглянемо процедуру прийняття управлінського рішення на прикладі наступної задачі.

Приклад 9.2. Керівництво компанії вирішує, чи створювати для випуску нової продукції велике підприємство, мале підприємство чи продати патент іншій фірмі. Розмір виграшу, який компанія може одержати, залежить від сприятливого чи несприятливого стану ринку (табл. 9.4).

За допомогою дерева рішень надати рекомендації щодо прийняти управлінського рішення.

Розв'язання. На основі даної таблиці виграшів (втрат) можна побудувати "дерево рішень" (рис. 9.8).

Ймовірність сприятливого і несприятливого станів економічного середовища дорівнює 0,5. Процедура прийняття управлінського рішення полягає в обчисленні для кожної вершини дерева очікуваних грошових оцінок, відкиданні безперспективних галузей і виборі галузей, яким відповідає максимальне значення ОГО.

Таблиця 9.4. Вихідні умови прикладу 9.2

№ п/п	Дії компанії	Виграш, при стані економічного середовища, у.о.	
		Сприятливому	Несприятливому
1	Будівництво великого підприємства	200 000	-180000
2	Будівництво малого підприємства	100000	-20 000
3	Продаж патенту	10000	10000

За формулою (9.2) розрахуємо середній очікуваний виграш, тобто ОГО для вершини 1:

$$OGO = 200000 \cdot 0,5 + (-180000) \cdot 0,5 = 10000 \text{ (у.о.)};$$

для вершини 2:

$$OGO = 100000 \cdot 0,5 + (-20000) \cdot 0,5 = 40000 \text{ (у.о.)};$$

для вершини 3:

$$OGO = 10000 \cdot 0,5 + 10000 \cdot 0,5 = 10000 \text{ (у.о.)}$$



Рисунок 2 - Дерево рішень для прикладу 2

Порівнявши розраховані ОГО, можна зробити такий висновок.

Найбільш доцільно вибрати стратегію 2, тобто будувати мале підприємство, а галузі (стратегії) 1 і 3 дерева рішення можна відкинути. ОГО найкращого рішення дорівнює 40000 у.о. Слід зазначити, що наявність стану з ймовірностями 50% (або 0,5) невдачі і 50% удачі на практиці часто означає, що справжні ймовірності гравцю швидше за все невідомі і він приймає таку гіпотезу (так зване припущення - п'ятдесят на п'ятдесят).

Лекція № 6

Тема лекції: Еволюційні технології та генетичні алгоритми в ІСППР.

План лекції

- 1 Системы мобильных агентов
2. Использование мобильных агентов для анализа данных
3. Система анализа распределенных данных

Ключові слова

штучний інтелект; експертна система; обробка природною мовою; системи візуалізації; роботи; програмні агенти; теорія багатоагентних систем; розподілених.

Література

1. Нвана Х. Программные агенты: Обзор. Knowledge Engineering Review, Vol.11, No.3, 205—244, Cambridge University Press
2. Шермер Б. Программные агенты, наблюдение и право на неприкосновенность частной жизни: законодательная база для наблюдения с помощью агентов. Leiden University Press, 2007, p.140.
3. Вулдридж М., Дженнингс Н. Интеллектуальные агенты: теория и практика. Knowledge Eng. Rev., vol. 10(2), pp. 115—152, 1995
4. Стюарт Рассел, Питер Норвиг. Искусственный интеллект: Современный подход. Prentice Hall, 2009, ISBN 978-0136042594
6. Стефен Хагг, Управление информационными системами в информационный век, 2006, p 224—228.
7. Серенко А., Детлор Б.. Интеллектуальные агенты как инновация, Artificial Intelligence & Society, 18(4), 364—381
8. Серенко А., Руби У., Кокосила М.. Незапланированные эффекты интеллектуальных агентов при использовании в Интернете: Социально-информационный подход. Artificial Intelligence & Society, 21(1-2), 141—166.

Зміст лекції

1. Системы мобильных агентов

Исследования в области агентных технологий начали проводиться в 70-х гг. и были связаны с работами по искусственному интеллекту. Выделяют два основных этапа в истории исследования агентов. Первый этап начался около 1977 г. и был связан с исследованиями в области распределенного искусственного интеллекта. Второй этап начался в начале 90-х гг. В это время основное внимание было сосредоточено на агентах, выполняющих некоторые действия от имени человека.

