МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Факультет Автоматизації і комп'ютерних систем
Кафедра Інформаційних систем

«До захисту в ЕК»	**«До захисту допущено»**
Декан факультету	Завідувач кафедри
_____ Форсюк А.В.	_____ Чумаченко С.М.
(підпис) (прізвище та ініціали)	(підпис) (прізвище та ініціали)
«___» _____ 2021 р.	«___» _____ 2021 р.

КВАЛІФІКАЦІЙНА РОБОТА
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА

зі спеціальності 122 «Комп'ютерні науки»
(код та назва спеціальності)
освітньо-професійної програми Інформаційні управляючі системи та технології
на тему: Аналіз навчання студентів ВНЗ методами Data Mining

Виконав: здобувач 2 курсу, групи ІС-2-4М
Вакало Ростислав Юрійович
(прізвище, ім'я, по батькові повністю) (підпис)

Керівник Харкянен Олена Валеріївна
(прізвище , ім'я та по батькові повністю) (підпис)

Консультанти _____ _____
(прізвище та ініціали) (підпис)
_____ _____
(прізвище та ініціали) (підпис)
_____ _____
(прізвище та ініціали) (підпис)

Рецензент Сідлецький В.М. _____
(прізвище та ініціали) (підпис)

Засвідчую, що в цій кваліфікаційній роботі немає запозичень із праць інших авторів без відповідних посилань.

Здобувач _____
(підпис)

Київ – 2021 р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) _Автоматизації і комп'ютерних систем_____
Кафедра ____Інформаційних систем_____
Освітній ступінь __магістр_____
Спеціальність_____122 «Комп'ютерні науки»_____
(код і назва)
Освітньо-професійна програма Інформаційні управляючі системи та технології
(назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Інформаційних систем

_____**Чумаченко С. М.**
"____" _____2021 року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

_____Вакала_Ростислава Юрійовича_____
(прізвище, ім'я, по батькові)

1. Тема роботи ____Аналіз навчання студентів ВНЗ методами Data Mining __
_____
_____
керівник роботи__Харкянен Олена Валеріївна, к.т.н., доцент_____,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом закладу вищої освіти від "18" листопада 2020 р. №953-кс2
2. Строк подання здобувачем роботи 25 січня 2021 р.
3. Вихідні дані до роботи Інформація про діяльність Національного університету харчових технологій, інформація про програмне забезпечення ПП "Політек-СОФТ" для вищих навчальних закладів України
_____
_____
4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)
Вступ.
Дослідження сучасних методів аналізу даних.
Дослідження методів Data Mining для аналізу навчання студентів ВНЗ.
Аналіз освітніх даних на основі технології Data Mining.
Висновки.
_____
5. Перелік графічного матеріалу
Додаток А. Схема сховища даних на рівні визначень
Додаток Б. Схема сховища даних на Dimensional рівні
Додаток В. Схема сховища даних в MS SQL Server

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1.	Харкянен О.В., доцент каф. ІС	12.10.20	23.10.20
2.	Харкянен О.В., доцент каф. ІС	10.11.20	30.11.20
3.	Харкянен О.В., доцент каф. ІС	01.12.20	25.12.20

7. Дата видачі завдання _____18 листопада 2020 року_____

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Дослідження діяльності ВНЗ	12.10.20 - 19.10.20	виконано
2	Дослідження методів Data Mining	20.10.20 – 29.10.20	виконано
3	Постановка задачі аналізу успішності навчання студентів ВНЗ	30.10.20 – 11.11.20	виконано
4	Дослідження діяльності НУХТ	12.11.20 – 26.11.20	виконано
5	Реалізація аналізу успішності навчання засобами Data Mining	27.11.20 – 14.01.21	виконано
6	Оформлення роботи	15.01.21 - 20.01.21	виконано
7	Підготовка автореферату	21.01.21 - 27.01.21	виконано
8	Підготовка презентації та доповіді	28.01.21 - 09.02.21	виконано

Здобувач _____ _____Вакало Р.Ю._____
(підпис) (прізвище та ініціали)

Керівник роботи _____ _____Харкянен О.В.___
(підпис) (прізвище та ініціали)

Анотація

Кваліфікаційна робота на здобуття освітнього ступеня магістра «Аналіз навчання студентів ВНЗ методами Data Mining», розроблена Вакало Р.Ю. складається з 79 сторінок, 3 розділів, 23 рисунків, 1 таблиці, 3 додатків та 15 літературних джерел.

В кваліфікаційній роботі було досліджено та проаналізовано методи Data Mining для задач аналізу навчання студентів ВНЗ. Визначено задачі інформаційної підтримки діяльності ВНЗ методами інтелектуального аналізу даних.

Досліджено та адаптовано методи Data Mining для вирішення задач аналізу навчання студентів ВНЗ.

Здійснено аналіз успішності навчання студентів ВНЗ на прикладі Національного університету харчових технологій методами Data Mining з використання MS Analysis Studio та клієнта Data Mining MS Excel.

Ключові слова: ІНТЕЛЕКТУАЛЬНА АНАЛІЗ ДАНИХ, МЕТОДИ DATA MINING, ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД.

ANNOTATION

The thesis «The analysis of educating students in higher education institutions through the use of Data Mining methods», accomplished by Vakalo R. Yu. The paper consists of 80 pages, 3 sections, 23 figures, 1 table, 3 appendices and 13 literary sources.

In the master's thesis, Data Mining methods were investigated and analyzed for use in learning analysis. The review of educating students in higher school is carried out.

The objectives of information support of higher education institutions activities through Data Mining that can be used to achieve them are identified.

Data Mining methods are studied and adapted to solve problems of analyzing student learning;

The evaluation of training success was carried out with a help of Data Mining technology.

Keywords: INTELLECTUAL DATA ANALYSIS, HIGHER EDUCATIONAL INSTITUTION, TIME SERIES, FORECASTING, CLUSTER ANALYSIS.

Зміст

Вступ

Актуальність теми.

Проведення ефективної політики та реформ у сфері освіти вимагає застосування нових методів аналізу для підготовки організаційних і управлінських рішень, адекватних сучасним завданням. У даній ситуації інформаційно-аналітичне забезпечення стає одним з головних «сервісів» у вирішенні проблеми модернізації управління якістю освіти.

У зв'язку із зростаючими обсягами статистичної інформації в навчально-виховній та організаційно-управлінській діяльностях ВНЗ, що накопичується в розподілених, розрізнених джерелах даних, і вимогами до аналізу інформації, які постійно змінюються актуальним стає використання методів інтелектуального аналізу даних (Data Mining) для моніторингу навчальної діяльності, аналізу стану системи освіти у ВНЗ, прогнозування її розвитку, тощо [1]. Інтелектуальний аналіз даних – це потужна технологія для аналізу важливої інформації зі сховища даних або інших джерел. Ця технологія аналізу даних використовується для ідентифікації прихованих закономірностей у великому наборі даних. Інтелектуальний аналіз даних успішно використовується в різних областях, включаючи й освітнє середовище.

В даний час методи Data Mining отримали широке поширення в різних сферах діяльності. Дослідженнями в цій області займаються такі вчені, як А.А. Барсегян, М.С. Купріянов, Г. Пятецкій-Шапіро, Х. Ромесбург, Дж. Хан. Проблеми аналізу даних освітнього процесу розглядалися в роботах таких вчених, як Р. Бакер, Л.І. Григор'єв та інші.

Зв’язок роботи з науковими програмами, планами, темами. Наукова робота виконувалася згідно з планами науково-дослідних робіт кафедри інформаційних систем Національного університету харчових технологій.

Об’єктом дослідження є інформаційний ресурс накопичений в процесі діяльності Національного університету харчових технологій.

Предметом дослідження є методи і моделі для аналізу навчання студентів ВНЗ.

Мета й завдання дослідження. Метою магістерської роботи є дослідження та застосування методів Data Mining в задачах інформаційної підтримки діяльності ВНЗ.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

- вивчити діяльність ВНЗ та виділити задачі аналізу успішності навчання, які можна вирішити застосувавши технологію Data Mining;
- дослідити та адаптувати методи Data Mining для вирішення задач аналізу навчання студентів;
- здійснити аналіз успішності навчання застосувавши технологію Data Mining.

Методи дослідження. Проектування структури сховища та вітрин даних було здійснено в ALL Fusion ERWin Data Modeler. Реалізація сховища і вітрин даних здійснена в MS SQL Server. Для аналізу успішності навчання використано MS Analysis Services з надбудовою "Інтелектуальний аналіз" в MS Excel.

Наукова новизна одержаних результатів. Наукова новизна магістерської роботи полягає у пропозиціях щодо використання технології Data Mining для аналізу успішності навчання студентів ВНЗ.

Практичне значення отриманих результатів. Практична цінність роботи полягає у запропонованих способах використання технології інтелектуального аналізу даних для аналізу та прогнозування навчання студентів ВНЗ.

Особистий внесок здобувача.

–Проаналізовано та виділено задачі аналізу успішності навчання, які можна вирішити на основі інформації, наявної у базі даних ВНЗ методами Data Mining.

–Спроектовано структуру сховища даних для збереження інформації для проведення аналізу навчання студентів.

–Здійснено аналіз навчання студентів ВНЗ на основі використання методів Data Mining.

Розділ 1. Дослідження сучасних методів аналізу даних

1.1. Методи Data Mining для аналізу інформації засобами інформаційних технологій

Термін Data Mining (укр. видобуток даних, інтелектуальний аналіз даних) введений Григорієм Пятецким-Шапіро у 1989 році. З його визначенням, Data Mining — це процес виявлення в сирих даних раніше невідомих, нетривіальних, практично корисних і доступних інтерпретації знань, необхідних для прийняття рішень у різних сферах людської діяльності.

На сьогоднішній день існує декілька підходів до побудови моделей даних, а саме:

– статистичний (англ. Statistical Modelling): базується на теорії та зосереджується на перевірці гіпотез;
– на основі машинного навчання (англ. Machine Learning): евристичний, концентрується на поліпшенні роботи агентів;
– обчислювальний (по суті — інтелектуальний аналіз даніх): інтеграція теорії та евристик, сконцентрований на єдиному процесі аналізу даних, включає евристику

Традиційні методи аналізу даних (статистичні методи) і аналітична обробка в реальному часі (Online Analytical Processing, далі — OLAP) в основному орієнтовані на перевірку заздалегідь сформульованих гіпотез (verification-driven data mining) і на «грубий» розвідувальний аналіз, що становить основу оперативної аналітичної обробки даних, у той час як одне з основних положень Data Mining — пошук неочевидних закономірностей. Інструменти Data Mining можуть знаходити такі закономірності і будувати гіпотези про взаємозв'язки самостійно. Оскільки формулювання гіпотези щодо залежностей є найскладнішим завданням, перевага Data Mining в порівнянні з іншими методами аналізу є очевидним[2].

Суть і мету інтелектуального аналізу даних можна охарактеризувати таким чином: це технологія, призначена для пошуку у великих обсягах даних (англ. Big Data) неочевидних, об'єктивних і корисних на практиці закономірностей. Неочевидних — тобто таких, що не виявляються стандартними методами обробки інформації або експертним шляхом. Об'єктивних — тобто таких, будуть повністю відповідати дійсності (на відміну від суб'єктивної експертної думки). Корисних на практиці — тобто таких, яким можна знайти практичне застосування.

Нерідко Data Mining ототожнюють з виявленням знань у базах даних (англ. Knowledge Discovery in Databases), хоча більш правильно вважати Data Mining одним із кроків цього процесу. Зв'язок Data Mining з іншими областями зображений на рисунку 1.1.

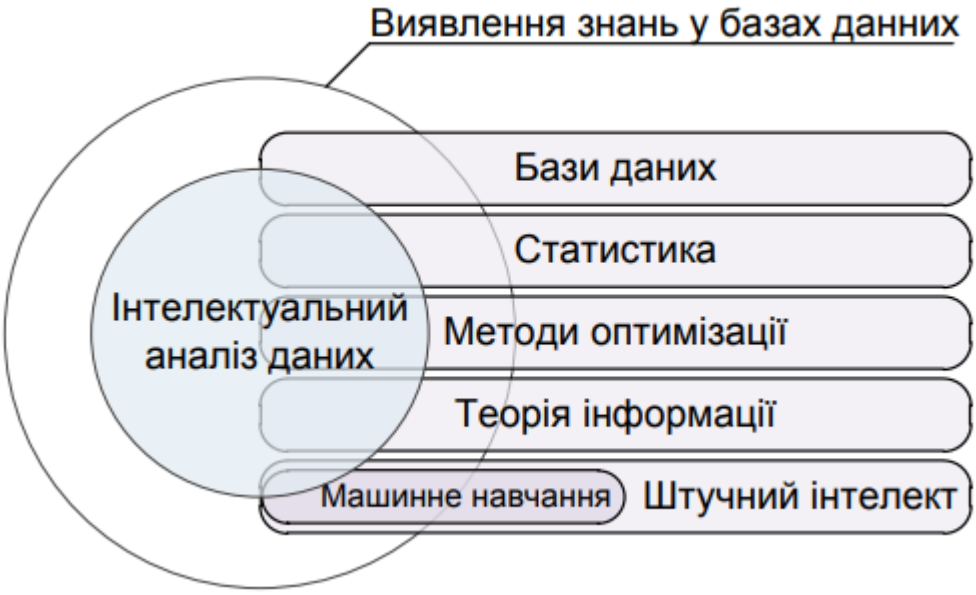


Рис. 1.1. Зв'язок Data Mining з іншими областями

Інтелектуальний аналіз даних — потужний засіб у рекламі, у бізнесі, у економіці тощо. Це не просто область науки, це область життєдіяльності, що є як науковою, оскільки включає в себе наукові дисципліни, так і прикладною, оскільки спеціалісту у сфері інтелектуального аналізу даних необхідні навички програмування та алгоритміки. Крім цього, це в якомусь сенсі культура та

мистецтво, оскільки алгоритми Data Mining вимагають постійно шукати нові шляхи вирішення проблем. Етапи інтелектуального аналізу зображені на рисунку 1.2.

Рис. 1.2. Етапи інтелектуального аналізу

Методи Data Mining

Основна особливість Data Mining — це поєднання широкого математичного інструментарію (від класичного статистичного аналізу до нових кібернетичних методів) і останніх досягнень у сфері інформаційних технологій. У технології Data Mining гармонійно поєдналися строго формалізовані методи та методи неформального аналізу, тобто кількісний та якісний аналіз даних.

Класифікація методів Data Mining

До методів і алгоритмів інтелектуального аналізу даних відносяться наступні: штучні нейронні мережі, дерева рішень, символьні правила, методи найближчого сусіда і k-найближчого сусіда, метод опорних векторів, байєсовські мережі, лінійна регресія, кореляційно-регресійний аналіз; ієрархічні методи кластерного аналізу, неієрархічні методи кластерного аналізу, в тому числі алгоритми k-середніх і k-медіани; методи пошуку асоціативних правил, у тому числі алгоритм Apriori; метод обмеженого перебору, еволюційне програмування і генетичні алгоритми, різноманітні методи візуалізації даних тощо.

Більшість аналітичних методів, що використовуються в технології Data Mining — це відомі математичні алгоритми. Новим в їх застосуванні є можливість їх використання при вирішенні тих чи інших конкретних проблем, обумовлена можливостями, які з'явилися завдяки розвитку технічних і програмних засобів. Слід зазначити, що більшість методів інтелектуального аналізу даних були розроблені в рамках теорії штучного інтелекту.

Розділимо методи інтелектуального аналізу даних на статистичні на кібернетичні. Під статистичними методами маємо на увазі наступні:

– попередній аналіз природи статистичних даних (перевірка гіпотез стаціонарності, нормальності, незалежності, однорідності, оцінка виду функції розподілу, її параметрів і т.п.);

– виявлення зв'язків і закономірностей (лінійний і нелінійний регресійний аналіз, кореляційний аналіз та ін.);

– багатовимірний статистичний аналіз (лінійний і нелінійний дискримінантний аналіз, кластерний аналіз, компонентний аналіз, факторний аналіз та ін.);

– динамічні моделі і прогноз на основі часових рядів. Другий напрямок — це множина підходів, об'єднаних ідеєю комп'ютерної математики та використання теорії штучного інтелекту. До цієї групи відносимо такі методи:

– штучні нейронні мережі (розпізнавання, кластеризація, прогноз);

– еволюційне програмування (в т.ч. алгоритми методу групового обліку аргументів);

– генетичні алгоритми (оптимізація);

– асоціативна пам'ять (пошук аналогів, прототипів);

– нечітка логіка;

– дерева рішень;

– системи обробки експертних знань.

Методи Data Mining також можна класифікувати за задачами. Відповідно до такої класифікації виділяємо дві групи. Перша з них — це підрозділ методів на вирішення задач сегментації (тобто задачі класифікації і кластеризації) і задачі прогнозування. У відповідності з другою класифікацією, методи Data Mining можуть бути спрямовані на отримання описових і прогнозуючих результатів. Описові методи служать для знаходження шаблонів (або зразків), що описують дані, які піддаються інтерпретації з точки зору аналітика. До методів, спрямованим на отримання описових результатів, відносяться ітеративні методи кластерного аналізу, в тому числі: алгоритм k-середніх, k-медіани, ієрархічні методи кластерного аналізу, самоорганізаційні карти Кохонена, методи крос-табличної візуалізації, різні методи візуалізації тощо[2].

Прогнозуючі методи використовують значення одних змінних для передбачення/прогнозування невідомих (пропущених) або майбутніх значень інших (цільових) змінних. До методів, спрямованим на отримання прогнозуючих результатів, відносяться такі методи: нейронні мережі, дерева рішень, лінійна регресія, метод найближчого сусіда, метод опорних векторів та ін.

Класифікація — це задача розбиття множини об'єктів або спостережень на апріорно задані групи, звані класи, всередині кожної з яких вони передбачаються схожими один на одного, мають приблизно однакові властивості і ознаки. При цьому рішення отримується на основі аналізу значень атрибутів.

Класифікація є однією з найважливіших задач інтелектуального аналізу даних. Вона застосовується в маркетингу при оцінці кредитоспроможності позичальників, визначенні лояльності клієнтів, розпізнаванні образів, медичній діагностиці та багатьох інших додатках. Якщо аналітику відомі властивості об'єктів кожного класу, то якщо нове спостереження належить до певного класу, дані властивості автоматично поширюються і на нього.

Застосування нейронних мереж для задач класифікації

Розв’язання задачі класифікації є одним з найважливіших застосувань нейронних мереж.

Штучні нейронні мережі — здатні до навчання системи, що імітують діяльність людського мозку[1].

Незважаючи на велику різноманітність варіантів нейронних мереж, всі вони мають спільні риси. Так, всі вони, так само, як і мозок людини, складаються з великого числа зв'язаних між собою однотипних елементів — нейронів, які імітують нейрони головного мозку. На рисунку 1.3 показана схема нейрону.

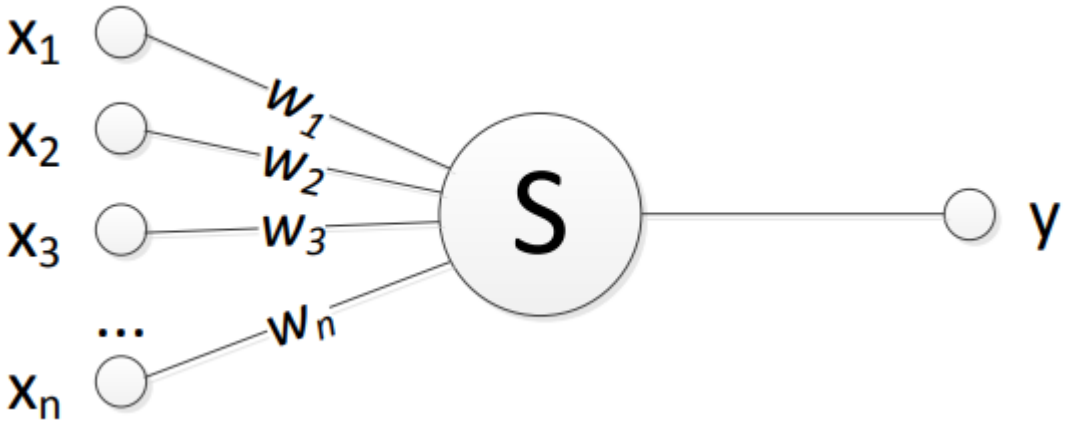


Рис. 1.3. Схема штучного нейрону

З рисунку видно, що штучний нейрон, так само, як і живий, складається із синапсів, що пов'язують входи нейрону з ядром; ядра нейрона, яке здійснює обробку вхідних сигналів і аксона, що пов'язує нейрон з нейронами наступного шару. Кожен синапс має вагу, яка визначає, наскільки відповідний вхід нейрону впливає на його стан. Стан нейрону визначається за формулою:

$$S = \sum_{i=1}^{n} x_i w_i \tag{1.1}$$

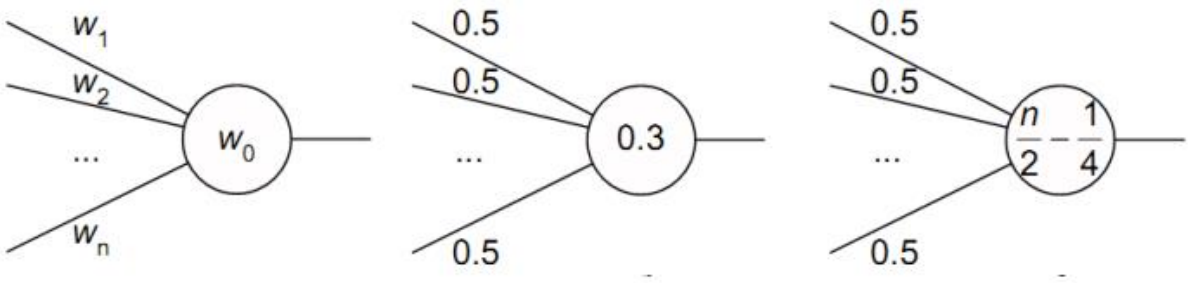
де n— число входів нейрона, x_i — значення і-го входу нейрона, w_i — вага і-го синапса.

Далі визначається значення аксона за формулою:

$$Y = f(S), \tag{1.2}$$

де f— деяка функція, що називається активаційною.

У випадку лінійної нейронної мережі, перцептрон підраховує S та порівнює його із заданим значенням — порогом активації (англ. threshold) видає 1, якщо $S > w_0$ (рис. 1.3.а). Будь-яка булева функція може бути представлена у вигляді побудованної з перцептронів штучної мережі глибини 2. Приклад диз'юнкції та кон'юнкції подано на рисунку 1.4.б-в.



а) загальна схема б) диз'юнкція в) кон'юнкція

Рис. 1.4 Приклади перцептронів

Найчастіше у якості активаційної функції використовується так звана сигмоїда, що має наступний вигляд:

$$f(x) = \frac{1}{1 + e^{-ax}} \tag{1.3}$$

Основна перевага цієї функції в тому, що вона диференційована на всій області визначення і має дуже просту похідну:

$$f`(x) = af(x)(1 - f(X)) \tag{1.4}$$

При зменшенні параметра сигмоїда стає більш пологою, вироджуючись в горизонтальну лінію на рівні 0,5 при $a = 0$. При збільшенні сигмоїда все більше наближується до функції одиничного стрибка.

Алгоритм побудови класифікатора на основі нейронних мереж

Алгоритм побудови класифікатора на основі нейронних мереж має наступні кроки.

Для того, щоб побудувати якісний класифікатор, необхідно мати якісні дані. Жоден з методів побудови класифікаторів, заснований на нейронних мережах або статистичний, ніколи не дасть класифікатор потрібної якості, якщо наявний набір прикладів не буде достатньо повним і представницьким для задачі, з якою доведеться працювати системі.

1. Робота з даними:

1.1. скласти базу даних із прикладів, характерних для даної задачі;

1.2. розбити всю сукупність даних на дві множини: навчальну і тестову (або на 3: навчальну, тестову і підтверджуючу).

2. Попередня обробка:

2.1. вибрати систему ознак, характерних для даної задачі, і перетворити дані відповідним чином для подачі на вхід мережі (нормування, стандартизація

і т.д.). В результаті бажано отримати лінійно відокремлюваний простір множини зразків;

2.2. вибрати систему кодування вихідних значень (класичне кодування, 2 на 2 кодування і т.д.)

3. Конструювання, навчання та оцінка якості мережі:

3.1. вибрати топологію мережі, тобто кількість шарів, число нейронів у шарах і т.д.;

3.2. вибрати функцію активації нейронів (наприклад «сигмоїду»);

3.3. вибрати алгоритм навчання мережі;

3.4. оцінити якість роботи мережі на основі підтверджуючої множини або іншим критерієм, оптимізувати архітектуру (зменшення ваг, проріджування простору ознак тощо);

3.5. зупиниться на варіанті мережі, що забезпечує найкращу спроможність до узагальнення та оцінити якість роботи по тестовій множині.

4. Використання та діагностика:

4.1. з'ясувати степінь впливу різних факторів на прийняте рішення (евристичний підхід);

4.2. переконатися, що мережа дає необхідну точність класифікації (кількість неправильно розпізнаних прикладів нечисленна);

4.3. при необхідності повернуться на етап 2, змінивши спосіб представлення зразків або змінивши базу даних;

4.4. практично використовувати мережу для вирішення поставленої задачі.

Для того, щоб побудувати якісний класифікатор, необхідно мати якісні дані. Жоден з методів побудови класифікаторів, заснований на нейронних мережах або статистичний, ніколи не дасть класифікатор потрібної якості, якщо наявний набір прикладів не буде достатньо повним і представницьким для задачі, з якою доведеться працювати системі[2].

Дерева прийняття рішень

Дерева прийняття рішень — це спосіб представлення правил в ієрархічній структурі, де кожному об'єкту відповідає єдиний вузол, що дає результуюче рішення. Під правилом розуміється логічна конструкція, представлена у вигляді «якщо ... то ...» (рис. 1.5).

Дерева прийняття рішень (дерева класифікацій, регресійні дерева) — один з методів автоматичного аналізу даних. Перші ідеї створення дерев рішень сходять до робіт Ховленда (Hoveland) і Ханта (Hunt) кінця 50-х років XX століття. Однак, основоположною роботою, що дала імпульс для розвитку цього напрямку, стала книга Ханта (Hunt, EB), Мерін (Marin J.) і Стоуна (Stone, PJ) «Experiments in Induction», що побачила світ у 1966 р.

Ситуації, в яких варто застосовувати дерева прийняття рішень, зазвичай виглядають наступним чином: є множіна випадків, кожен з яких описується деяким кінцевим набором дискретних атрибутів, і в кожному випадку дано значення деякої (невідомої) булевої функції, що залежить від цих атрибутів. Задача: створити економічну конструкцію, яка б описувала цю функцію і дозволяла класифікувати нові дані.

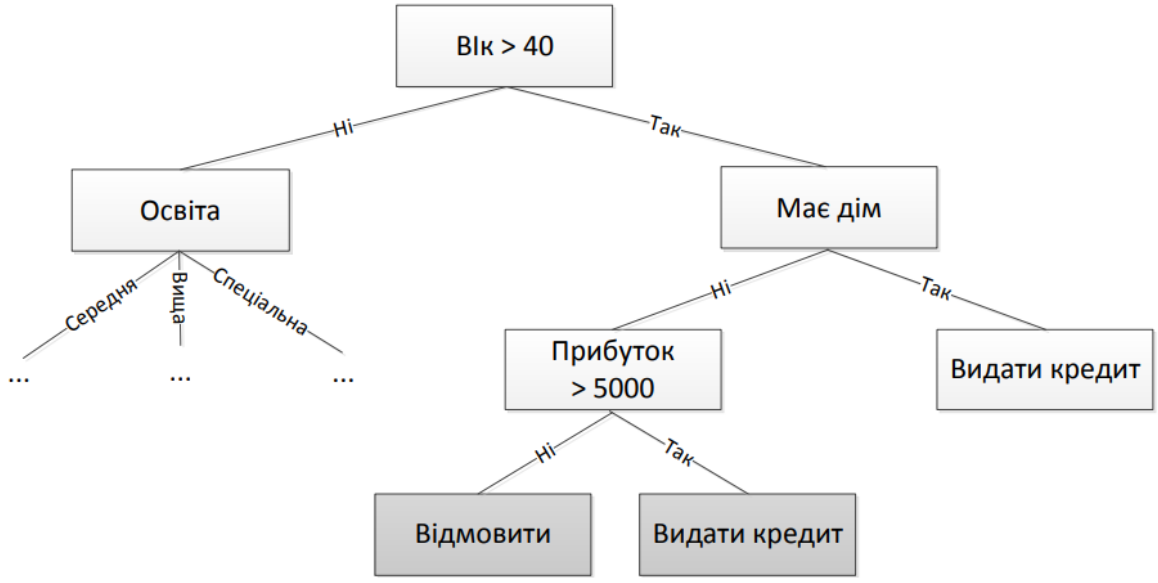


Рис.1.5 Приклад дерева прийняття рішень

Область застосування дерев рішень у даний час широка, але всі задачі, які вирішуються цим апаратом можуть бути об'єднані в наступні три класи:

– опис даних, при якому дерева рішень дозволяють зберігати інформацію про дані в компактній формі, у 42 вигляді дерева рішень, що містить точний опис об'єктів;

– класифікація, з якою дерева рішень відмінно справляються; проте цільова змінна для даної задачі має мати дискретні значення;

– Регресія; якщо цільова змінна має безперервні значення, дерева рішень дозволяють встановити залежність цільової змінної від незалежних (вхідних) змінних; наприклад, до цього класу відносяться завдання чисельного прогнозування (передбачення значень цільової змінної) [2].

Етапи побудови дерев рішень

При побудові дерев рішень особлива увага приділяється наступним питанням: вибір критерію атрибуту, за яким буде відбуватися розбиття, зупинки навчання і відсікання гілок. Розглянемо всі ці питання по порядку.

Яким чином слід вибрати ознаки. Для побудови дерева прийняття рішення на кожному внутрішньому вузлі необхідно знайти таку умову (перевірку), яка б розбивала множину, асоційовану з цим вузлом, на підмножини якомога більш оптимальним способом. В якості такої перевірки повинен бути вибраний один з атрибутів. Загальне правило для вибору атрибуту можна сформулювати наступним чином: обраний атрибут повинен розбити множину так, щоб результуючі підмножини складалися з об'єктів, що належать до одного класу, або були максимально наближені до цього, тобто кількість об'єктів з інших класів («домішок») у кожному з цих множин була якомога меншою.

Інтуїтивно зрозуміло, що для отримання оптимального дерева прийняття рішень, необхідно на кожному кроці обирати атрибути, що «найкраще» характеризують цільову функцію. Ця вимога формалізується за допомогою поняття ентропії. Припустимо, що є деяка множина A з n елементів, m з яких

мають деяку властивість S. Тоді ентропія множини A по відношенню до властивості S — це:

$$H(A,S) = \frac{m}{n}\log_2\frac{m}{n} - \frac{n-m}{n}\log_2\frac{n-m}{n} \quad (1.5)$$

Тобто ентропія залежить від пропорції, в якій розділяється множина. По мірі зростання цієї пропорції від 0 до $1/2$ ентропія також зростає, а після $1/2$ — симетрично спадає.

Якщо властивість S не бінарна, а може приймати s різних значень, кожне з яких реалізується в mi випадках, то ентропія узагальнюється таким чином:

$$H(A,S) - -\sum_{i=1}^{S}\frac{m_i}{n}\log\frac{m_i}{n} \quad (1.6)$$

Поняття ентропії тісно пов'язане з теорією інформації. Грубо кажучи, ентропія — це середня кількість біт, які необхідні, щоб закодувати атрибут S елементу множини A. Якщо ймовірність появи S рівна $1/2$, то ентропія рівна 1, і необхідний повноцінний біт, а якщо S з'являється нерівномірно, то можна закодувати послідовність елементів A ефективніше.

Таким чином, при виборі атрибуту для класифікації, треба обирати його так щоб після класифікації ентропія стала якомога меншою. Ентропія при цьому буде різною у різних нащадків, і загальну суму треба порахувати з урахуванням того, скільки результатів залишилось у розгляді у кожного з нащадків[2].

Загальноприйнятим в теорії дерев прийняття рішень є теоретико-інформаційний критерій вибору відповідного атрибуту. Припустимо, що множина A елементів, деякі з яких мають властивість S, класифіковано за

допомогою атрибута Q, що має q можливих значень. Тоді приріст інформації (англ. Information Gain) визначається наступним чином:

$$Gain(A,Q) = H(A,S) - \sum_{i=1}^{q} \frac{|A_i|}{A} \cdot H(A,S) \tag{1.7}$$

де A_i — множина елементів A, на яких атрибут Q приймає значення i.

На кожному кроці жадібний алгоритм має обирати той атрибут, для якого приріст інформації максимальний.

Алгоритм найближчого сусіда

Людина, зустрічаючись з новою задачею, використовує свій життєвий досвід, згадує аналогічні ситуації, які колись з ними відбувалися. Про властивості нового об'єкта ми судимо, покладаючись на схожі знайомі спостереження. Наприклад, зустрівши іноземця на вулиці, ми можемо здогадатися про його походження по акценту, жестах і зовнішності. Для цього необхідно згадати найбільш схожу на нього людину, походження якої відомо.

Подібність об'єктів покладена в основу алгоритму k-найближчих сусідів (англ. k-nearest neighbor algorithm, KNN). Алгоритм здатний виділити серед всіх спостережень k-відомих об'єктів (k-найближчих сусідів), схожих на новий невідомий раніше об'єкт. На основі класів найближчих сусідів виноситься рішення щодо нового об'єкта. Важливим завданням даного алгоритму є підбір коефіцієнта k — кількості записів, які будуть вважатися схожими.

Алгоритм KNN широко застосовується в Data Mining. Формулюється алгоритм наступним чином. Нехай є n спостережень, кожному з яких відповідає запис у таблиці. Усі записи належать до будь-якого класу. Необхідно визначити клас для нового запису[2].

Суть алгоритму

На першому кроці алгоритму слід задати число k — кількість найближчих сусідів. Якщо прийняти k = 1, то алгоритм втратить узагальнюючу здатність (тобто здатність видавати правильний результат для даних, що не зустрічалися

раніше в алгоритмі), оскільки новому запису буде присвоєний клас, близький до нього. Якщо встановити занадто велике значення, то багато локальних особливостей не буде виявлено.

На другому кроці знаходяться k записів з мінімальною відстанню до вектора ознак нового об'єкта (пошук сусідів). Функція для розрахунку відстані повинна відповідати наступним правилам:

- $d(x, y) \geq 0$,
- $d(x, y) = 0$ тоді і тільки тоді, коли $x = y$;
- $d(x, y) = d(y, x)$;
- $d(x, z) \leq d(x, y) + d(y, z)$, за умови, що точки x, y, z лежать на одній прямій.

Для впорядкованих значень атрибутів знаходиться Евклідова відстань:

$$D_E = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{1.7}$$

де n — кількість атрибутів.

Для строкових змінних, які не можуть бути впорядковані, може бути застосована функція відмінності, яка задається наступним чином:

$$d(x, y) = \begin{cases} 0, & x = y \\ 1, & x \neq y \end{cases}. \tag{1.8}$$

Часто перед розрахунком відстані необхідна нормалізація.

Мінімаксна нормалізація:

$$X^x = \frac{X - \min(X)}{\max(X) - \min(X)}. \tag{1.9}$$

Нормалізація за допомогою стандартного відхилення:

$$X^x = \frac{X - X_{cp}}{\sigma_x} \tag{1.10}$$

де σ_x — стандартне відхилення, X_{cp} — середнє значення.

При знаходженні відстані іноді враховують значущість атрибутів. Вона визначається експертом або аналітиком суб'єктивно, покладаючись на власний досвід. У такому випадку при знаходженні відстані кожен і-ий квадрат різниці в сумі множиться на коефіцієнт Z_i. Наприклад, якщо атрибут A в три рази важливіше атрибута B $(Z_A = 3, Z_B = 1)$, то відстань буде знаходитися наступним чином:

$$D_E = \sqrt{3 \cdot (x_A - y_A)^2 + (x_B - y_{B)}{}^2} \tag{1.11}$$

Подібний прийом називають розтягуванням осей (англ. Stretching the Axes), що дозволяє знизити помилку класифікації.

Слід зазначити, що якщо для запису A найближчим сусідом є B, то це не означає, що B — найближчий сусід A. Дана ситуація представлена на рисунку 1.6 : при $k = 1$ найближчій для точки B буде точка A, а для A — X; при збільшенні коефіцієнта до $k = 7$, точка B так само не буде входити в число сусідів.

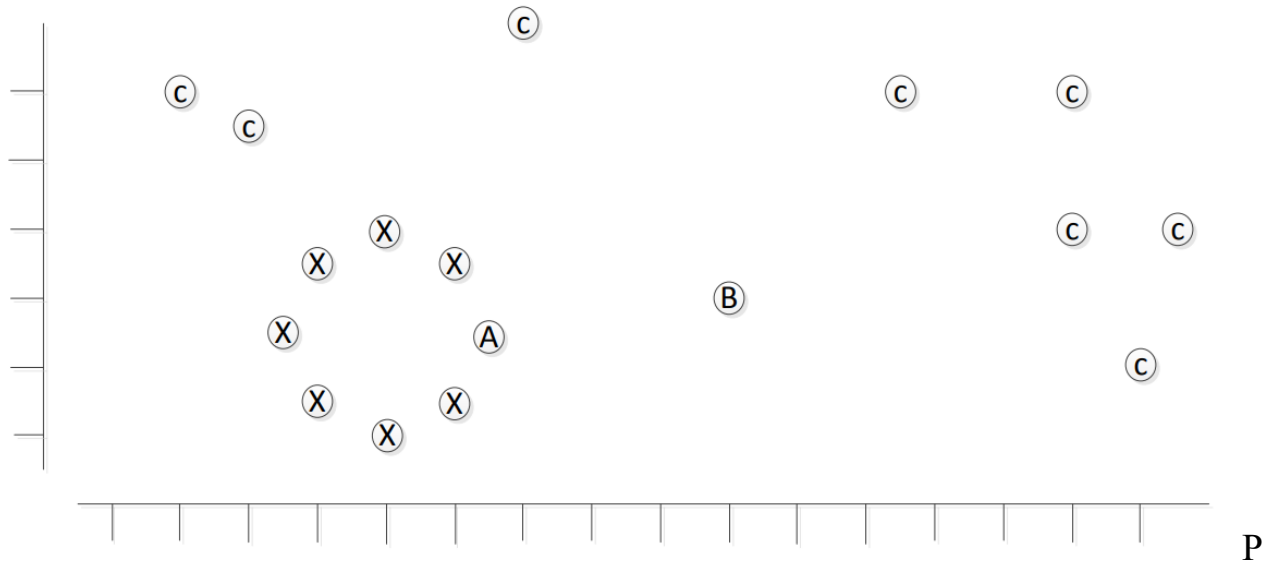


Рис. 1.6 Найближчі сусіди A і B

На наступному кроці, коли знайдені записи, найбільш схожі на новий, необхідно вирішити, як вони впливають на клас нового запису. Для цього використовується функція поєднання (англ. Combination Function). Одним з основних варіантів такої функції є просте незважене голосування (англ. Simple Unweighted Voting).

Після визначення числа k, визначається, скільки записів буде мати право голосу при визначенні класу. Потім виявляються ці записи, відстань від яких до нового виявилося мінімальною. Тепер можна приступити до простого незваженого голосування.

Відстань від кожного запису при голосуванні більше не грає ролі. Всі мають рівні права у визначенні класу. Запис віддає перевагу тому класу, до якого він належить, а той, який набере найбільшу кількість голосів, присвоюється новому запису.