В настоящее время вопрос об определении термина "агент" находится в стадии интенсивного обсуждения. Существует следующее обобщенное определение:

Агентом называется система, находящаяся в некоторой среде и являющаяся ее частью. Агент воздействует на среду при выполнении собственных задач и сам подвергается воздействию с ее стороны. Таким образом, изменения, произведенные агентом в среде, отражаются на нем самом в будущем.

Агенты могут обладать следующими свойствами:

- автономность (autonomous) — способность выполнять поставленную задачу без необходимости вмешательства извне;

- реактивность (reactivity) — способность воспринимать изменение среды и предпринимать ответные действия;
- целенаправленность (goal oriented)— способность выполнять поставленные перед ним задачи;
- устойчивость (persistent)— способность восстанавливать свое состояние после аварийного завершения;
- общительность (communicative) — возможность взаимодействовать с другими элементами среды;
- адаптивность (adaptive) — способность изменять свое поведение в зависимости от накопленного опыта и текущей обстановки;
- мобильность (mobility) — возможность перемещаться в среде;
- гибкость (flexible) — возможность изменять собственное поведение.

Совокупность нескольких агентов, работающих совместно, а следовательно обладающих свойством общительности, называют *многоагентными системами*.

Для работы многоагентная система должна включать в себя *вспомогательные службы* (например, поисковую службу).

Для работы агентов, обладающих свойствами мобильности (мобильных агентов — МА), на узлах устанавливаются программные компоненты – платформы, обеспечивающие среду выполнения и интерфейс между агентами и компьютером. МА в процессе своего выполнения перемещаются между узлами сети, взаимодействуют с их ресурсами и другими агентами, что требует решения дополнительных проблем в организации систем мобильных агентов (СМА).

СМА — это распределенное приложение, обеспечивающее работу МА.

Стандарты многоагентных систем

В настоящее время существуют десятки систем, использующих МА. Для их совместимости разрабатываются спецификации и стандарты.

В настоящее время существует два основных стандарта, разработанные в ассоциациях

OMG (Object Management Group) и
FIPA (Foundation for Intelligent Physical Agents).

Ассоциация OMG разработала стандарт MASIF (Mobile Agent System Interoperability Facilities). В нем она уделила внимание стандартизации следующих проблем технологии МА:

- *управление агентами*. Унифицируется синтаксис и правила выполнения операций управления жизненным циклом МА: создание агентов, приостановка и возобновление их выполнения, перемещение агента и завершение его работы;

- *идентификация агентов*. Унифицируется синтаксис имен агентов. Введение подобного синтаксиса позволяет идентифицировать агента в системе;

- *типизация и адресация агентской платформы*. Унифицируется синтаксис описания типа и адреса агентской платформы. Это необходимо для получения агентом информации о типе платформы, предоставляемой ее сервисами, а также для идентификации локальной и удаленной платформ друг друга.

Согласно стандарту MASIF, СМА делится на регионы/ Регион объединяет платформы, обладающие общими полномочиями. Также он включает в себя реестр, содержащий информацию о платформах и агентах, находящихся в этом регионе.

Для совместимости со стандартом MASIF он должен реализовывать интерфейс MAFFinder. В нем определены операции регистрации создания, удаления и перемещения агентов, мест и платформ.

Платформа (в стандарте MASIF называется "агентная система") может создавать, интерпретировать, запускать, перемещать и уничтожать МА.

Для совместимости со стандартом MASIF она должна реализовывать интерфейс MAFAgentSystem. В нем определены операции приема/передачи, создания/уничтожения, прерывания/возобновления агентов. Платформа идентифицируется именем и адресом.

В состав платформы входят как минимум одно место и интерфейс соединения. Место обеспечивает среду выполнения агентов на компьютере. В нем могут одновременно находиться несколько агентов. Интерфейс соединения реализует службу связи, службу имен и службу безопасности.

MASIF поддерживает некоторые сервисы CORBA, но не требует обязательного их использования. Например, служба имен (Naming Service) поддерживает имена CORBA объектов и может использоваться объектами MAFAgentSystem и MAFFinder. Для поддержки жизненного цикла агента может использоваться сервис LifeCycle. Для сериализации и десериализации агентов возможно использование сервиса Externalization. Сервис Security Service может быть использован для авторизации агентов и управления их доступом к сетевым ресурсам.

FIPA — некоммерческая организация, созданная в 1996 г. Ее главной задачей является разработка спецификаций, определяющих взаимодействие агентов.