Якщо декілька класів набрали рівну кількість голосів здійснюється зважене голосування (англ. Weighted Voting). У такій ситуації враховується також і відстань до нового запису. Чим менше відстань, тим більш значущий внесок вносить голос. Голоси за клас знаходяться за наступною формулою:

$$votes(class) = \sum_{i=1}^{n} \frac{1}{d^2(X, Y_i)} \quad (1.12)$$

де $d^2(X,Y)$— квадрат відстаней від відомого запису Y_i до нового X, n — кількість відомих записів, для яких розраховуються голоси, class — назва класу.

Клас, який набрав найбільшу кількість голосів, присуджується новому запису. При цьому ймовірність того, що кілька класів наберуть однакові голоси, набагато нижча. Цілком очевидно, що при $k = 1$ новому запису присвоюється клас найближчого сусіда.

Пошук асоціативних правил.

Асоціативні правила дозволяють знаходити закономірності між пов'язаними подіями. Прикладом такого правила, служить твердження, що покупець, що придбає «Хліб», придбає і «Молоко» з імовірністю 75%. Перший алгоритм пошуку асоціативних правил, що називався AIS був розроблений в 1993 році співробітниками дослідницького центру IBM Almaden. З цією піонерської роботи зріс інтерес до асоціативних правил; на середину 90-х років минулого століття припав пік дослідницьких робіт в цій області, і з тих пір щороку з'являлося по декілька алгоритмів.

Вперше ця задача була запропонована для знаходження типових шаблонів покупок, що здійснюються в супермаркетах, тому іноді її ще називають аналізом ринкової корзини (англ. Market Basket Analysis).

Асоціативні правила

Нехай є база даних, що складається з купівельних транзакцій. Кожна транзакція — це набір товарів, куплених покупцем за один візит. Таку транзакцію щє називають ринковою корзиною.

Нехай $I = \{i_1, i_2, i_3, \ldots, i_n\}$ — множина (набір) товарів (елементів), D — множина транзакцій, де кожна транзакція T — набір елементів з I, $T \subseteq I$. Кожна транзакція являє собою бінарний вектор, де $t[k] = 1$, якщо i_k присутній в

транзакції, інакше $t[k] = 0$. Будемо говорити, що транзакція T містить X, деякий набір елементів з I, якщо $X \subset T$.

Асоціативним правилом називається імплікація $X \Rightarrow Y$, де $X \subset I$, $Y \subset I$ і $X \cap Y = \emptyset$.

Правило $X \Rightarrow Y$ має підтримку (англ. Support) s, якщо s% транзакцій з D містять $X \cup Y$, $supp(X \Rightarrow Y) = supp(X \cup Y)$.

Достовірність (англ. Confidence) правила вказує, яка ймовірність того, що з X слідує Y. Правило $X \Rightarrow Y$ справедливе з достовірність c, якщо c% транзакцій з D, що містять X, також містять Y, $conf(X \Rightarrow Y) = supp(X \cup Y) / supp(X)$[2].

Іншими словами, метою даного аналізу є встановлення наступних залежностей: якщо в транзакції зустрівся деякий набір елементів X, то на підставі цього можна зробити висновок про те, що інший набір елементів Y також повинен з'явитися в цій транзакції. Встановлення таких залежностей дає нам можливість знаходити дуже прості і інтуїтивно зрозумілі правила.

Алгоритми пошуку асоціативних правил призначений для знаходження всіх правил $X \Rightarrow Y$, причому підтримка і достовірність цих правил повинні бути вище деяких наперед визначених порогів, званих відповідно мінімальною підтримкою (minsupport) і мінімальної достовірністю (minconfidence).

Задача знаходження асоціативних правил розбивається на дві підзадачі:

- знаходження всіх наборів елементів, які задовольняють порогу minsupport; такі набори елементів називаються такими, що часто зустрічаються;
- генерація правил з наборів елементів, знайдених згідно з достовірністю, що задовольняє порогу minconfidence.

Один з перших алгоритмів, що ефективно вирішує подібний клас задач — це алгоритм APriori. Крім цього алгоритму останнім часом був розроблений ряд інших алгоритмів: DHP, Partition, DIC та інші.

Значення для параметрів мінімальна підтримка і мінімальна достовірність вибираються таким чином, щоб обмежити кількість знайдених правил. Якщо підтримка має велике значення, то алгоритми знаходитимуть правила, добре

відомі аналітикам або настільки очевидні, що немає ніякого сенсу проводити такий аналіз. З іншого боку, низьке значення підтримки веде до генерації величезної кількості правил, що, звичайно, вимагає істотних обчислювальних ресурсів. Тим не менше, більшість цікавих правил знаходиться саме при низькому значенні порогу підтримки. Хоча занадто низьке значення підтримки веде до генерації статистично необґрунтованих правил.

Пошук асоціативних правил зовсім не тривіальна задача, як може здатися на перший погляд. Одна з проблем — алгоритмічна складність при знаходженні наборів елементів, що часто зустрічаються, оскільки із зростанням числа елементів в I $(|I|)$ експоненціально зростає число потенційних наборів елементів.

Узагальнені асоціативні правила

Під час пошуку асоціативних правил було зроблено припущення, що всі аналізовані елементи однорідні. Аналізуючи ринковий кошик, потрібно зазначити, досліджуються товари, що мають абсолютно однакові характеристики, за винятком назви. Однак, не складе великих труднощів доповнити транзакцію інформацією про те, в яку товарну групу входить товар і побудувати ієрархію товарів. Розглянемо приклад такого групування (таксономії) у вигляді ієрархічної моделі на рисунку 1.7.

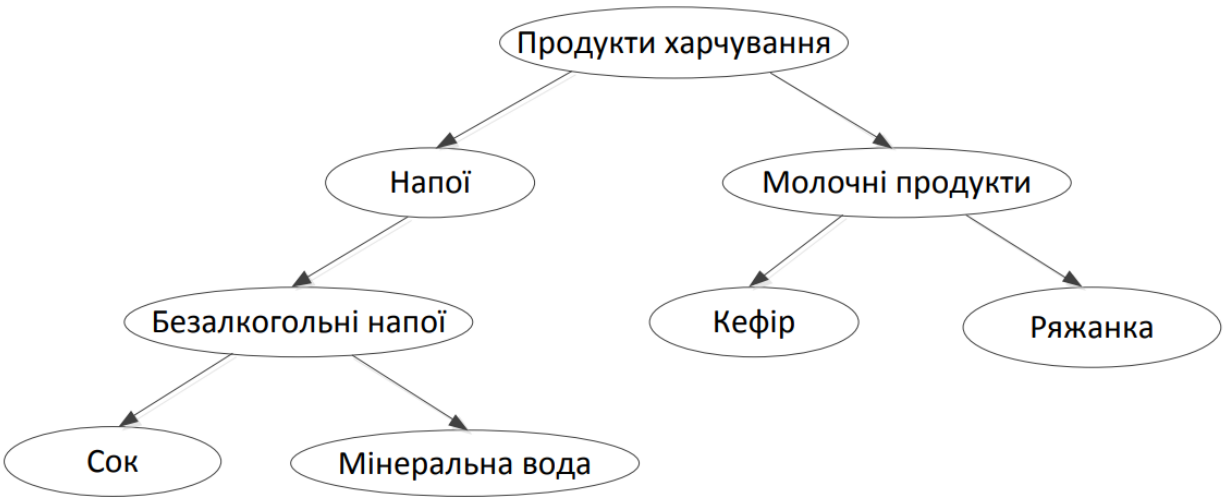


Рис. 1.7 Ієрархічна модель

Нехай дана база транзакцій D і відомо в які групи (таксони) входять елементи. Тоді можна отримувати з даних правила, що зв'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» або «Сік». Ці правила носять назву узагальнених асоціативних правил.

Загальним асоціативним правилом (англ. Generalized Association Rules) називається імплікація $X \Rightarrow Y$, де $X \subset I$, $Y \subset I$ і $X \cap Y = \emptyset$ і жодний з елементів, що входить до набору Y не є предком жодного елемента, що входять в X. Підтримка і достовірність рахуються так само, як і у випадку асоціативних правил.

Введення додаткової інформації про групування елементів у вигляді ієрархії дає наступні переваги.

- Це допомагає встановити асоціативні правила не тільки між окремими елементами, а й між різними рівнями ієрархії (групами).
- Окремі елементи можуть мати недостатню підтримку, але в цілому група може задовольняти порог minsupport.

Для знаходження таких правил можна використовувати будь-який з вищеназваних алгоритмів. Для цього кожну транзакцію потрібно доповнити всіма предками кожного елемента, що входить в транзакцію. Однак, застосування цих алгоритмів може призведи до наступних проблем:

1. елементи на верхніх рівнях ієрархії прагнуть до значно більших значень підтримки у порівнянні з елементами на нижніх рівнях;
2. з додаванням в транзакції груп збільшилась кількість атрибутів і відповідно розмірність вхідного простору; це ускладнює завдання, а також веде до генерації більшої кількості правил;
3. поява надлишкових правил, що суперечать визначенню узагальненого асоціативного правила, наприклад, «Сік»-«Прохолодні напої». Очевидно, що практична цінність такого "відкриття" нульова при 100%

достовірності; отже, потрібні спеціальні оператори, що видаляють подібні надмірні правила.

Для знаходження узагальнених асоціативних правил бажано використання спеціалізованого алгоритму, який усуває вищеописані проблеми і до того ж працює в 2-5 разів швидше, ніж стандартний APriori.

Групувати елементи можна не тільки по входженню в певну товарну групу, а й за іншими характеристиками, наприклад за ціною (дешево, дорого), брендом тощо [2].

Кластеризація даних методами Data Mining

Під кластеризацією зазвичай мають на увазі процес перевірки набору «точок» та групування точок у «кластери» згідно міри довжини. Суть полягає у тому, що точки з одного кластера матимуть невелику відстань одна від одної, а точки з різних кластерів навпаки — велику. Під «точками» мається на увазі n-вимірний вектор характеристик.

Придатний до кластеризації набір даних представляє собою множину точок, які належать деякому простору. Простором можна вважати універсальну множину, з якої беруться точки набору даних (наприклад, евклідовий простір має багато важливих властивостей, які можуть бути корисними при кластеризації; зокрема, точки евклідового простору є векторами дійсних чисел). Довжина вектора визначається кількістю вимірів простору. Компонентами вектора є координати відповідних точок.

Усі простори, для яких може бути виконана кластеризація, мають міру довжини, яка задає відстань між двома точками у просторі. Загальна Евклідова відстань (квадратній корінь сум квадратів різниці між координатами точок у кожному вимірі) працює для всіх Евклідових просторів, як і інші методи вимірювання відстаней у Евклідових просторах: манхетенська (сума магнітуд різниць у кожному вимірі) та L∞-відстань (максимальна магнітуда різниць в будь-якому просторі).

Алгоритми кластеризації можна розділити на дві групи згідно двох фундаментально різних стратегій.

1. Ієрархічні або агломеративні алгоритми стартують кожен із власної точки у кластері. Кластери об’єднуються за близькістю використовуючи одне з 81 багатьох визначень «близькості». Об’єднання зупиняється коли подальше об’єднання приводить до небажаних результатів. Наприклад, алгоритм може зупинитись, коли заздалегідь визначена кількість кластерів, або використовується міра компактності кластерів і відмовитись від побудови кластера шляхом об’єднання двох менших кластерів, якщо результуючий кластер має точки на дуже великій відстані одна від одної.

2. Інший клас алгоритмів містить етап ініціалізації початкових точок, що визначають кластери. Точки задаються у будь-якому порядку і кожна приписується до кластера, до якого вона найбільше підходить. Цей процес зазвичай займає коротку фазу, в якій задаються початкові кластери. Деякі види алгоритмів також дозволяють об’єднання або розділення кластерів, або дозволять позбутися точок, які знаходяться на віддаленні від усіх кластерів.

Алгоритми кластеризації також можна розділити за наступними критеріями.

1. Працює алгоритм у Евклідовому просторі або в довільній системі виміру. В евклідовому просторі можна сумувати набір точок за їх центроїдою - середнім арифметичним точок. В неевклідовому просторі не існує визначення центроїди і потрібні інші шляхи сумування кластерів.

2. Використовує алгоритм лише первинну пам'ять у випадку малого обсягу даних, або йому буде потрібна вторинна пам'ять, якщо даних забагато. Алгоритми для великих обсягів даних часто використовують скорочення, оскільки неможливо проаналізувати всі пари точок у первинній пам'яті[2]. Також важливо сумувати кластери в основній пам'яті, оскільки утримувати усі точки усіх кластерів в первинній пам'яті неможливо.

Постановка задачі кластеризації

Кластеризація — це автоматичне розбиття елементів деякої множини на групи в залежності від їх подібності. Елементами множини є дані або вектори характеристик. Власне групи прийнято називати кластерами.

Кластеризація (об'єднання в групи схожих об'єктів) є однією із фундаментальних задач Data Mining. Список прикладних областей, де вона застосовується, широкий: сегментація зображень, маркетинг, боротьба з шахрайством, прогнозування, аналіз текстів, аналіз освітніх даних тощо. Задачу кластеризації в тому чи іншому вигляді формулювали в таких наукових напрямках, як статистика, розпізнавання образів, оптимізація, машинне навчання. Звідси різноманіття синонімів поняттю кластер — клас, таксон, згущення.

Також кластеризація є важливою формою абстракції даних.

Кластеризація є розділом сучасної теоретичної інформатики, що бурхливо розвивається, і в цій області можна отримати ряд цікавих дослідницьких результатів.

Кластеризація розбиває множину об’єктів на групи, які визначаються лише її результатом. Класифікація відносить кожен об’єкт до однієї із заздалегідь визначених груп.

Кластеризація включає в себе наступні етапи.

1. Виділення характеристик
2. Визначення метрики
3. Розбиття об’єктів на групи
4. Представлення результатів

Для початку необхідно обрати властивості, які характеризують наші об’єкти. Далі — спробувати зменшити розмірність простору характеристичних векторів, тобто виділити найбільш вагомі властивості об’єктів. Зменшення розмірності пришвидшує процес кластеризації, а в деяких випадках дозволяє візуально оцінювати результати.

Виділені характеристики потрібно нормалізувати.

Далі всі об’єкти представляються у вигляді характеристичних векторів.

Наступним етапом є вибір метрики, за якою будемо визначати схожість об’єктів.

Метрика обирається в залежності від:

1. простору, в я кому розташовані об’єкти;
2. неявних характеристик кластерів.

Наприклад, якщо всі координати об’єкта неперервні і дійсні, а кластери мають представляти собою дещо на кшталт гіперсфер, то використовується класична метрика Евкліда:

$$d_2(x_i, x_j) = \sqrt{\sum_{k=1}^{d} (x_{ik} - x_{jk})^2} = \|x_i - x_j\|_2 \tag{1.13}$$

Для категорійних атрибутів поширена міра подібності Чекановського-Серенсена і Жаккара:

$$|t_1 \cap t_2| / |t_1 \cup t_2| \tag{1.14}$$

При виконанні кластеризації важливо, кількість побудованих кластерів. Передбачається, що кластеризація має виявити природні локальні згущення об’єктів. Тому число кластерів є параметром, що часто суттєво ускладнює вид алгоритму, якщо воно невідоме; і таким, що суттєво впливає на якість результату, якщо воно відоме.

Проблема вибору числа кластерів зовсім нетривіальна. Достатньо сказати, що для отримання задовільного теоретичного рішення часто вимагається зробити сильне припущення про властивості деякого заздалегідь заданого сімейства розподілів[2]. Але на початку дослідження про дані заздалегідь нічого невідомо, тому алгоритми кластеризації зазвичай будуються як деякий

спосіб перебору числа кластерів, і визначення його оптимальних значень в процесі перебору.

Число методів розбиття на множини і кластери достатньо велике. Всі їх можна розділити на ієрархічні та неієрархічні.

В неієрархічних алгоритмах характер їх роботи і умову зупинки треба завчасно регламентувати часто достатньо великим числом параметрів, що іноді важко. Але в таким алгоритмах досягається гнучкість у варіюванні кластеризації і зазвичай визначається числом кластерів.

З іншої сторони, коли об'єкти характеризуються великим числом ознак (параметрів), то важливе значення набуває задача групування ознак. Вхідна інформація знаходиться в квадратній матриці зв'язків ознак.

В ієрархічних алгоритмах фактично відмовляються від визначення числа кластерів, будуючи повне дерево вкладених кластерів (дендрограму). Число кластерів визначається з припущень, що не відносяться до алгоритму. Труднощі таких алгоритмів: вибір близькості кластерів, проблема інверсій індексації, негнучкість, яка зазвичай небажана.

Прогнозування і часові ряди.

Основою для прогнозування служить історична інформація, що зберігається в базі або сховищі даних у вигляді часових рядів. Існує поняття Data Mining часових рядів (Time-Series Data Mining). На основі ретроспективної інформації у вигляді часових рядів можливий розв'язок різних задач Data Mining.

Приведемо дві принципові відмінності часового ряду від простої послідовності спостережень:

- члени часового ряду, на відміну від елементів випадкової вибірки, не є статистично незалежними.
- члени часового ряду не є однаково розподіленими.

Часовий ряд – послідовність спостережуваних значень будь-якої ознаки, упорядкованих у невипадкові моменти часу.

Відмінністю аналізу часових рядів від аналізу випадкових вибірок є припущення про рівні проміжки часу між спостереженнями та їх хронологічний порядок. Прив'язка спостережень до часу відіграє тут ключову роль, тоді як при аналізі випадкової вибірки вона не має ніякого значення.

Типовий приклад часового ряду – дані біржових торгів.

Інформація, накопичена в різноманітних базах даних підприємства, є часовими рядами, якщо вона розташована в хронологічному порядку і зроблена в послідовні моменти часу.

Аналіз часового ряду здійснюється з метою:

- визначення природи ряду;
- прогнозування майбутніх значень ряду.

У процесі визначення структури й закономірностей часового ряду передбачається виявлення: шумів і викидів, тренду, сезонного компонента, циклічного компонента. Визначення природи часового ряду може бути використане як своєрідна «розвідка» даних. Знання аналітика про наявність сезонного компонента необхідне, наприклад, для визначення кількості записів вибірки, яка повинна брати участь у побудові прогнозу.

Аналіз часового ряду ускладнюють шуми й викиди (будуть докладно розглянуті в наступних темах курсу). Існують різні методи визначення й фільтрації викидів, що дають можливість виключити їх з метою більш якісного Data Mining.

Основними складовими часового ряду є тренд і сезонний компонент. Тренд є систематичним компонентом часового ряду, який може змінюватися в часі.

Трендом називають невипадкову функцію, яка формується під дією загальних або довгочасних тенденцій, що впливають на часовий ряд.

Прикладом тенденції може виступати, наприклад, фактор зростання досліджуваного ринку.

Автоматичного способу виявлення трендів у часових рядах не існує. Але якщо часовий ряд включає монотонний тренд (тобто відзначене його стійке

зростання або стійке спадання), аналізувати часовий ряд у більшості випадків неважко.

Існує велика різноманітність постановок задач прогнозування, які можна поділити на дві групи: прогнозування односерійних рядів і прогнозування мультисерійних, або взаємовпливаючих, рядів.

Група прогнозування односерійних рядів включає задачу побудови прогнозу однієї змінної за ретроспективними даними тільки цієї змінної, без врахування впливу інших змінних і факторів.

Група прогнозування мультисерійних, або взаємовпливаючих, рядів включає задачу аналізу, де необхідно враховувати взаємовпливаючі фактори на одну або декілька змінних.

Крім розподілу на класи по односерійності й багатосерійності, ряди також бувають сезонними й несезонними.

Останній розподіл має на увазі наявність або відсутність у часового ряду такої складової як сезонність, тобто включення сезонного компонента.

Сезонна складова часового ряду є періодично повторюваним компонентом часового ряду.

Властивість сезонності означає, що через приблизно рівні проміжки часу форма кривої, яка описує поведінку залежної змінної, повторює свої характерні обриси.