В них рассматриваются следующие основные темы:

- *управление агентами*. Унифицируется архитектура платформ агентов, которые включают в себя службы маршрутизации сообщений и управления жизненным циклом агентов;

- *язык взаимодействия агентов (ACL)*. Описывается синтаксис языка, предназначенного для взаимодействия агентов разных систем;

- *взаимодействие с неагентскими программами*. Унифицируются способы взаимодействия агентов с программным обеспечением, которое не использует агентную технологию, через "оболочки", включающие специфицированную онтологию и динамический механизм регистрации, и способы взаимодействия с человеком;

- *управление безопасностью агентов*. Выделяются ключевые угрозы безопасности в управлении агентом и описываются возможные средства защиты;

- *управление МА*. Унифицируются операции ("передать" и "запустить") агентской платформы, необходимые для управления МА;

- *служба онтологии*¹. Описывается служба, обеспечивающая правильное понимание запросов, сообщений, терминов в контексте области применения (коммерция, спорт, медицина и т. д.);

- *области применения*. Рассматриваются приложения, использующие агентскую технологию: сетевой помощник, аудио/видеоконференции, помощник планирования путешествий. С их помощью производится тестирование разработанных спецификаций.

Согласно спецификациям FIPA, СМА состоит из платформ, в состав которых входят следующие компоненты (рис. 12.2):

¹ **Онтология** (в информатике) — это попытка всеобъемлющей и детальной формализации некоторой области знаний с помощью концептуальной схемы. Обычно такая схема состоит из структуры данных, содержащей все релевантные классы объектов, их связи и правила (теоремы, ограничения), принятые в этой области. Этот термин в информатике является производным от древнего философского понятия «онтология».

Онтологии используются в процессе программирования как форма представления знаний о реальном мире или его части. Основные сферы применения — моделирование бизнес-процессов, семантическая паутина (англ. Semantic Web), искусственный интеллект.

- *система управления агентом* (AMS — Agent Management System) управляет созданием, удалением, деактивацией, возобновлением, аутоинтефикацией и миграцией агентов, находящихся на платформе. Она обеспечивает сервис "белых страниц", который хранит текущее местонахождение МА;

- *служба каталога* (DF— Directory Facilitator) реализует сервис "желтых страниц", который хранит описание агентов;

- *менеджер безопасности платформы агентов* (APSM — Agent Platform Security Manager) отвечает за осуществление политики безопасности на транспортном уровне и проверку выполнения операций управления агентами;

- *канал связи агентов* (ACC — Agent Communication Channel) использует информацию, предоставляемую системой управления агентов для маршрутизации сообщений между агентами.

Для достижения поставленных целей ассоциации FIPA и OMG объединяют свои усилия. Первым результатом совместного сотрудничества является разработка расширения языка UML для агентов — Agent UML (AUML). Некоторые элементы языка AUML предполагается включить во вторую версию языка UML.

Системы мобильных агентов

В настоящее время существует несколько десятков CMA. Многие из них (Gypsy, JADE, Ajanta, JATLite и др.) *разработаны в университетах с целью исследования этой технологии*. Некоторые системы (такие как ASDK, JAFMAS и др.) *существуют на уровне библиотек*, предоставляя программисту только базовые классы для реализации основных компонентов: агентов, платформ, механизмов взаимодействия и безопасности. *На их основе разрабатываются самостоятельные системы*, например MagNet и E-Commercia.

В последнее время появляются и коммерческие системы, такие как Gossip фирмы Tryllian, Bee-gent и Plangent корпорации Toshiba. К сожалению, техническая документация для них недоступна.

Некоторые проекты, проводимые в рамках исследования технологии МА (ARA в университете Кайзерслаутер, Mole в университете Штутгарта, Odyssey компании General Magic и др), в настоящее время закрыты. Существует ряд систем, использующих МА только для решения собственных задач. К такому типу систем относится система Voyager фирмы ObjectSpace.

Система мобильных агентов JADE.

Система JADE (Java Agent Development Framework) была разработана в Италии фирмой CSELT S.p.A. совместно с университетом города Парма. Система реализована на Java и совместима со стандартом FIPA.

Среду выполнения агентов на узле обеспечивают контейнеры, которые входят в состав распределенной платформы (рис. 7.3). Контейнеры реализованы как RMI-объекты, поэтому для получения информации о действующих в системе контейнерах используется RMI-реестр.