Властивість сезонності важлива при визначенні кількості ретроспективних даних, які будуть використовуватися для прогнозування.

1.2. Діяльність вищих навчальних закладів, як об’єкту аналізу

1.2.1 Загальна характеристика НУХТ

Національний університет харчових технологій — навчально-науковий комплекс технічного профілю, де здійснюється підготовка висококваліфікованих фахівців освітньо-кваліфікаційних рівнів «бакалавр», «спеціаліст» і «магістр» за 215 освітніми програмами для м'ясо-молочної, хлібопекарської, кондитерської, фармацевтичної і мікробіологічної

промисловості, інших галузей агропромислового комплексу та харчового машинобудування країни. У комплексі здобувають знання близько 30 тисяч студентів та слухачів денної та заочної форм навчання.

НУХТ — вищий заклад освіти IV рівня акредитації. Навчально-виховний процес, науково-методична та науково-дослідна робота забезпечуються в межах шести факультетів та інститутів: Навчально-науковому інституті харчових технологій, Навчально-науковому інженерно-технічному інституті ім. акад. І. С. Гулого, Навчально-науковому інституті економіки і управління, факультеті біотехнології та екологічного контролю, факультеті автоматизації і комп'ютерних систем, факультеті готельно-ресторанного та туристичного бізнесу. Крім того, до складу комплексу входять 2 інститути післядипломної освіти, 6 філій: у Львові (дві), Смілі, Сумах, Кам'янець-Подільському та Полтаві, а також 11 коледжів, які знаходяться у різних куточках країни.

В університеті функціонують 42 кафедри (35 із них — випускові), які мають 24 філії та 8 навчально-науково-виробничих комплексів на передових підприємствах, у проектних і науково-дослідних установах. Крім того, за останні роки створені 3 навчально-виробничих центри, редакційно-видавничий центр, центр інформаційних технологій, який об'єднує факультетські комп'ютерні центри, комп'ютерні класи та 52 локальні мережі. В їх роботі задіяні понад 3000 комп'ютеров.

У навчально-науковому комплексі «Національний університет харчових технологій» (НУХТ) працюють близько 5000 співробітників, серед яких: понад 120 професорів, докторів наук; близько 800 доцентів, кандидатів наук. Серед них 21 академік, 16 лауреатів Державної премії України, заслужені діячі науки і техніки України, заслужені працівники вищої школи, відмінники освіти України.

Фахівців вищої кваліфікації готують в аспірантурі та докторантурі. В університеті функціонують 7 спеціалізованих вчених рад із захисту дисертацій за 13 науковими спеціальностями.

Співробітники університету за роки незалежності отримали понад 1600 патентів України на винаходи та корисні моделі, зокрема понад 400 — у співавторстві зі студентами. Крім того, отримано 27 патентів та 2 авторських свідоцтва за межами України.

1.2.2. Огляд інформаційної системи впровадженої у НУХТ

В Національному університеті харчових технологій для інформаційної підтримки діяльності всіх ланок університету застосовується програмне забезпечення ПП "Політек-СОФТ для вищих навчальних закладів України.

Впроваджена інформаційна система (ІС) складається з пакетів програм, що призначені для використання різними відділами ВНЗ. Для збереження інформації про навчання студентів призначений пакет програм "Деканат", саме його ми будемо розглядати, як основне інформаційне джерело даних для подальшого аналізу навчання студентів засобами Data Mining.

Пакет програм "Деканат" - це автоматизована система управління вищим навчальним закладом (АСУ ВНЗ), яка призначена для організації та підтримки навчального процесу в вищих навчальних закладах України I-IV рівнів акредитації.

Основною метою Пакету є скоротити час, що витрачають працівники вищих навчальних закладів на розв`язання повсякденних задач та спростити процедуру роботи з даними.

Пакет побудований за клієнт-серверною технологією, що дозволяє встановлювати його на множину комп`ютерів, які об`єднані в локальну мережу та працюють з єдиною базою даних. Використання додаткових web-сценаріїв забезпечує можливість доступу до бази даних в межах окремих програм Пакету з всесвітньої павутини Інтернет. В ролі сервера управління базами даних використовується FireBird.

До складу Пакету додатково входить програма "ПС-Адміністратор", яка призначена для щоденного тестування, резервного копіювання та, при

необхідності, відновлення бази даних. Ця Програма позбавляє вищий навчальний заклад від необхідності додатково наймати на роботу фахівця, що відповідає за обслуговування систем управління базами даних.

Така організація роботи Пакету забезпечує високу надійність збереження даних та їх достовірність, а його інформаційна сумісність з іншими продуктами ПП "Політек-СОФТ" забезпечує імпортування даних, які вже були внесені в бази даних інших продуктів. Наприклад, можливо імпортувати анкетні дані студентів з пакету програм "ПС-Абітурієнт" та не вводити їх повторно в базу даних Пакету.

До роботи з Пакетом можуть бути залучені як окремі працівники вищого навчального закладу (навчальна частина, секретарі деканатів та кафедр), так і всі учасники навчального процесу (викладачі та студенти). Таке розширення обсягу використання пакету надає новий програмний модуль "ПС-Журнал успішності-Web" (електронний журнал успішності).

Пакет має зручний конструктор звітів, який дозволяє створювати та редагувати вже існуючі звітні документи, використовуючи HTML - мову розмітки гіпертексту. Звітні документи, які генерує Пакет, можна переглядати перед відправкою на друк в програмах MS Word, MS Excel, інтернет-браузері та додатково редагувати до Ваших вподобань.

"Деканат" - пакет програм, що призначений для автоматизації планування та обліку навчального процесу.

Основні можливості Пакету:

- формування даних щодо структури навчального процесу;
- формування даних щодо всіх викладачів та їх планового навантаження, розклад їх роботи;
- формування даних щодо щоденних даних про фактичну роботу кожного викладача з кожної дисципліни;
- формування великого обсягу даних щодо всіх студентів та їх успішності за весь період навчання;

- формування даних щодо наявності корпусів та аудиторій, їх заповнення, розклад занять.

Особливості Пакету:

- великий обсяг та повнота інформації, яка зберігається в базі даних;
- великий обсяг звітів, які можна підготувати на основі даних з бази даних з урахуванням вимог Міністерства освіти і науки України;
- інформаційна сумісність з іншими продуктами ПП "Політек-СОФТ".

В якості джерела даних про навчання студентів НУХТ може бути використаний також використаний сайт дистанційного навчання студентів НУХТ, який базується на платформі Moodle.

1.3.Огляд сучасних інструментів для проведення інтелектуального аналізу даних

Розглянемо найбільш затребувані на ринку програмного забезпечення програмі продукти, які реалізують технологію Data Mining.

Система ADaM

Система ADaM (Algorithm Development and Mining System), розроблена Центром Інформаційних Технологій і Систем (ITSC) в Університеті Алабами , яка використовується для дистанційної обробки наукових даних технологіями Data Mining. Створені засоби Data Mining складаються з взаємодіючих компонентів, які можна для різних прикладних задач включати у спеціалізовані додатки. ADaM містить понад 100 компонентів, які можуть бути конфігуровані так, щоб за замовленням користувача створювати необхідні процеси інтелектуального аналізу даних. Нові компоненти можуть бути легко додані, щоб пристосувати систему до інших проблем науки.

Кожний компонент ADaM підтримується C, C++, або іншим програмним інтерфейсом додатку (API), загальними інструментальними засобами опису (Perl, Python, сценарії оболонок) і кінець кінцем інтерфейсом WEB-сервісів, що

забезпечує використання Web і Grid додатків. Компоненти ADaM – універсальні модулі інтелектуального аналізу даних (mining) і обробки зображень, які можуть бути легко пристосовані для численних рішень і задач[3].

Система SAS Data mining:

Система статистичного аналізу є продуктом SAS. Він був розроблений для аналітики та управління даними, для користувачів пропонується використання зручного графічного інтерфейсу.

Особливості:

- Інструменти SAS Data mining допоможуть аналізувати великі дані.
- Це зручний інструмент для інтелектуального аналізу даних, аналізу тексту та оптимізації.
- SAS пропонує розподілену архітектуру обробки пам'яті, яка відмінно масштабується.

Система Sisense.

Sisense є ще одним ефективним інструментом інтелектуального аналізу даних. Він миттєво аналізує і візуалізує як великі, так і розрізнені набори даних. Це зручний інструмент для створення інформаційних панелей з різноманітними візуалізаціями.

Особливості:

- Дозволяє створювати інтерактивні інформаційні панелі без технічних навичок.
- Об'єднувати незв'язані дані в одному централізованому місці.
- Дозволяє отримати доступ до інструментальних панелей навіть в мобільному пристрої.
- Зручна візуалізація результатів.
- Ідентифікує критичні метрики, використовуючи фільтрацію і обчислення
- Обробляє великомасштабні дані на одному звичайному сервері[3].

Система Oracle Data Mining.

У березні 1998 компанія Oracle оголосила про спільну діяльність з 7 партнерами, які є постачальниками інструментів Data Mining. Далі було здійснено включення в Oracle8i засобів підтримки алгоритмів Data mining. У червні 1999 року Oracle купує Darwin (Thinking Machines Corp.) і у 2000-2001 роках виходять нові версії Darwin, Oracle Data Mining Suite. У червні 2001 року виходить Oracle9i Data Mining.

Oracle Data Mining є опцією або модулем в Oracle Enterprise Edition. Опція Oracle Data Mining (ODM) призначена для аналізу даних методами, що відносяться до технології вилучення знань, або Data Mining. ODM підтримує всі етапи технології вилучення знань, включаючи постановку задачі, підготовку даних, автоматичну побудову моделей, аналіз і тестування результатів, використання моделей в реальних застосуваннях. Важливо, що моделі будуються автоматично на основі аналізу наявних даних про об'єкти, спостереження і ситуації за допомогою спеціальних алгоритмів. Основу опції ODM складають процедури, що реалізовують різні алгоритми побудови моделей класифікації, регресії, кластеризації. На етапі підготовки даних забезпечується доступ до будь-яких реляційних баз, текстових файлів, файлів формату SAS. Додаткові засоби перетворення і очищення даних дозволяють змінювати вигляд сценарію, проводити нормалізацію значень, виявляти невизначені або відсутні значення. На основі підготовлених даних спеціальні процедури автоматично будують моделі для подальшого прогнозування, класифікації нових ситуацій, виявлення аналогій. ODM підтримує побудову п'яти різних типів моделей. Графічні засоби надають широкі можливості для аналізу отриманих результатів, верифікації моделей на тестових наборах даних, оцінки точності і стійкості результатів. Уточнені і перевірені моделі можна включати в існуючі додатки шляхом генерації їх описів на C++, Java, а також розробляти нові спеціалізовані застосування за допомогою того, що входить до складу середовища ODM засобу розробки Software Development Kit (SDK) [4].

Аналітична платформа Deductor.

Склад і призначення аналітичної платформи Deductor (розробник - компанія BaseGroup Labs).

Deductor складається з двох компонентів: аналітичного додатка Deductor Studio і багатомірного сховища даних Deductor Warehouse.

Deductor Warehouse - багатовимірне сховище даних (СД), що акумулює всю необхідну для аналізу предметної області інформацію. Використання єдиного сховища дозволяє забезпечити несуперечність даних, їх централізоване зберігання і автоматично створює всю необхідну підтримку процесу аналізу даних. Deductor Warehouse оптимізований для вирішення саме аналітичних задач, що позитивно позначається на швидкості доступу до даних. Deductor Studio - це програма, призначена для аналізу інформації з різних джерел даних. Вона реалізує функції імпорту, обробки, візуалізації і експорту даних. Deductor Studio може функціонувати і без сховища даних, отримуючи інформацію з будь-яких інших джерел, але найбільш оптимальним є їх спільне використання. Розглянемо цей процес детальніше. На початковому етапі в програму завантажуються або імпортуються дані з якого-небудь довільного джерела. Сховище даних Deductor Warehouse є одним з джерел даних. Підтримуються також інші сторонні джерела. Зазвичай в програму завантажуються не всі дані, а якась вибірка, необхідна для подальшого аналізу. Після здобуття вибірки можна отримати детальну статистику по ній, проглянути, як виглядають дані на діаграмах і гістограмах. Такий розвідувальний аналіз дає можливість приймати рішення про необхідність передобробки даних. Наприклад, якщо статистика показує, що у вибірці є порожні значення (пропуски даних), можна застосувати фільтрацію для їх усунення. Передоброблені дані далі піддаються трансформації. Наприклад, нечислові дані перетворяться в числові, що необхідне для деяких алгоритмів. Безперервні дані можуть бути розбиті на інтервали, тобто виробляється їх дискретизація. До трансформованих даних застосовуються методи глибшого аналізу. На цьому етапі виявляються приховані залежності і закономірності в даних, на підставі яких будуються різні

моделі. Модель є шаблоном, який містить формалізовані знання. Останній етап інтерпретація – призначений для того, щоб з формалізованих знань отримати знання на мові наочної області. Вся робота по аналізу даних в Deductor Studio базується на виконанні наступних дій: імпорт даних, обробка даних, візуалізація, експорт даних. Відправною точкою для аналізу завжди є процедура імпорту даних. Отриманий набір даних може бути оброблений будь-яким з доступних способів. Результатом обробки також є набір даних, який, у свою чергу, знову може бути оброблений. Імпортований набір даних, а також дані, отримані на кожному етапі обробки, можуть бути експортовані для подальшого використання в інших, наприклад, в облікових системах. Результати кожної дії можна відобразити різними способами: OLAP-кубы (крос-таблиця, крос-діаграма), плоска таблиця, діаграма, гістограма, статистика, аналіз за принципом "що-якщо", граф нейромережі, дерево - ієрархічна система правил, інше. Послідовність дій, які необхідно провести для аналізу даних, називається сценарієм. Сценарій можна автоматично виконувати на будь-яких даних.

Інформація у сховищі Deductor Studio міститься в структурах типу "зірка", де в центрі розташовані таблиці фактів, а "променями" є виміри. Така архітектура сховища найбільш адекватна завданням аналізу даних. Кожна "зірка" називається процесом і описує певну дію. У Deductor Warehouse може одночасно зберігатися безліч процесів, що мають загальні виміри.

Сховищем Deductor Warehouse фізично є реляційна база даних, яка містить таблиці для зберігання інформації і таблиці зв'язків, що забезпечують цілісне зберігання відомостей. Поверх реляційної бази даних реалізований спеціальний прошарок, який перетворить реляційну систему до багатомірної. Багатомірна ідеологія використовується тому, що воно набагато краще реляційної відповідає ідеології аналізу даних. Завдяки цьому прошарку користувач оперує багатомірними поняттями, такими як "вимір" або "факт", а система автоматично виробляє всі необхідні маніпуляції, необхідні для роботи з

реляційною СУБД. Окрім консолідації даних, робота із створення закінченого аналітичного рішення містить декілька етапів:

Очищення даних. На цьому етапі проводиться редагування аномалій, заповнення пропусків, згладжування, очищення від шумів, виявлення дублікатів і протиріч.

Трансформація даних. Виробляється заміна порожніх значень, квантування, таблична заміна значень, перетворення до ковзаючого вікна, зміна формату набору даних.

Data Mining. Будуються моделі з використанням нейронних мереж, дерев рішень, асоціативних правил тощо [4].

Програмний пакет Microsoft Analysis Services

Microsoft Analysis Services (Служби аналізу від Microsoft) - частина Microsoft SQL Server, системи управління базами даних (СКБД). Microsoft включила набір служб в SQL Server, пов'язаних з бізнес-аналізом і зберіганням даних. Ці служби включають в себе служби інтеграції (Integration Services) та служби аналізу (Analysis Services). Analysis Services, в свою чергу, включають в себе набір засобів для роботи з OLAP і інтелектуальним аналізом даних.

Microsoft Analysis Services підтримує різні набори прикладних програмних інтерфейсів (API) та об’єктних моделей для різних операцій у різних програмних середовищах. Служби Analysis Services надають такі функції і засоби для створення рішень з інтелектуального аналізу даних: набір стандартних алгоритмів інтелектуального аналізу даних; конструктор інтелектуального аналізу даних, призначений для створення й перегляду моделей інтелектуального аналізу даних, управління ними та побудови прогнозів; мова розширень інтелектуального аналізу даних.

Модель – це основа видобування даних в SQL Server. По суті, модель є сукупністю метаданих, що відображають деякі правила і закономірності у початкових даних. При цьому структура моделі визначає набір ключових атрибутів аналізу, тоді як її зміст несе безпосередньо статистичну інформацію –

тут простежується схожість з ідеологією звичайних таблиць. Проте варто мати на увазі, що на основі одного і того самого набору початкових даних можна побудувати декілька різних моделей. У цьому сенсі побудова правильної моделі гарантує нам отримання саме тих «прихованих» залежностей, які ми прагнемо виявити. За те, як виконуватиметься аналіз даних, відповідає алгоритм аналізу. Всі утиліти аналізу даних, включаючи Microsoft SQL Server Analysis Services, використовують безліч алгоритмів. Використання готових алгоритмів спрощує роботу із створення застосування, хоча за допомогою аналітичного сервера і мов програмування можна створити і свої власні моделі. Процес побудови моделі реалізований в Analysis Services у вигляді майстра, що дає змогу крок за кроком задавати параметри моделі і виконувати її обробку, що, на думку розробників, спрощує проведення аналізу.

Алгоритм Microsoft Decision Trees ґрунтується на відомому методі побудови дерев рішень. У його межах значення кожного з досліджуваних атрибутів класифікується на основі значень решти атрибутів з використанням правил вигляду «якщо — то». Результат роботи такого алгоритму — деревоподібна структура, кожен вузол якої є якимось питанням. Щоб вирішити, до якого класу віднести деякий об'єкт або ситуацію, потрібно відповісти на питання, що стоять у вузлах цього дерева, починаючи з його кореня. Алгоритм Microsoft Decision Trees будує модель, створюючи різні зрізи даних, які називають також вузлами дерева. Він додаватиме вузли кожного разу, коли вхідний стовпець (input column), тобто стовпець, який аналізують, буде значною мірою пов'язаний із стовпцем, що передбачається (predicatable column). Спосіб поділу залежить від статистики та структури даних.

Алгоритм Microsoft Clustering. Цей алгоритм використовує інший, не менш відомий метод пошуку логічних закономірностей — метод «найближчого сусіда». У процесі роботи алгоритму початкові дані об'єднуються в групи (кластери) на основі аналогічних або схожих значень атрибутів. Отримані набори даних аналізуються, що дає змогу виявити приховані закономірності або побудувати ймовірнісний прогноз. За цим алгоритмом проводять глибший

аналіз даних, ніж за деревом рішень, але і він має свої обмеження. Його переважно застосовують для наборів даних зі схожими атрибутами, значення яких належать певному інтервалу.

Алгоритм Naive Bayes. Ця ймовірнісна модель корисна при класифікації і поглибленому дослідженні даних. Засіб перегляду виразно показує відмінності між двома станами вхідної змінної. Алгоритм забезпечує розрахунок умовної вірогідності між вхідними значеннями і значеннями, що передбачаються.

Алгоритм Sequence Clustering. Алгоритм Sequence Clustering поєднує прогнозування, що забезпечується алгоритмом кластеризації, з технологією побудови послідовностей. Послідовністю може бути будь-яка група подій, пов'язаних з користувачем. Алгоритм знаходить найзагальніші послідовності, групуючи події разом, тобто при цьому формується шлях групування.

Алгоритм Time Series. Цей алгоритм, розроблений Microsoft, дає змогу аналізувати і прогнозувати будь-які дані, залежні від часу. За допомогою технології регресивних дерев цей алгоритм здатний виявляти закономірності у декількох послідовностях та бачити, як пов'язані між собою різні події.

Алгоритм Neural Networks. Нейронні мережі – це клас аналітичних методів, що побудовані на (гіпотетичних) принципах навчання мислячих істот і функціонування мозку і які дають змогу прогнозувати значення деяких змінних у нових спостереженнях за даними інших спостережень (для цих же або інших змінних) після проходження етапу так званого навчання на наявних даних. Нейронні мережі є одним з методів «видобування» даних[5].