Главной особенностью рассматриваемой системы является централизованная система управления агентами AMS. Система управления агентами предназначена для создания, удаления, деактивации, возобновления и перемещения МА на платформе. Она реализует сервис "белой страницы" для всех агентов находящихся на ней. Сервис хранит информацию о текущем местонахождении агентов и аналогичен региональному реестру в системе Grasshopper.

В соответствии со стандартом FIPA, платформа системы JADE включает в себя службу каталога DF и канал связи агентов ACC. Служба каталога реализует сервис "желтой страницы" и хранит информацию об агентах, действующих в системе. Канал общения агентов управляет маршрутизацией сообщений между агентами, используя информацию от системы управления агентами.

Все описанные ранее компоненты реализованы как стационарные агенты и размещаются на главном контейнере.

Агент может взаимодействовать с другими агентами посредством асинхронной передачи ACL-сообщений. При отправке сообщения указывается ID получателя. Платформа (точнее ACC) сама определяет текущее местонахождение агентов и более подходящий механизм транспортировки. Все получаемые сообщения ставятся в очередь, доступ к которой реализуется агентской платформой. Следовательно, сообщения будут доставлены агенту, даже если он в момент получения запроса находится в состоянии ожидания. Если в процессе взаимодействия один из агентов перемещается, то связь между ними теряется.

Система JADE не предоставляет возможности поиска агента, переместившегося на другую распределенную платформу, следовательно, область взаимодействия агентов ограничена одной платформой. Централизованный сервис "белых страниц" предоставляет возможность агентам получать информацию о состоянии системы в рамках одной платформы.

2. Использование мобильных агентов для анализа данных

Проблемы распределенного анализа данных

Ценность и качество получаемых в результате анализа знаний зависит от объемов анализируемых данных. Чем больший объем подвергается анализу, тем большее количество закономерностей лучшего качества могут быть извлечены. В настоящее время большие объемы данных хранятся распределенно (т.е. на разных вычислительных машинах, территориально удаленных друг от друга и объединенных вычислительной сетью, например, интранет). Кроме того, распределение данных может быть вызвано природой источников информации (например, датчики на транспортной магистрали). В связи с этим возникает необходимость в адаптации алгоритмов и методов интеллектуального анализа для распределенных данных.

Наиболее подходящим способом адаптации представляется агентная технология и, в частности, системы мобильных агентов (СМА). Агентная технология объединяет в себе опыт работ по искусственному интеллекту и созданию распределенных систем.

Агенты-аналитики

Анализ данных в большинстве случаев не ограничивается применением одного алгоритма или решением одной задачи. Как правило, *аналитик последовательно использует несколько аналитических средств с целью получения новых знаний.* Такими средствами являются *методы и алгоритмы анализа Данных.* При этом знания, полученные в результате применения одного алгоритма, являются предпосылкой применения следующего алгоритма.

На основе уже имеющихся у аналитика знаний производится выбор метода анализа, а также способ и условия его применения. В результате применения метода знания аналитика обогащаются, следовательно, можно повторить итерацию на тех же или на новых данных. При этом выбор средства и/или способа его применения может быть другим. *Изменение условий анализа должно привести к получению новых знаний.*

Описанный цикл выполняется до тех пор, пока в результате очередной итерации будут получены уже новые знания.

Полученные знания могут быть применены на практике. В результате такой практической деятельности вносятся изменения в среду, из которой поступают данные.

Описанный процесс можно представить так, как это изображено на рис. 7.4.

Основная идея использования агентной технологии в области интеллектуального анализа данных (ИАД) заключается в том, чтобы *инкапсулировать*² в агенте цикл извлечения знаний из данных. В этом случае агент должен выступать в роли аналитика. В его обязанности будут входить следующие задачи:

- накопление знаний;
- выбор средства анализа и способа его применения на основе имеющихся знаний;
- применение выбранного средства к данным.

Агент должен решать эти задачи самостоятельно или с помощью своего владельца. Для решения этих задач агент должен обладать следующими свойствами:

- *целенаправленность* — целью агента является извлечение максимального количества знаний из данных;
- *общительность* — агент должен иметь возможность обмениваться своими знаниями с другими агентами для более продуктивного анализа;
- *адаптивность* — агент должен использовать полученные в результате анализа и общения с другими агентами знания для новых исследований и применения их к данным;
- *мобильность* — для работы с данными, находящимися на разных узлах, объединенных сетью.