1.4. Постанова задачі аналізу навчання студентів ВНЗ

В сучасному суспільстві основним завданням закладів вищої освіти є забезпечення студентів знаннями та високі оцінки учасників освітнього процесу. Враховуючи різні навчальні методики, підходи до викладання матеріалу та поведінку студентів впровадження різноманітних інформаційних технологій є невід'ємною частиною освітнього процесу. Сучасні технології значно розширюють інструментарій і методи навчання, урізноманітнюють

спосіб доставки знань.

Інформаційні технології і системи також активно використовуються для інформаційної підтримки управління як навчальним процесом так і навчальним закладом вцілому. Інформаційні системи стали невід'ємною частиною навчання та нерозривно інтегровані в процеси прийняття рішень у вищих навчальних закладах. Впроваджені у закладах вищої освіти ІС накопичили великі масиви даних про активність студентів, вподобання, засоби комунікації, певні персональні дані, успішність навчання тощо. Ці дані можуть бути об'єктом для аналізу та можуть бути використані для прогнозування успішності, створення індивідуальних навчальних планів, визначення поведінки студента та стилю навчання, створення моделей мотивації.

З метою забезпечення аналізу даних, накопичених в інформаційних джерелах ВНЗ в роботі пропонується:

– вивчити діяльність ВНЗ на прикладі Національного університету харчових технологій і галузь аналізу даних - Educational Data Mining;

– виділити задачі аналізу навчання студентів, які можуть бути вирішені з використанням технології Data Mining на основі даних наявних інформаційних джерел ВНЗ;

– спроектувати структури для збереження даних у вигляді найбільш зручному для описаних задач аналізу;

– налаштувати алгоритми Data Mining для вирішення описаних задач аналізу і сформувати рекомендації щодо використання отриманих результатів.

1.5. Висновок до розділу 1

В першому розділі здійснено дослідження сучасних методів аналізу даних.

Досліджено і описано найбільш використовуване програмне забезпечення для проведення інтелектуального аналізу даних.

Описано діяльність Національного університету харчових технологій, як об’єкту аналізу даних. Описане ПЗ, що використовується в НУХТ – як джерело

даних для подальшого дослідження і проведення аналізу навчання студентів засобами Data Mining.

На основі проведених досліджень здійснена постанова задачі аналізу навчання студенів засобами Data Mining.

Розділ 2. Дослідження методів Data Mining для аналізу навчання студентів ВНЗ

2.1. Виділення задач аналізу навчання студентів ВНЗ, які можна вирішити методами Data Mining

Проведене в першому розділі дослідження діяльності ВНЗ і зокрема НУХТ, методів технології Data Mining, програмних засобів, що її реалізують, а також інформаційної системи, впровадженої у НУХТ, як джерела даних для проведення аналізу, дозволило виділити з загальних задач аналізу діяльності ВНЗ ті задачі, які можливо вирішити засобами інтелектуального аналізу даних. В таблиці 2.1 наведено перелік задач аналізу, методи інтелектуального аналізу для їх вирішення і очікуваний результат аналізу.

Таблиця 2.1. Задачі аналізу успішності навчання

Задачі аналізу	**Результат**	**Методи Data Mining**
Прогнозування успішності з кожної дисципліни	Прогноз успішності навчання по дисциплінах	Прогнозування методом часових рядів
Прогнозування успішності по групах	Прогноз успішності навчання по групах	
Прогнозування успішності по курсах	Прогноз успішності навчання по курсах	
Прогнозування успішності за ОПП	Прогноз успішності навчання за ОПП	
Прогнозування успішності навчання для кожного студента	Прогноз успішності навчання студента	
Прогнозування відвідування занять студентами	Прогноз відвідування занять студентами	
Прогнозування кількості абітурієнтів з різних регіонів і країни	Прогноз вступу в розрізі регіонів	
Аналіз залежність між пропусками студентами занять з дисциплін і успішністю	Вплив пропусків занять студентами (з поважних/неповажних причин) на успішність навчання	
Аналіз успішності навчання по періодах (місяць, семестр)	Аналіз успішності навчання по періодах	

Задачі аналізу	Результат	Методи Data Mining
Аналіз успішності навчання студентів з різних регіонів країни	Аналіз навчання студентів з різних регіонів країни	Кластерний аналіз Асоціативні правила Пошук виключень Нейронні мережі
Аналіз успішності студентів в залежності від попередньої освіти	Аналіз успішності навчання студентів з різною початковою освітою	
Аналіз успішності студентів в залежності від статі	Аналіз успішності за статтю	
Аналіз впливу пропусків занять на успішність навчання	Аналіз пропусків занять на успішність навчання	
Аналіз впливу різноманітних факторів на успішність навчання	Аналіз успішності навчання	

2.2. Сховище даних, як джерело аналітичної інформації

Проблеми розрізненості зберігання даних в рамках одного підприємства, необхідність залучення технічних фахівців для вилучення з корпоративних баз даних необхідної для прийняття рішень інформації призвели у 80 рр. XX ст. до ідеї централізованого зберігання даних, необхідних для подальшого аналізу. Таким чином, виник термін «сховище даних».

Сховища даних є спеціалізованими базами даних, що мають наступні властивості:

- предметна орієнтованість. У сховищі містяться дані, що всебічно описують певну предметну область;
- інтегрованість. Дані збираються з безлічі різних джерел, узагальнюються і зберігаються в єдиному корпоративному сховищі;
- забезпечення несуперечності даних. Дані з різних джерел можуть містити дублюючі, суперечливі відомості, тому перед їх завантаженням в сховище вони проходять процедури перевірки, узгодження, доповнення, узагальнення;

- незмінюваність. На відміну від баз даних транзакційних систем, в яких оперативні дані можуть редагуватися користувачами, дані в сховищі використовуються виключно в режимі читання і недоступні для коригування;
- підтримка хронології. Оскільки для цілей аналізу і прогнозування розвитку предметної області необхідно бачити її показники в динаміці, дані зберігаються в хронологічному порядку, максимально довгий термін;
- оптимізація під виконання складних аналітичних запитів. Сховище проектується таким чином, щоб мінімізувати час на формування аналітичної звітності, необхідної для підтримки прийняття рішень для керівників і менеджерів.

Якщо в базах даних транзакційних систем дані надходять в процесі бізнес-діяльності (продажі товарів фіксуються в системі за фактом продажу, товари, що надійшли на склад, враховуються за фактом надходження на склад і т.п.), то для поповнення даних у сховищі здійснюється їх періодичне вивантаження з інформаційних джерел. Процес розміщення інформації в сховищах даних передбачає періодичний збір, очищення та інтеграцію розрізнених даних з подальшим їх перетворенням в статичні, постійні структури.

Джерелами даних для інформаційного сховища є дані з розрізнених ІС, які, як правило, працюють на основі реляційних СУБД, що обслуговують повсякденну діяльність підприємства. Джерелами можуть бути і дані, що надходять від зовнішніх організацій - інформаційних агентств, консалтингових компаній, засобів масової інформації, Інтернет-сайтів .

Залежно від ступеня деталізації і часу зберігання у сховищі виділяються поточні детальні дані, архівні дані, агреговані (сумарні, узагальнені) дані, метадані (репозиторій).

На відміну від баз даних транзакційних систем, де агреговані дані не зберігаються, а кожен раз обчислюються заново, сховище містить і детальні, і агреговані дані. Це обумовлено необхідністю забезпечення швидкого виконання запитів користувачів: в сховищі міститься така велика кількість

даних, що обчислення сумарних показників «на льоту» займало б значну кількість часу.

У сховищі міститься інформація з різних джерел, яка може мати різну періодичність оновлення, різну структуру, ступінь достовірності, власників даних - відомості про ці характеристики інформації називаються метаданими і зберігаються в репозиторії сховища. У репозиторії можуть також зберігатися бізнес-терміни, правила та алгоритми обчислення показників, які визначені для даного бізнесу. Фізично репозиторій є окремою базу даних або це набір таблиць в рамках бази даних сховища.

Сховище може бути реалізовано у вигляді віртуального сховища даних, вітрин даних та глобального сховища даних.

Під віртуальним сховищем даних розуміють спеціальні засоби доступу до даних транзакційних систем, що забезпечують роботу з цими даними як зі сховищем даних. Цими засобами доступу можуть бути як «уявлення» в базі даних, так і окремі програмні продукти. Перевагами віртуального сховища є простота і низька ціна реалізації, єдина платформа з джерелом інформації, відсутність необхідності перевантаження даних з джерел інформації в сховищі даних. До недоліків такого підходу відносяться проблеми продуктивності, трансформації даних, інтеграції даних з іншими джерелами, відсутність підтримки хронології, перевірки коректності даних, залежність від доступності та структури основної бази даних.

Реалізація сховища даних на основі вітрин даних передбачає функціонування двох рівнів: рівня джерел даних і рівня вітрин даних, які будуються на основі принципів проектування сховищ даних і містять дані конкретної вузької предметної області. В рамках одного підприємства вітрин даних може бути кілька: вітрина даних по постачальниках, вітрина даних по вироблених товарах, вітрина даних по доходах і видатках для бухгалтерії та ін. Єдине центральне сховище даних при цьому не створюється. Перевагами вітрин даних є простота і низька ціна реалізації в порівнянні зі створенням централізованого сховища даних, висока продуктивність, виділення

завантаження і трансформації даних в окремий процес, оптимізований під аналіз структури зберігання даних. Вітрини даних також дозволяють підтримувати хронологію даних, описувати структуру даних у вигляді метаданих. До недоліку вітрин даних можна віднести те, що вони не дають єдиного джерела інформації про все підприємство. Згодом інтегрувати вітрини в єдине централізоване сховище може виявитися проблематичним через різницю форматів і структур зберігання даних. Крім того, різні вітрини можуть використовувати частково дані, що повторюються, які потрібно вилучати з джерела для кожної вітрини окремо, що вимагає додаткових витрат на обслуговування.

Глобальне сховище даних передбачає реалізацію трирівневої архітектури системи. На першому рівні розташовуються джерела даних - внутрішні транзакційні системи, зовнішні джерела. Другий рівень містить центральне сховище, в яке завантажується інформація з джерел даних. При різному регламенті надходження даних з джерел в якості проміжної ланки може використовуватись оперативний склад даних, в якому дані готуються, перетворюються, перевіряються для їх подальшого завантаження в центральне сховище. Описи завантажених даних поміщаються в репозиторій. Третій рівень являє собою набір предметно-орієнтованих вітрин даних, джерелом інформації для яких є центральне сховище даних. Саме з вітринами даних і працює більшість кінцевих користувачів.

Концептуально організацію сховища даних можна представити у вигляді схеми на рисунку 2.1.

В основі побудови сховища даних лежить принцип багатовимірного представлення даних, при якому в структурі інформації виділяються вимірювання і факти. Під вимірами розуміються категоріальні (дискретні) атрибути, найменування і властивості об'єктів, що беруть участь в бізнес-процесі, наприклад, найменування підприємств, назви товарів, регіонів, магазинів. Факти - це кількісні значення показників, що описують бізнес-процес. Прикладами фактів можуть бути ціни на товари, обсяг продажів, обсяг

доходів, обсяг витрат, рентабельність, частка на ринку, оцінки студентів в процесі навчання.

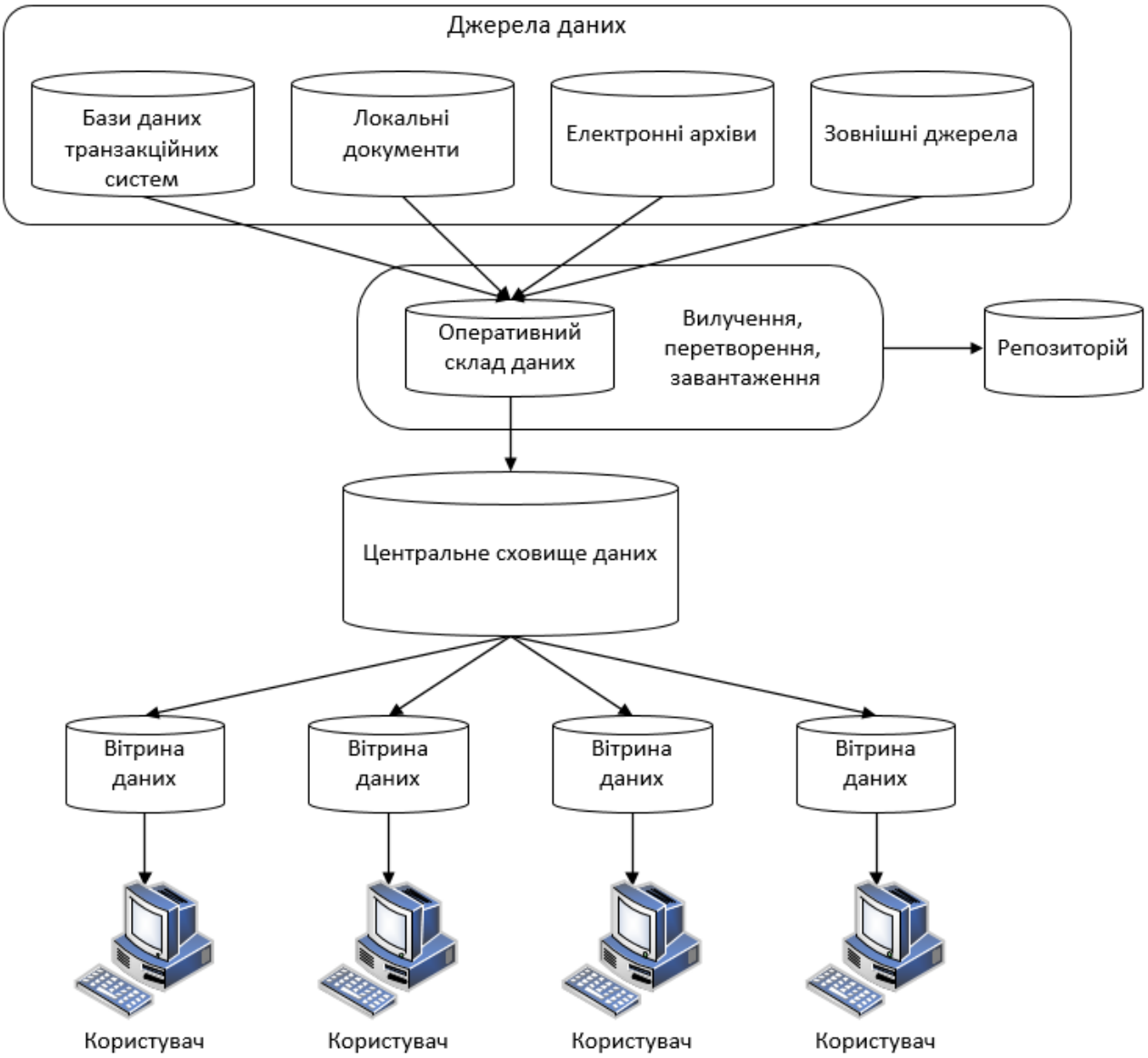


Рис. 2.1 Схема організації сховища даних

Відповідно до принципу багатовимірного представлення даних в сховищі даних виділяються таблиці фактів, таблиці вимірів і консольні таблиці. У таблицях фактів містяться кількісні значення показників з посиланнями на результати вимірювання, до яких вони належать. У таблицях вимірів (довідниках) зберігаються всі можливі значення вимірів. Консольні таблиці можуть використовуватися для зберігання більш складних вимірів з вкладеністю і ієрархією.

Наприклад, якщо у сховищі повинні міститись дані про успішність навчання студентів, то дані про дисципліни, що викладаються, спеціальності, освітні ступені будуть зберігатись у відповідних таблицях вимірів "Дисципліна", "Спеціальність", "Освітній ступінь", а кількісні значення продажів - в таблиці фактів "Аналіз успішності". До таблиць вимірів можуть бути приєднані консольні таблиці.

Залежно від складності предметної області таблиці бази даних сховища можуть бути пов'язані за схемою а - «зірка»; б - «сніжинка»; в - «сузір'я» (рис. 2.2).

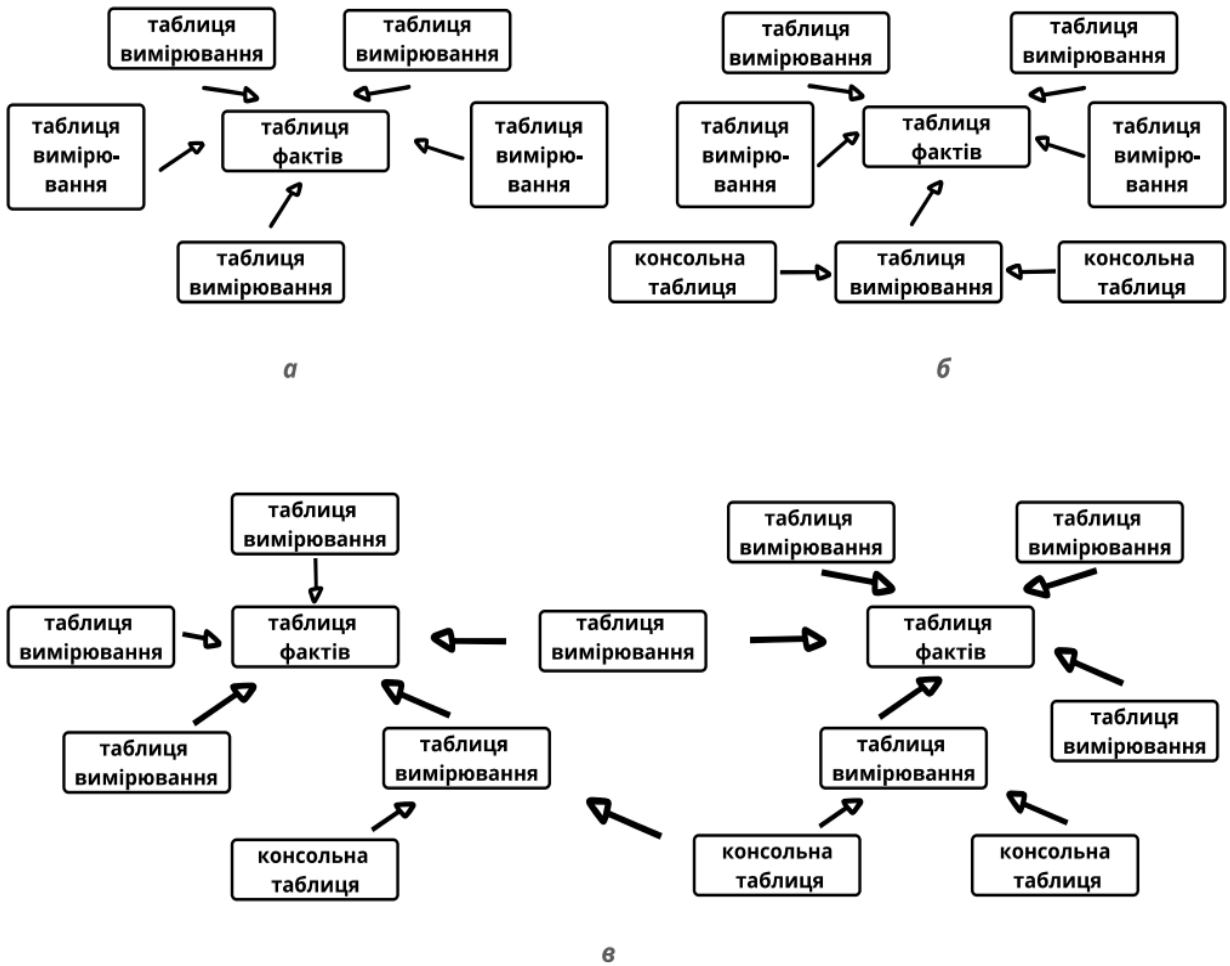


Рис. 2.2 Схеми побудови сховищ даних:

В схемі «зірка» одна таблиця фактів зв'язується з декількома таблицями вимірів. Схема «сніжинка» передбачає додаткові зв'язки таблиць вимірів з консольними таблицями. Якщо в сховищі даних присутні декілька таблиць

фактів, які використовують загальні таблиці вимірювань і консольні таблиці, то сховище побудоване за схемою «сузір'я».

Технологічно сховища даних тісно пов'язані із засобами оперативної аналітичної обробки даних (OLAP-технологіями), що дозволяють аналітикам, керівникам і керівникам вищої ланки вивчати великі обсяги взаємопов'язаних даних за допомогою швидкого інтерактивного відображення інформації на різних рівнях деталізації[6].

Для реалізації задач описаних в розділі 1, табл. 2.1 розроблено сховище даних, що містить деталізовану і загреговану інформацію про навчання студентів ВНЗ. Структура даних на рівні визначень наведена на рисунку 2.3.

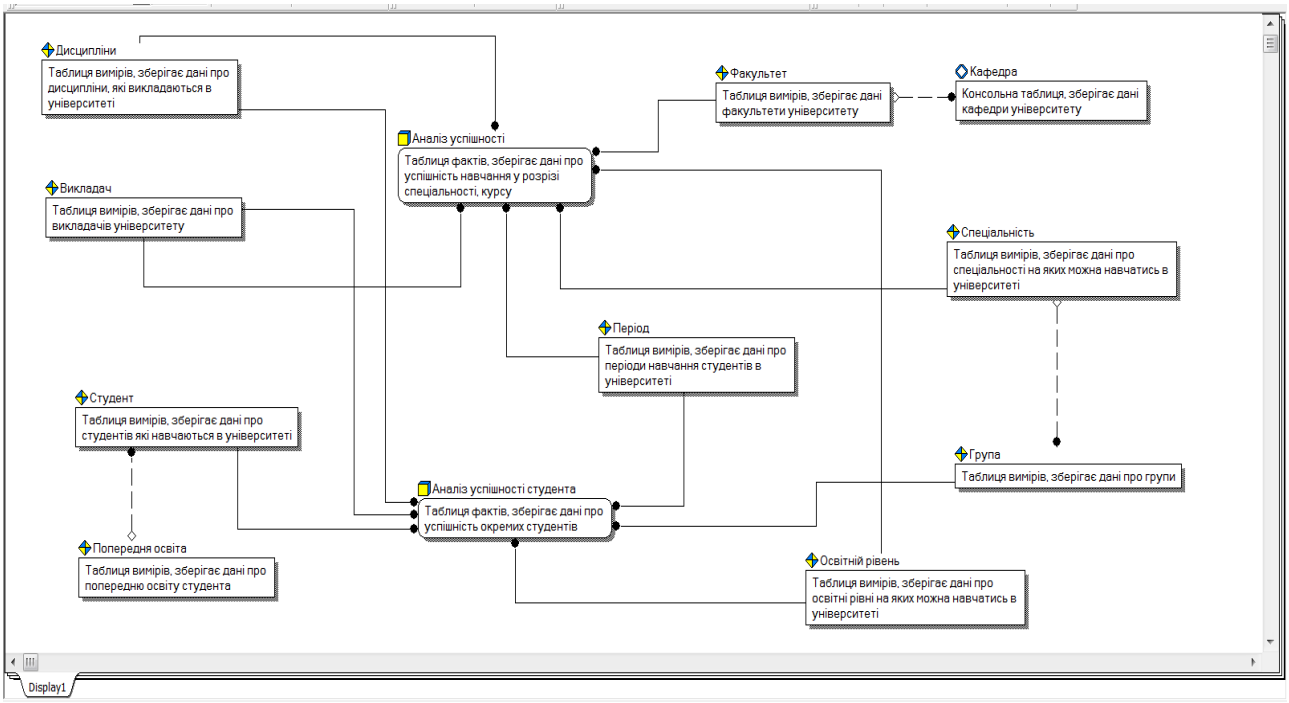


Рис. 2.3 Структура сховища даних на рівні визначень.

Схема спроектованого СД – "сніжинка". Сховище складається з двох таблиць фактів, а дев'яти таблиць вимірів і консольної таблиці.

Таблиці вимірів призначені для збереження інформації про дисципліни, що викладаються, викладачів, студентів, студентські групи, факультеті і кафедри, тощо.

Таблиця фактів "Аналіз успішності студента" містить деталізовані дані про успішність кожного студента з дисциплін, що викладаються.

Таблиця фактів "Аналіз успішності" містить загреговані дані про навчання в розрізі академічних груп і курсів.

Структура сховища на рівні визначень наведена у додатку А, на рівні атрибуті – у додатку Б.

2.3. Застосування Data Mining для аналізу освітніх даних

Інтелектуальний аналіз освітніх даних (Educational Data Mining) є цікавим напрямом дослідження, який призначений для отримання корисних, раніше невідомих закономірностей з навчальних баз даних для кращого розуміння і поліпшення успішності, оцінки процесу навчання студентів, прийняття ефективних управлінських рішень [7]. Застосування методів інтелектуального аналізу даних до освітніх баз даних дозволить підвищити ефективність системи вищої освіти [8]. Неявна інформація, отримана від видобутку освітніх наборів даних може бути застосована наприклад, для точного передбачення підсумкової оцінки студентів, зниження відсіву студентів, для класифікації студентів за додатковими предметами, які найбільш їм підходять тощо. Використання Educational Data Mining допоможе особам, які приймають рішення (ОПР) виявити асоціації, шаблони і тенденції, які сприятимуть до поліпшенню освітніх процесів.

Основна мета інтелектуального аналізу даних – повністю автоматичне або напівавтоматичне знаходження в зібраних даних залежностей, що представляють практичну цінність в контексті сфери застосування даної технології. Методи інтелектуального аналізу даних поділяються на три групи [9,10]: пошуковозалежні (discovery), прогнозування (predictive modelling) і аналіз аномалій (forensic analysis). Пошук залежностей полягає в перегляді бази даних з метою автоматичного виявлення залежностей. Проблема тут полягає у відборі дійсно важливих залежностей з величезного числа даних існуючих в базі даних. Прогнозування передбачає, що користувач може пред'явити системі записи з незаповненими полями і запросити відсутні значення. Система сама аналізує вміст бази і робить правдоподібне пророкування щодо цих значень.

Аналіз аномалій – це процес пошуку даних, які сильно відхиляються від стійких залежностей. Технології інтелектуального аналізу даних дозволяють вирішувати безліч завдань із залученням методів математичної статистики і теорії ймовірності, а також методів штучного інтелекту. Найбільшого поширення знайшли методи, що дозволяють вирішувати наступні завдання:

- Класифікація – віднесення об'єкта (події, предмета) до одного із заздалегідь відомих класів за його характеристикам;
- Регресія – прогнозування значення будь-якого вихідного параметра об'єкта по набору вхідних параметрів;
- Кластеризація – завдання полягає в поділі об'єктів на кластери за значеннями, які притаманні об'єктам параметрів. Вирішення цього завдання допомагає краще зрозуміти дані;
- Пошук асоціативних правил – виявлення закономірностей між якими-небудь пов'язаними об'єктами. Вирішення цього завдання допомагає краще зрозуміти природу аналізованих даних і може служити для прогнозування появи подій.
- Прогнозування послідовностей – знаходження залежностей між об'єктами або подіями у формі правил, що вказують, після якої події А настає подія В;
- Аналіз відхилень – аналіз даних на предмет входження явних нехарактерних шаблонів.

Проблеми аналізу і моделювання освітнього процесу у ВНЗ формулюються схожим чином, і вирішення більшості з них зводиться до тієї чи іншої задачі інтелектуального аналізу даних або до їх комбінації[11].

Для розв’язання освітніх завдань з використанням методів інтелектуального аналізу даних в роботі використано програмний пакет MS Analysis Services – сучасний інструмент аналізу даних, в якому реалізовані різні методи Data Mining. Це вільно поширюваний програмний пакет з відкритим вихідним кодом для аналізу даних, що являє собою набір засобів візуалізації і алгоритмів для інтелектуального аналізу даних.

У роботі досліджено можливості застосування методів інтелектуального аналізу даних для прогнозування і аналізу студентів різних спеціальностей. Для практичного розв’язання поставлених завдань застосовуються різні методи Data Mining, зокрема дерева рішень, кластер ний аналіз, прогнозування методом часових рядів.

2.4. Висновки до розділу 2.

В другому розділі здійснено виділення задач аналізу навчання студентів ВНЗ, які можна вирішити методами Data Mining.

Досліджено і описано застосування Data Mining для вирішення задач навчання студентів ВНЗ.

Описано сховище даних, як джерело для збереження аналітичної інформації.

На основі проведених досліджень здійснена постанова задачі аналізу навчання студенів ВНЗ засобами Data Mining.

Розділ 3. Аналіз освітніх даних на основі технології Data Mining

3.1. Постанова задач аналізу успішності навчання студентів НУХТ

Національний університет харчових технологій – це навчальний заклад із великим науковим потенціалом. Широко відомі в Україні і за її межами 35 наукових шкіл університету за 16 тематичними напрямами наукових досліджень і науково-технічних розробок. Збагачені сучасним змістом, нині вони тісно пов’язані з науково-технічним прогресом і визначають його пріоритети у галузі харчових технологій.

В університеті велику увагу приділяють аналізу якості навчання, що відображено на сайті НУХТ, сайтах інститутів, факультетів та кафедр. Впроваджене анонімне анкетування студентів з різних організаційних та освітніх питань. Окрім традиційних форм навчання в університеті впроваджена дистанційна платформа навчання, а діяльність структурних підрозділів забезпечує єдина інформаційна система. Таким чином, за роки використання інформаційних систем в університеті накопичений унікальний масив реальних даних як про успішність навчання так і про іншу діяльність ВНЗ.

Досліджуючи похідний від інтелектуального аналізу даних напрямок – інтелектуальний аналіз даних навчального процесу (англ. Educational Data Mining) розглянемо реалізацію, наведених в розділі 2, табл. 2.1. задач на прикладі аналізу навчання студентів НУХТ.

3.2. Реалізація задач аналізу успішності навчання студентів НУХТ засобами Data Mining

З метою реалізації задач, наведених у розділі 2, табл. 2.1. проаналізуємо навчання студентів НУХТ на основі використання методів кластеризації та дерев рішень Data Mining.

Розглянемо вплив різних факторів на успішність навчання студентів на прикладі аналізу успішності академічних груп бакалаврів 1–4 курсів спеціальності 122 "Комп'ютерні науки".

Для аналізу успішності навчання згрупуємо студентські курси у кластери і розглянемо різноманітні параметри, які супроводжують процес навчання . Наприклад, наявність пропусків занять з поважних/неповажних причин, успішність навчання по місяцях,семестрах тощо.

Джерелом інформації для проведення аналізу даних є сховище даних (див. Додаток В).

Групування даних здійснимо методом кластерного аналізу Data Mining.

Вихідною інформацією для проведення кластеризації є початкова множина даних, отримана зі сховища даних.

Позначимо через J множину оцінок по кожному курсу навчання:

$J = \{j_1, j_2, \dots j_i, \dots, j_n\}$, де j_i – оцінки студентів певного курсу з кожної дисципліни.

Необхідно побудувати множину кластерів K та відображення E множини J на множину K, тобто $E: J \to K$.

Відображення E задає модель даних, що є рішенням задачі. Якість рішення задачі визначається кількістю вірно кваліфікованих даних.

Кожен курс визначається набором характеристик, які супроводжують процес навчання та даними про отримані оцінки:

$j_i = \{z_1, z_{2, \dots,} z_h, \dots, z_m\}$,

де z_m – параметри, які аналізуються в процесі навчання (дисципліна, семестр, курс, місяць, семестр, пропуски занять).

Кожна змінна z_h може приймати значення з деякої множини значень характеристик:

$z_h = \{\nu_h^1, \nu_h^2, \dots\}$

Задача кластеризації полягає у побудові множини:

$K = \{k_1, k_2, \dots, k_l, \dots, k_g\}$,

де k_l – кластер, який містить схожі за характеристиками курси з множини J:

$k_l = \{j_i, j_p \mid j_i \in J, j_p \in J$ та $d(j_i, j_p) < \sigma\}$,

де σ - величина, яка визначає міру близькості для включення об'єктів в один кластер;

$d(j_i, j_p)$ – відстань між об'єктами.

Невід'ємне значення $d(j_i, j_p)$ є відстанню між елементами j_i та j_p, якщо виконуються наступні умови:

$d(j_i, j_p) \geq 0$, для всіх j_i та j_p;

$d(j_i, j_p) = 0$, тоді і тільки тоді, коли $j_i = j_p$;

$d(j_i, j_p) = d(j_p, j_i)$;

$d(j_i, j_p) \leq d(j_i, j_r) + d(j_r, j_p)$

Якщо відстань $d(j_i, j_p)$ менше деякого значення σ, то елементи, які характеризують курси, близькі і розміщуються в одному кластері. В іншому випадку кажуть, що елементи відмінні один від одного та їх розміщують у різні кластери [9].

Використовуючи надбудову "Інтелектуальний аналіз даних" MS Excel здійснемо кластерний аналіз даних. Модель кластерного аналізу наведена на рис. 3.1

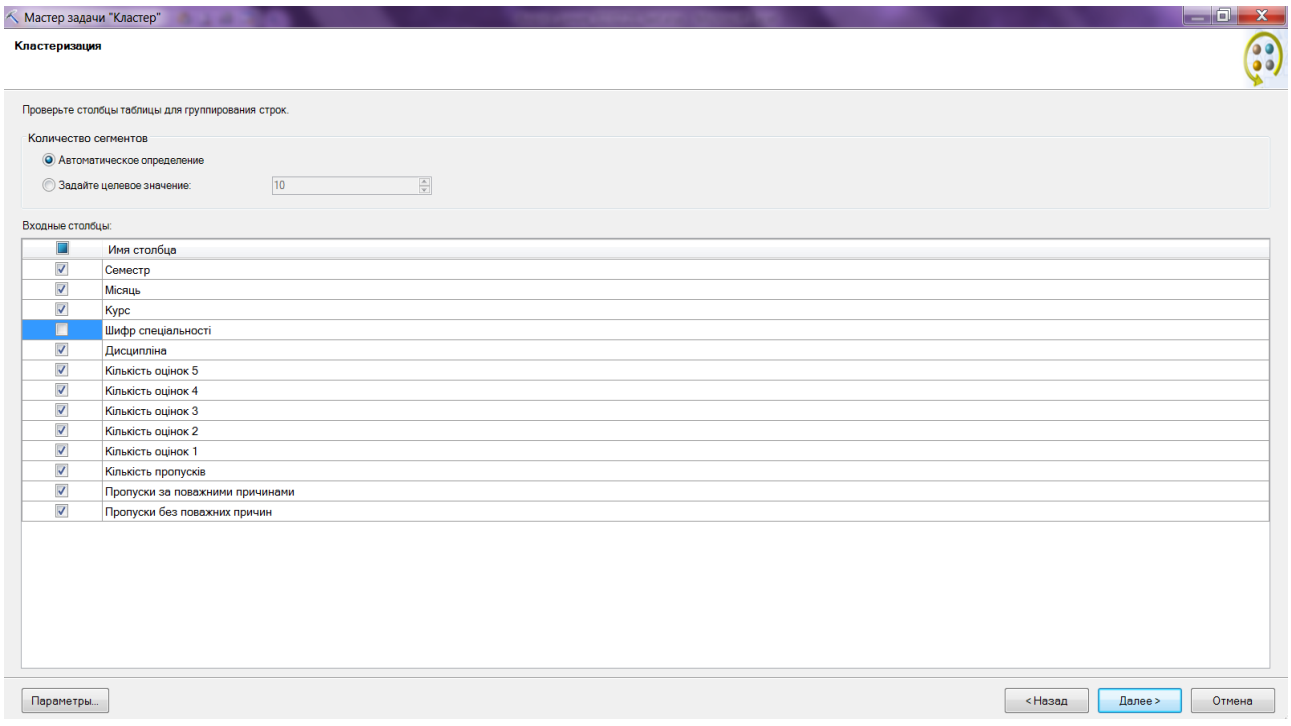


Рис. 3.1 Модель кластерного аналізу в MS Analysis Services

В результаті кластеризації вихідних даних отримана результуюча модель, яка складається з діаграми кластерів (рис. 3.2) і профілів кластерів (рис. 3.3 – 3.4).