Остальные свойства не являются обязательными для агента, выполняющего ИАД.

Варианты анализа распределенных данных

Многоагентная система характеризуется наличием в ней нескольких агентов и вспомогательных служб. В частности, инфраструктура многоагентной системы должна обеспечивать следующие свойства агентов: *общительность* и *мобильность*. Рассмотрим, как могут быть использованы эти свойства в рамках решения задачи анализа данных.

Общительность позволяет агентам обмениваться между собой сообщениями. Это свойство можно использовать для обмена знаниями между агентами. Модель, полученная в результате применения аналитического средства, может быть передана другому агенту. В результате принимающий агент перейдет в другое состояние, не применяя данное аналитическое средство к тем же данным.

Необходимо учитывать, что такой обмен знаниями может выполняться только между агентами, выполняющими анализ над подобными (одинаковыми, однородными и т. п.) данными. В противном случае такой обмен знаниями может привести к дезинформированию агентов.

Общение и обмен знаниями позволяет выполнять анализ параллельно несколькими агентами. Наиболее эффективен такой подход при анализе данных, хранящихся распределенно. В этом случае однородные данные хранятся на нескольких узлах, объединенных сетью. Распределение необходимо, как правило, из-за большого объема данных и для повышения производительности работы с ними.

² В языках программирования *инкапсуляция* имеет одно из следующих значений, либо их комбинацию:

- языковой механизм ограничения доступа к определённым компонентам объекта;
- языковая конструкция, способствующая объединению данных с методами (или другими функциями), обрабатывающими эти данные.

Инкапсуляция — один из четырёх важнейших механизмов объектно-ориентированного программирования (наряду с абстракцией, полиморфизмом и наследованием).

На каждом узле анализ локальных данных выполняет отдельный агент. Результатами, полученными при анализе, он обменивается с агентами, выполняющими параллельно с ним анализ данных на других узлах.

При анализе распределенных данных эффективно также и другое свойство — *мобильность*. Она позволяет агенту, последовательно перемещаясь между Узлами, выполнять анализ данных, взаимодействуя с ними локально (рис. 7.6). Мобильные агенты наиболее полезны, когда информация о доступных для анализа источниках данных появляется в процессе анализа.

Агенты могут одновременно обладать общительностью и мобильностью. В этом случае анализ распределенных данных может выполняться комбинированным подходом, т.е. агенты могут и обмениваться знаниями, и при необходимости перемещаться на другие узлы.

Методы и алгоритмы Data Mining наиболее эффективны при анализе больших объемов данных. Такие объемы в современных условиях хранятся распределенно в локальных вычислительных сетях. В связи с этим применение многоагентной технологии (в частности таких свойств агентов, как общительность и мобильность) для анализа в распределенных хранилищах должно повысить производительность аналитических систем.

3. Система анализа распределенных данных³

Общий подход к реализации системы

Система анализа распределенных данных представляет собой многоагентную систему. В процессе анализа существующих библиотек для разработки многоагентных систем и библиотек алгоритмов интеллектуального анализа данных были выбраны библиотеки JADE и Xelopes.

Полученная система включает в себя агентов, выполняющих следующие задачи:

- сбор информации о базе данных;
- сбор статистической информации о данных;
- решение одной задачи интеллектуального анализа данных;

- решение интегрированной задачи интеллектуального анализа данных. В разработанной системе в качестве базовых классов мобильных и стационарных агентов были использованы агенты JADE.

Мобильные агенты используются в системе распределенного анализа непосредственно для применения алгоритмов анализа к данным, находящимся на разных машинах, объединенных локальной сетью.

Стационарные агенты, не обладая свойством мобильности, выполняют вспомогательные функции. К таким *функциям* можно отнести следующие:

- доступ к базе данных;
- настройка мобильных агентов;
- отображение результатов работы мобильных агентов; и т. п.

Взаимодействие между агентами обоих видов осуществляется посредством языка взаимодействия агентов ACL, являющегося частью стандарта FIPA.

Мобильные агенты для выполнения анализа могут использовать две стратегии:

³ Многоагентная система анализа данных была разработана в СПбГЭТУ "ЛЭТИ" в рамках проекта "Многоагентная технология интеллектуального анализа данных и извлечения знаний", выполненного по ведомственной научной программе "Развитие научного потенциала высшей школы", 2005

- *последовательный анализ* — агент последовательно перемещается между узлами сети и выполняет анализ на каждом;
- *параллельный анализ* — агент создает клонов, которые, перемещаясь каждый на свой узел, выполняют анализ и возвращают его результаты исходному агенту.

Лекція № 7

Тема лекції: **Нечіткі методи інтелектуального аналізу даних. Класичні технології інтелектуального аналізу даних в ІСППР**

План лекції

- 1 Нечіткі методи інтелектуального аналізу даних**
- 2. Класичні технології інтелектуального аналізу даних в ІСППР**

Ключові слова

нечіткі методи видобування даних, база нечітких правил, нечітка логіка, база знань..

Література

1. Нвана Х. Программные агенты: Обзор. Knowledge Engineering Review, Vol.11, No.3, 205—244, Cambridge University Press
2. Шермер Б. Программные агенты, наблюдение и право на неприкосновенность частной жизни: законодательная база для наблюдения с помощью агентов. Leiden University Press, 2007, p.140.
3. Вулдридж М., Дженнингс Н. Интеллектуальные агенты: теория и практика. Knowledge Eng. Rev., vol. 10(2), pp. 115—152, 1995
4. Стюарт Рассел, Питер Норвиг. Искусственный интеллект: Современный подход. Prentice Hall, 2009, ISBN 978-0136042594
6. Стефен Хааг, Управление информационными системами в информационный век, 2006, p 224—228.
7. Серенко А., Детлор Б. Интеллектуальные агенты как инновация, Artificial Intelligence & Society, 18(4), 364—381
8. Серенко А., Руби У., Кокосила М.. Незапланированные эффекты интеллектуальных агентов при использовании в Интернете: Социально-информационный подход. Artificial Intelligence & Society, 21(1-2), 141—166.

Зміст лекції

1. Нечіткі методи інтелектуального аналізу даних

Бази нечітких знань являють собою сукупність фактів, лінгвістичних змінних та відповідних функцій приналежності (сукупно трактуватимемо їх як знання), якими можна оперувати, та нечітких висловлювань “ЯКЩО–ТО”, що мають назву нечітких продукційних правил виведення. Такі бази знань є цінним джерелом для опису нечітких понять, видобування даних та прийняття різнорідних рішень у різних галузях науки, бізнесу та виробництва, а також є ефективним засобом моделювання у багатьох задачах кібернетики та штучного інтелекту, що мають справу з нечіткостями, серед яких управління технологічними процесами, різного роду діагностики, розпізнавання образів та мови, прогнозування часових рядів тощо. Чіткі та нечіткі бази знань використовуються сьогодні в багатьох напрямках застосування інформаційних технологій: для побудови експертних та інтелектуальних систем, систем дистанційного навчання та контролю знань

тощо. Сьогодні поширені системи підтримки прийняття рішень, які використовують знання, отримані від експертів. Такі знання зберігаються у базах знань, які зазвичай слугують для різномірного інтелектуального аналізу та виявлення (виведення) певних закономірностей. Усе частіше моделювання складних залежностей в економіці, медицині, будівництві та в інших областях здійснюється за допомогою саме нечітких баз знань.

Нечіткі методи інтелектуального аналізу даних

№ з/п	Назва моделі	Основна суть та особливості	Ймовірні шляхи використання
1	2	3	4
1	Нечіткий кластерний аналіз	Формування нечітких кластерів на основі вхідного простору інформації для їхньої подальшої різноманітної інтерпретації. Кластери зазвичай перетинають вхідну інформацію у спільних областях.	<ul style="list-style-type: none"> • Тракткування розпізнаних кластерів як нечітких правил чи їхніх частин • Тракткування розпізнаних кластерів як функцій приналежності • Мінімізація (редукція) правил та їхньої кількості
2	Нечіткі дерева рішень	Побудова дерева рішень як процес виведення моделі правил для вирішення класифікаційних та регресійних задач. Використовують нечіткі критерії розбиття у внутрішніх вузлах.	<ul style="list-style-type: none"> • Виведення моделі правил • Мінімізація (редукція) правил та їхньої кількості шляхом відсікання неважливих піддерев, оптимізації процесу розбиття, оптимізації розмірів дерева
3	Нечіткі асоціативні правила	Знаходження асоціативних залежностей у формі правил. Нечіткість дозволяє вироблення нечітких границь та інтервалів значень, лінгвістичних змінних.	<ul style="list-style-type: none"> • Асоціативний аналіз вмісту бази знань
4	Нейронечіткі мережі	Синергізм нечіткої логіки та штучних нейронних мереж. Кодування нечіткої системи правил у вигляді мережі. Застосування алгоритмів навчання для її корегування.	<ul style="list-style-type: none"> • Навчання функцій приналежності • Виведення та оптимізація нечітких правил та їхньої кількості • Адаптація до змінного зовнішнього середовища
5	Еволюційні нечіткі системи	Синергізм нечіткої логіки та еволюційних алгоритмів. Являють собою оптимізаційно-пошукову технологію. Навчають базу знань завдяки генетичним операторам через еволюцію.	<ul style="list-style-type: none"> • Навчання, оптимізація та адаптація бази знань (набору правил, одиничних нечітких правил, функцій приналежності, лінгвістичних змінних) <ul style="list-style-type: none"> • Пошук у просторі бази знань