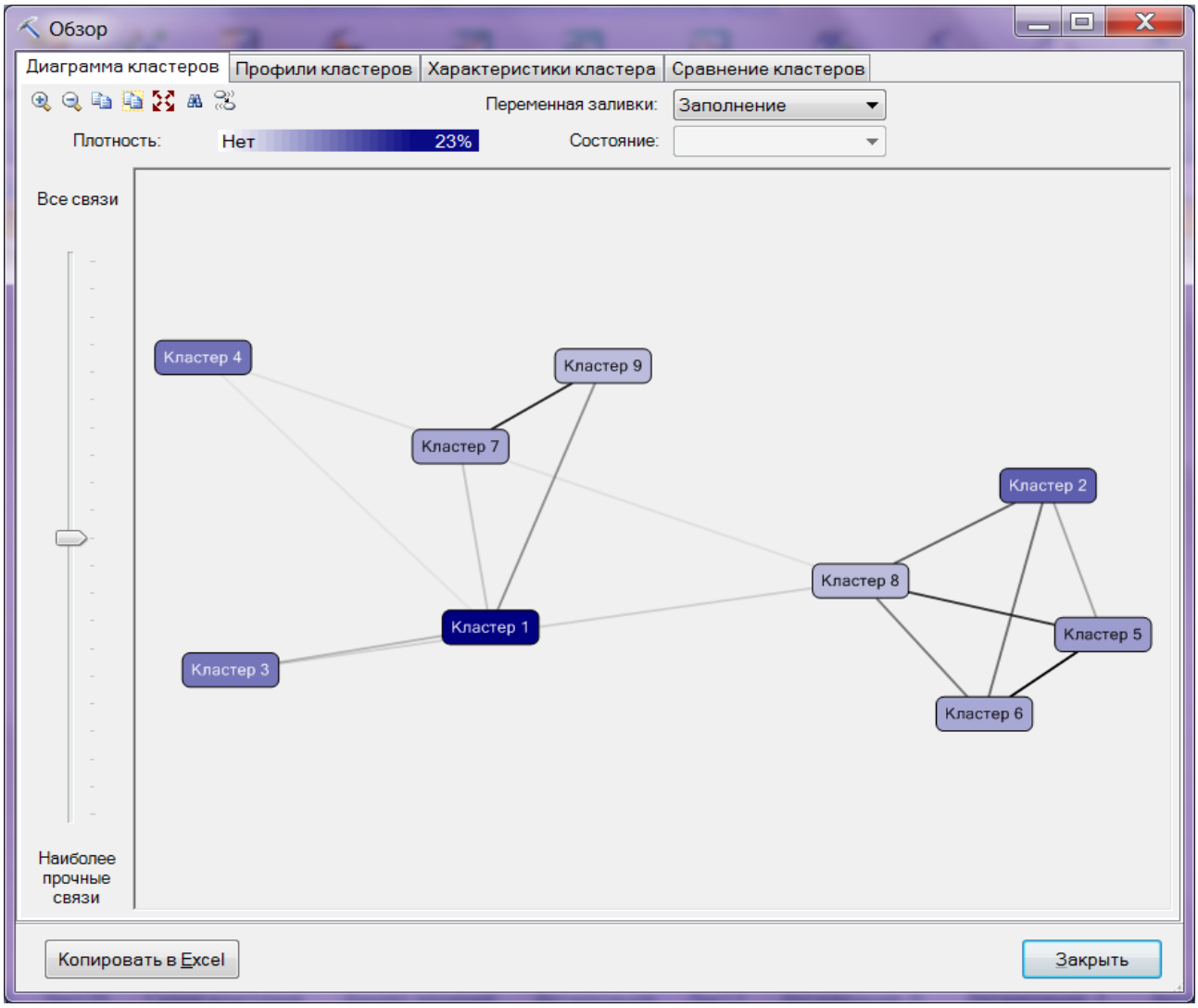


Рис. 3.2. Діаграма кластерів

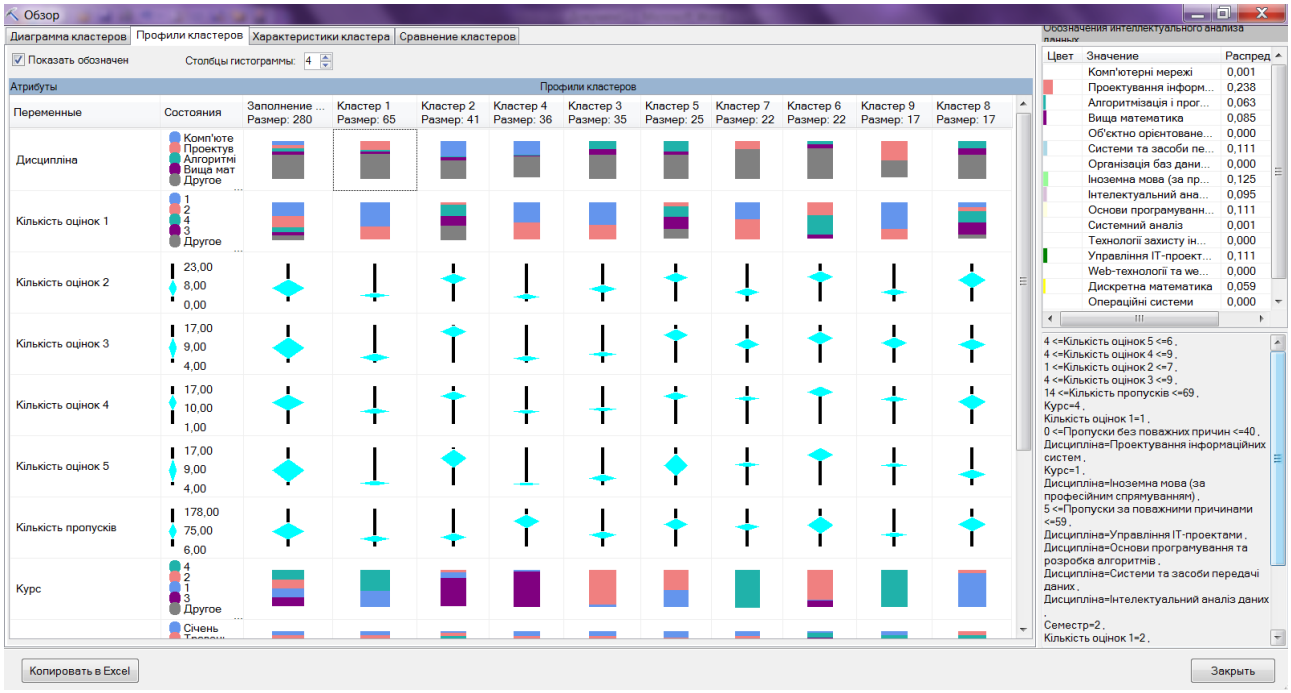


Рис. 3.3 а. Профілі кластерів (початок)

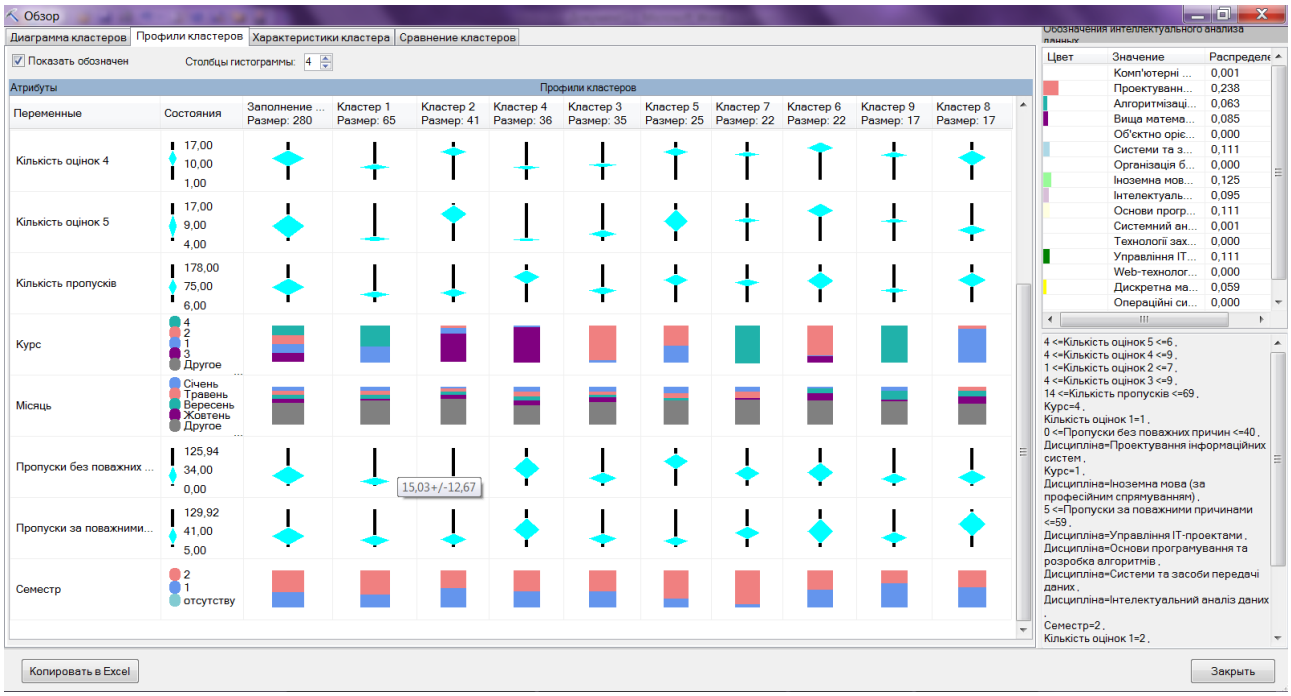
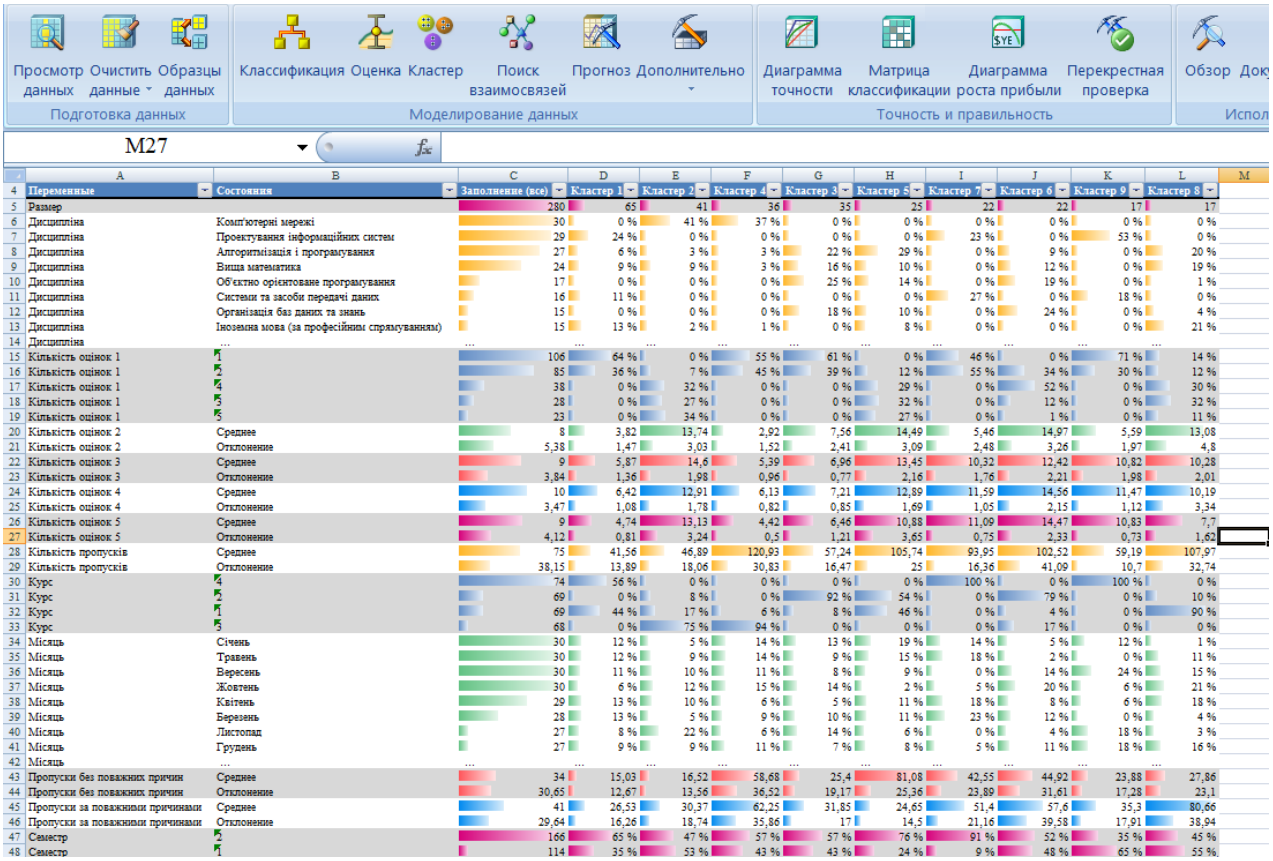


Рис. 3.3. б. Профілі кластерів (продовження)

Детальніше кластери можна проаналізувати завантаживши їх до таблиці MS Excel:



	A	B	C	D	E	F	G	H	I	J	K	L
4	Переменные	Состояния	Заполнение (все)	Кластер 1	Кластер 2	Кластер 4	Кластер 3	Кластер 5	Кластер 7	Кластер 6	Кластер 9	Кластер 8
5	Размер		280	65	41	36	35	25	22	22	17	17
6	Дисципліна	Комп'ютерні мережі	30	0 %	41 %	37 %	0 %	0 %	0 %	0 %	0 %	0 %
7	Дисципліна	Проектування інформаційних систем	29	24 %	0 %	0 %	0 %	0 %	23 %	0 %	53 %	0 %
8	Дисципліна	Алгоритмізація і програмування	27	6 %	3 %	3 %	22 %	29 %	0 %	9 %	0 %	20 %
9	Дисципліна	Вища математика	24	9 %	9 %	3 %	16 %	10 %	0 %	12 %	0 %	19 %
10	Дисципліна	Об'єктно орієнтоване програмування	17	0 %	0 %	0 %	25 %	14 %	0 %	19 %	0 %	1 %
11	Дисципліна	Системи та засоби передачі даних	16	11 %	0 %	0 %	0 %	0 %	27 %	0 %	18 %	0 %
12	Дисципліна	Організація баз даних та знань	15	0 %	0 %	0 %	18 %	10 %	0 %	24 %	0 %	4 %
13	Дисципліна	Іноземна мова (за професійним спрямуванням)	15	13 %	2 %	1 %	0 %	8 %	0 %	0 %	0 %	21 %
14	Дисципліна	...	...	...	...	...	...	...	...	...	...	...
15	Кількість оцінок 1	1	106	64 %	0 %	55 %	61 %	0 %	46 %	0 %	71 %	14 %
16	Кількість оцінок 1	2	85	36 %	7 %	45 %	39 %	12 %	55 %	34 %	30 %	12 %
17	Кількість оцінок 1	4	38	0 %	32 %	0 %	0 %	29 %	0 %	52 %	0 %	30 %
18	Кількість оцінок 1	3	28	0 %	27 %	0 %	0 %	32 %	0 %	12 %	0 %	32 %
19	Кількість оцінок 1	5	23	0 %	34 %	0 %	0 %	27 %	0 %	1 %	0 %	11 %
20	Кількість оцінок 2	Среднее	8	3,82	13,74	2,92	7,56	14,49	5,46	14,97	5,59	13,08
21	Кількість оцінок 2	Отклонение	5,38	1,47	3,03	1,52	2,41	3,09	2,48	3,26	1,97	4,8
22	Кількість оцінок 3	Среднее	9	5,87	14,6	5,39	6,96	13,45	10,32	12,42	10,82	10,28
23	Кількість оцінок 3	Отклонение	3,84	1,36	1,98	0,96	0,77	2,16	1,76	2,21	1,98	2,01
24	Кількість оцінок 4	Среднее	10	6,42	12,91	6,13	7,21	12,89	11,59	14,56	11,47	10,19
25	Кількість оцінок 4	Отклонение	3,47	1,08	1,78	0,82	0,85	1,69	1,05	2,15	1,12	3,34
26	Кількість оцінок 5	Среднее	9	4,74	13,13	4,42	6,46	10,88	11,09	14,47	10,83	7,7
27	Кількість оцінок 5	Отклонение	4,12	0,81	3,24	0,5	1,21	3,65	0,75	2,33	0,73	1,62
28	Кількість пропусків	Среднее	75	41,56	46,89	120,93	57,24	105,74	93,95	102,52	59,19	107,97
29	Кількість пропусків	Отклонение	38,15	13,89	18,06	30,83	16,47	25	16,36	41,09	10,7	32,74
30	Курс	4	74	56 %	0 %	0 %	0 %	0 %	100 %	0 %	100 %	0 %
31	Курс	2	69	0 %	8 %	0 %	92 %	54 %	0 %	79 %	0 %	10 %
32	Курс	1	69	44 %	17 %	6 %	8 %	46 %	0 %	4 %	0 %	90 %
33	Курс	3	68	0 %	75 %	94 %	0 %	0 %	0 %	17 %	0 %	0 %
34	Місяць	Січень	30	12 %	5 %	14 %	13 %	19 %	14 %	5 %	12 %	1 %
35	Місяць	Травень	30	12 %	9 %	14 %	9 %	15 %	18 %	2 %	0 %	11 %
36	Місяць	Вересень	30	11 %	10 %	11 %	8 %	9 %	0 %	14 %	24 %	15 %
37	Місяць	Жовтень	30	6 %	12 %	15 %	14 %	2 %	5 %	20 %	6 %	21 %
38	Місяць	Квітень	29	13 %	10 %	6 %	5 %	11 %	18 %	8 %	6 %	18 %
39	Місяць	Березень	28	13 %	5 %	9 %	10 %	11 %	23 %	12 %	0 %	4 %
40	Місяць	Листопад	27	8 %	22 %	6 %	14 %	6 %	0 %	4 %	18 %	3 %
41	Місяць	Грудень	27	9 %	9 %	11 %	7 %	8 %	5 %	11 %	18 %	16 %
42	Місяць	...	...	...	...	...	...	...	...	...	...	...
43	Пропуски без поважних причин	Среднее	34	15,03	16,52	58,68	25,4	81,08	42,55	44,92	23,88	27,86
44	Пропуски без поважних причин	Отклонение	30,65	12,67	13,56	36,52	19,17	25,36	23,89	31,61	17,28	23,1
45	Пропуски за поважними причинами	Среднее	41	26,53	30,37	62,25	31,85	24,65	51,4	57,6	35,3	80,66
46	Пропуски за поважними причинами	Отклонение	29,64	16,26	18,74	35,86	17	14,5	21,16	39,58	17,91	38,94
47	Семестр	2	166	65 %	47 %	57 %	57 %	76 %	91 %	52 %	35 %	45 %
48	Семестр	1	114	35 %	53 %	43 %	43 %	24 %	9 %	48 %	65 %	55 %

Рис. 3.4. Деталізація профілів кластерів в MS Excel

Проаналізуємо характеристики, які входять до отриманих кластерів.

Аналізуючи кольорову гамму, яка відповідає кожному курсу можемо відмітити, що до кластерів 1, 5, 8 – входить найбільше об'єктів *1 курсу*; до кластерів 3, 6 – *2 курсу*, до кластерів 2, 4 – *3 курсу*; до кластерів 1, 7, 9.

Аналізуючи характеристики кластерів 1, 5, 8, що містять найбільше об'єктів *1 курсу* можна зазначити, що кількість пропусків на середньому рівні (порівняно з іншими курсами), більшість пропусків – з поважних причин і розподіляється рівномірно по місяцях та семестрах, успішність при цьому має приблизно однакову кількість позитивних та негативних оцінок.

Аналізуючи характеристики кластерів 3, 6, що містять найбільше об'єктів *2 курсу* звертає на себе увагу досить велика кількість пропусків на початку *1 семестру* (вересень, жовтень), в тому числі без поважних причин, при цьому кількість позитивних оцінок (5, 4, 3) – також досить велика, хоча й є негативні.

Аналізуючи характеристики кластерів 2, 4, що містять найбільше об'єктів *3 курсу* можна зазначити, що студенти загалом навчаються позитивно

(отримуючи оцінки 5, 4, 3), мають невелику кількість негативних оцінок та помірну кількість пропусків. При цьому кластер 2 – має великий розмір, характеризується великою кількістю позитивних оцінок (4, 5), задовільних оцінок і невеликою кількість пропусків, а кластер 4 містить невелику кількість позитивних оцінок і досить багато пропусків. Що говорить про вплив пропусків на загальну успішність.

Аналізуючи характеристики кластерів 1, 9, що містять найбільше об'єктів *4 курсу* можна зазначити, що кількість пропусків – коливається в порівнянні з іншими курсами на середньому рівні (приблизно однакова кількість пропусків з поважних і неповажних причин), при цьому кількість 5, 4 приблизно дорівнює кількості 3.

Таким чином, аналіз кластерів дав можливість проаналізувати вплив різноманітних факторів на успішність навчання студентів по курсах.

З метою подальшого детальнішого аналізу характеристик, які впливають на оцінки студентів побудуємо дерево рішень.

Метод дерева рішень виявляє причинно-наслідкову ієрархію умов для визначення характеристик, які впливають на отримання на основі наявних даних, виділяє найбільш суттєві з них та обмежує інші.

Глибше аналізуючи фактори впливу на навчання студентів доцільно побудувати дерева рішень, які дозволять з'ясувати:

- чи є залежність між успішністю навчання і пропусками занять?
- яка успішність навчання по курсах і які фактори впливають?
- яка успішність навчання по семестрах (місяцях)?

Наприклад, проаналізуємо, які характеристики найбільше впливають на отримання студентами задовільних оцінок.

Побудуємо дерева рішень для аналізу розподілу факторів, що супроводжують отримання задовільних оцінок методами *класифікації* та *оцінки*. Моделі дерев рішень подібні і наведені на рис. 3.5 та 3.6.