2. Класичні технології інтелектуального аналізу даних в ІСППР

1. Теорія систем (системотехніка, Уільям Ешбі, Норберт Вінер). Властивості *складних систем*: унікальність (не існує аналогів), непередбачуваність (неможливо побудувати діючу модель), цілеспрямованість поведінки. *Складні системи* → *розподілені системи* (характеризуються слабкими зв'язками між елементами).
2. Теорія автоматів + теорія ігор. З поєднання цих двох підходів виникли такі напрямки колективи автоматів, ігри автоматів (Поспелов Д.А., Варшавський В.И.).
3. Штучний інтелект → розподілений штучний інтелект.

Від теорії систем запозичена методологія та основні (базові) принципи побудови систем, від теорії автоматів і теорії ігор запозичений математичний апарат, від штучного інтелекту

– ідеї та принципи колективної поведінки.
 1.1.3. Сучасні комп'ютерні системи розвиваються дуже стрімкими темпами. Результат – над велике ускладнення таких систем. Більшість сучасних комп'ютерних систем запрограмовані на вирішення конкретних задач. Якщо перед такою системою виникне нова (непередбачена розробником) задача, то КС буде неспроможна її вирішити, оскільки не буде мати відповідної програми. Способи подолання

1. Створення КС, здатних діяти самостійно (підвищення оперативності управління).
2. Створення КС, які вміють самостійно навчатися (долати нестачу інформації / невизначеність)
3. Створення КС, які здатні діяти узгоджено та цілеспрямовано без централізованого управління.

1.1.4. Основна мета ТКП – створення складних систем здатних самостійно вирішувати задачі без попереднього програмування (інструктажу) людиною за умов нестачі інформації та швидких змін в оточенні системи. Предметом ТКП є принципи і методи побудови таких систем. Основні задачі ТКП:

1. Дослідження співвідношення децентралізованого і централізованого управління
2. Дослідження питань про однорідність та неоднорідність колективної поведінки
3. Дослідження співвідношення співпраці та суперництва в колективі агентів. (співвідношення індивідуальних та колективних вигравів(інтересів))
4. Дослідження колективних моделей реальності(колективне знання)
5. Питання колективного прийняття рішень(голосування)
6. Питання спілкування (інформаційної взаємодії) колективних агентів. Мови спілкування агентів

1.2. Інтелектуальна система (агент).

1.2.1. Агент (*agent*) – це сутність, що діє самостійно, але за чийсь дорученням. "Агентність" (*agency*) – здатність діяти за власною ініціативою (здатність приймати та реалізувати рішення) → "дієвість". **Інтелектуальна система (агент)** – це комп'ютерна система, яка розміщена в деякому середовищі (рис.1) і яка здатна діяти в цьому середовищі автономно (самостійно) для досягнення цілей, які поставлені перед нею її розробником. Послідовність дій агента, спрямованих на досягнення поставленої мети (під впливом середовища), будемо називати його поведінкою.

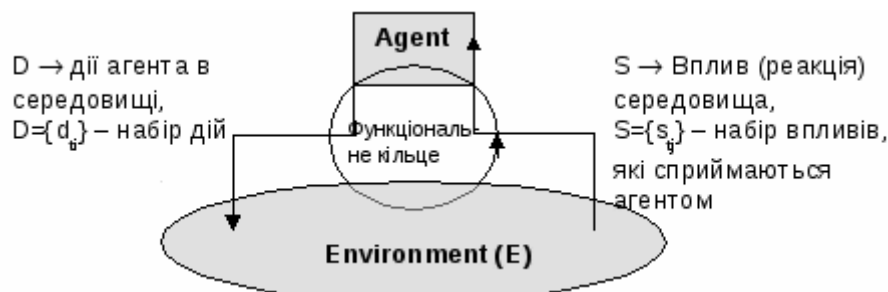


Рис.1. Взаємодія агента з середовищем.

1.2.2. Колектив або багатоагентна система (*multiagent system*) – це набір інтелектуальних систем (агентів), які діють разом для досягнення спільних цілей (рис.2). N – кількість агентів в колективі. Центр/користувач – той хто віддає накази (ставить задачі) колективу.

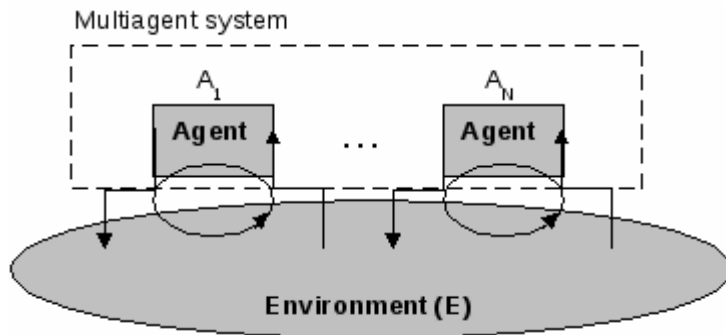


Рис.2. Взаємодія колективу с середовищем.

1.3. Поняття функціональної цілісності. Принцип колективної дії. Елемент (частина) системи – агент. Система – колектив агентів (багатоагентна система).

1.3.1. *Функціональна цілісність* системи передбачає такий тип внутрішньої взаємодії її елементів(частин) при якому властивості цілого(тобто всієї системи) не можна звести до суми властивостей елементів (Уільям Ешбі).

1.3.2. *Емерджентність* (emergence - виникнення) – виникнення (в динаміці) якісно нових властивостей при поєднанні елементів в систему.

1.3.3. *Принцип колективної дії (гіпотеза про перевагу колективної дії над індивідуальною [за певних умов])* – колективними діями можна досягнути більших результатів ніж індивідуальними.

1.4. Поняття елементарного поведінкового акту. Гіпотеза М.Л. Цетліна про простоту.

1.4.1. (з точки зору індивідуальної поведінки агента) Уільям Ешбі: Вирішення задач людиною завжди зводиться до вибору одного варіанту з багатьох можливих. Майже кожен поведінкову задачу (що робити?), яку вирішує людина, можна звести до формули: "Вкажіть один елемент з деякої множини". З віком людина навчається як найкраще вирішувати задачу цілеспрямованого вибору.

1.4.2. Клод Шенон (приклад): "Повідомлення про машину, що вирішує лабіринту задачу". Миша Шенона (знайти найкоротший шлях виходу з лабіринту – метод спроб та помилок): елементарний поведінковий акт: в який коридор лабіринту повертати (Т-лабіринт), складна поведінка (сукупність елементарних поведінкових актів): проходження лабіринту.

1.4.3. **Основна проблема, яку вирішує агент – це проблема вибору: яку з усіх доступних дій вибрати та здійснити (реалізувати) для того, щоб досягнути поставлену перед ним мету. Складний поведінковий акт: це послідовність дій (послідовність елементарних виборів у часі).**

1.4.4. Гіпотеза М.Л. Цетліна про простоту (з точки зору колективної поведінки агентів):

- будь яка складна поведінка складається із сукупності простих поведінкових актів,
- сумісна реалізація та локальна взаємодія простих поведінкових актів призводить до виникнення (емерджентність) складного поведінкового процесу,
- сумісне функціонування колективу простих агентів в складному середовищі здатне забезпечити виконання колективом поставлених перед ним складних задач.

1.5. Наукові напрямки, що складають ТКП.

1. Багатоагентні системи, Multyagent Systems (MAS)
2. Розподілений штучний інтелект, Distributed Artificial Intelligence (DAI)

3. Інтелект рою, Swarm Intelligence
4. Розподілена робототехніка, Distributed Robotics
5. Штучне життя, Artificial Life (A-Life)
6. Колективи автоматів, Cooperative automata,
7. Колективний Інтелект, Collective Intelligence (COIN)