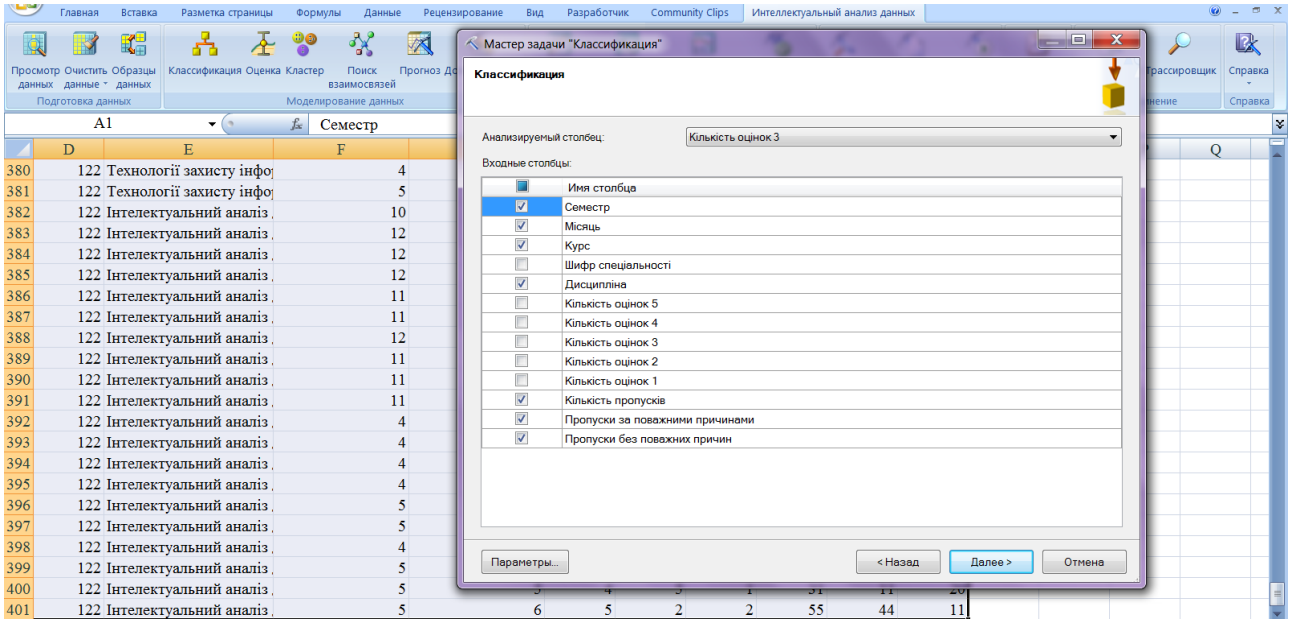


Рис. 3.5. Модель дерева рішень побудована методом класифікації

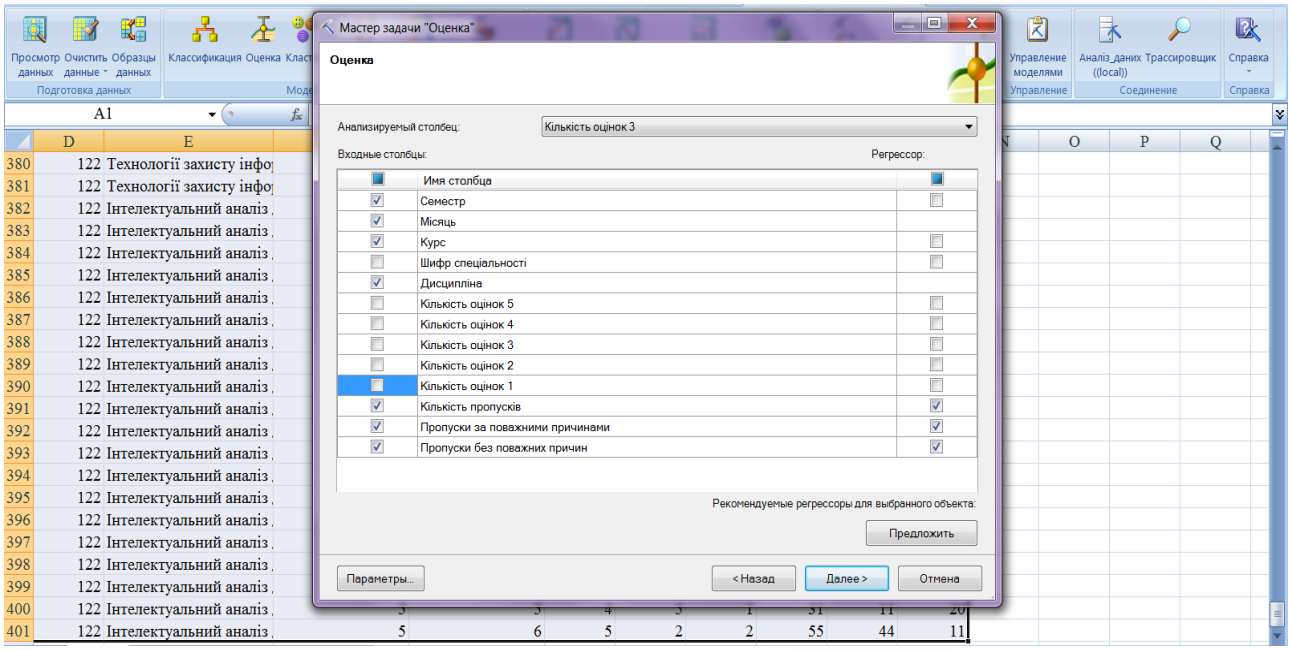


Рис. 3.6. Модель дерева рішень побудована методом оцінок

Аналізуючи побудовані дерева рішень можна визначити вплив пропусків з різних причин на навчання студентів в розрізі курсів, семестрів, дисциплін (рис. 3.7 – 3.9).

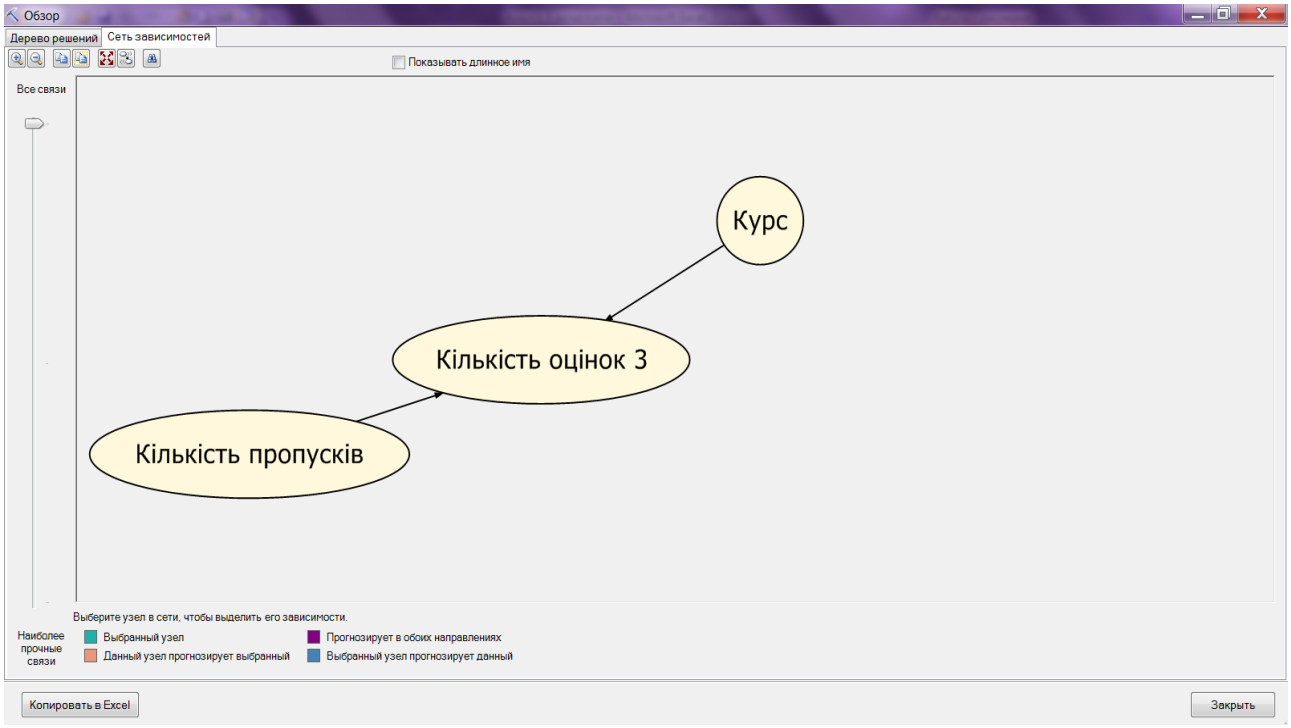
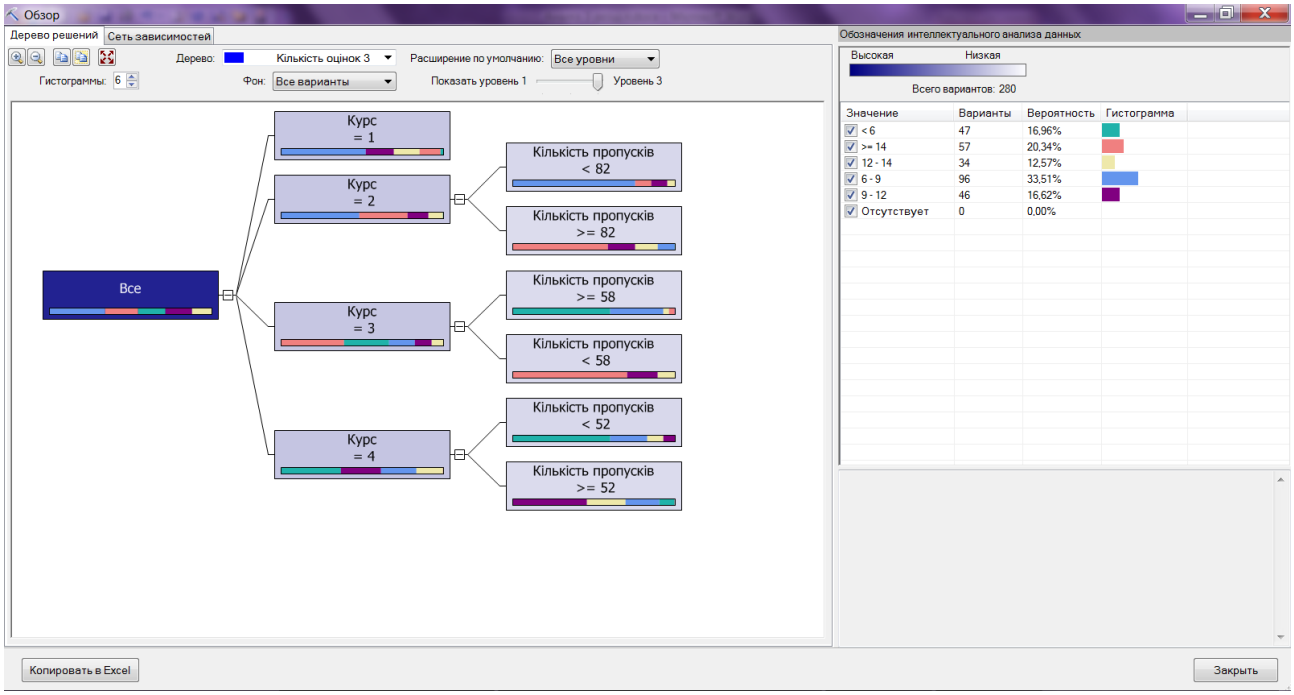


Рис. 3.7. Мережа залежностей впливу характеристик на кількість незадовільних оцінок.

Результати побудови дерев рішень наведені на рис. 3.8.



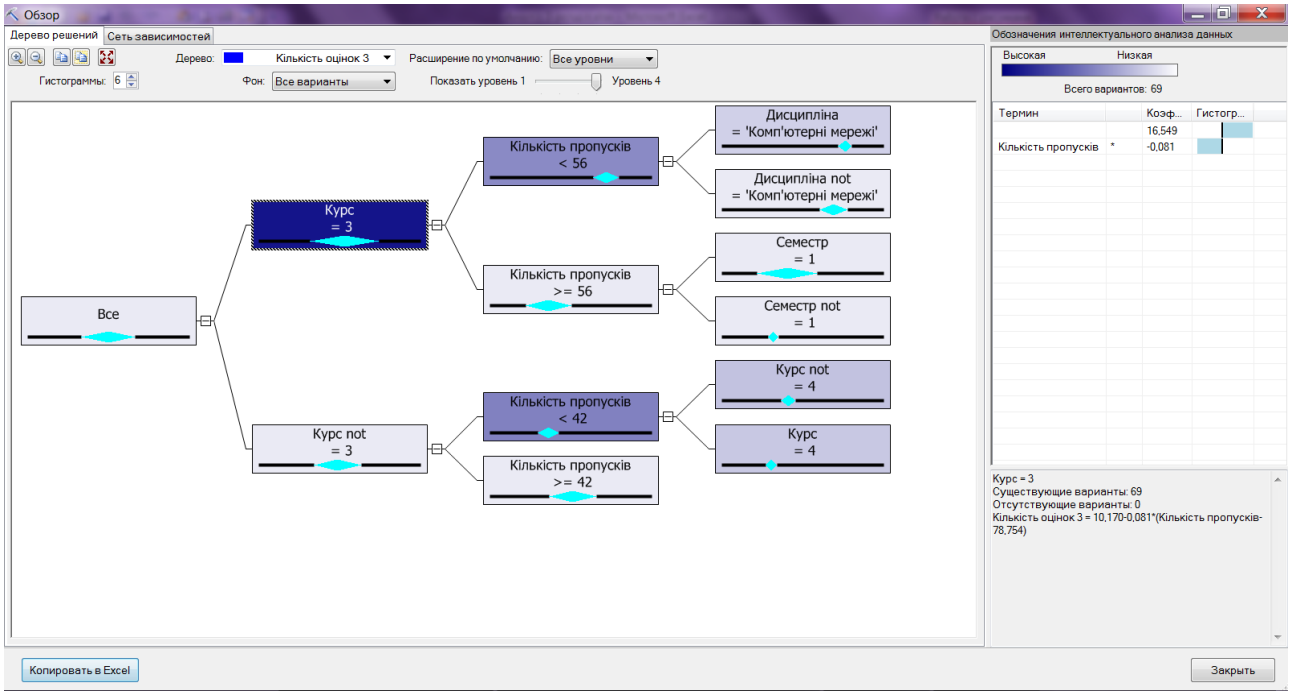


Рис.3.7 Дерева рішень для оцінки впливу різних факторів на успішність навчання

Побудувавши аналогічним способом модель дерева рішень отримаємо мережу залежностей факторів, що впливають на оцінки відмінно.

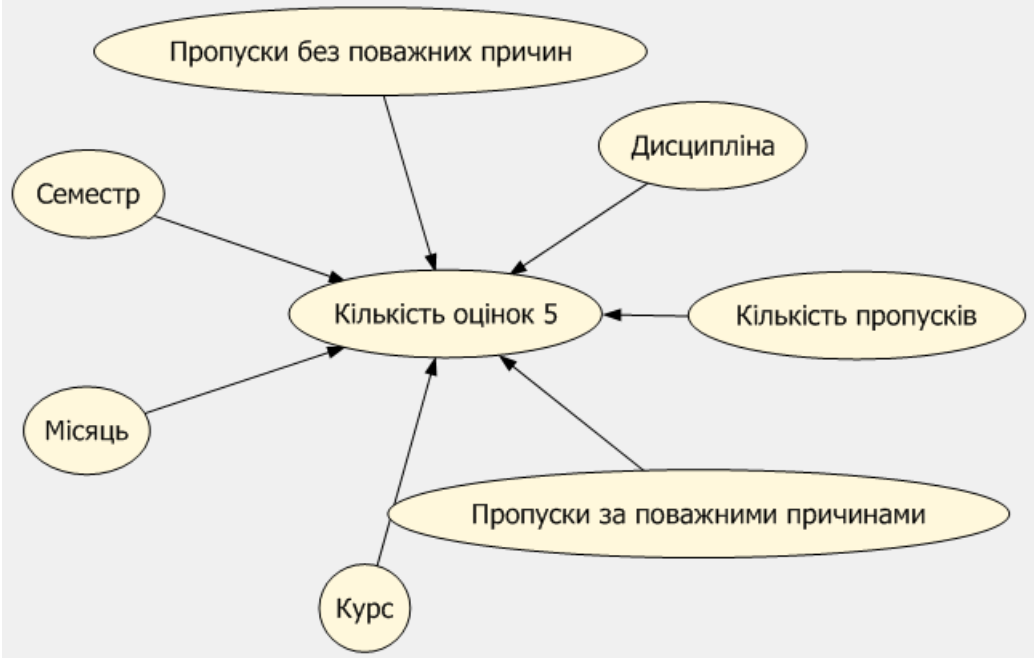


Рис. 3.8. Мережа залежностей впливу характеристик на кількість оцінок "5".

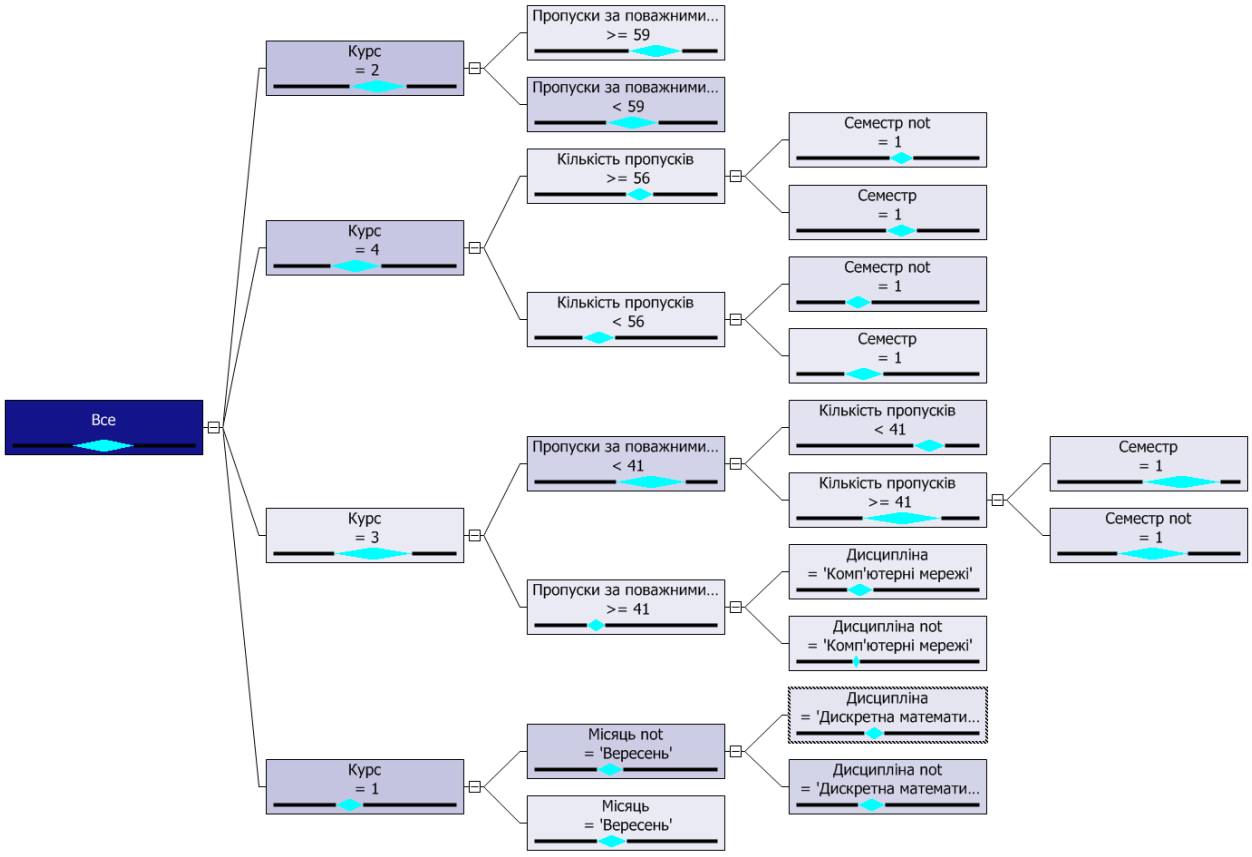


Рис.3.9. Дерево рішень для оцінки впливу різних факторів на кількість п'ятірок.

Проаналізувавши і виявивши проблеми можна сформувати рекомендації щодо їх усунення.

Детальніший аналіз факторів впливу можна здійснити на основі таблиці "Аналіз успішності студента" наявної у сховищі даних.

3.3. Прогнозування успішності навчання студентів НУХТ методами Data Mining.

Для вирішення задач прогнозування, наведених у розділі 2, табл. 2.1. використаємо накопичену у СД інформацію про навчання студентів у вигляді часових рядів. Прогнозування з використанням інтервальних часових рядів Xt={x1, x2, …, xn} (де t – момент часу, на який зареєстровано спостереження) дозволяє будувати короткострокові та довгострокові прогнози, які базуються на виявленні закономірностей та аналізі тенденцій попередніх періодів.

Наприклад, спрогнозуємо кількість відмінних оцінок з дисципліни "Вища математика" на наступний місяць.

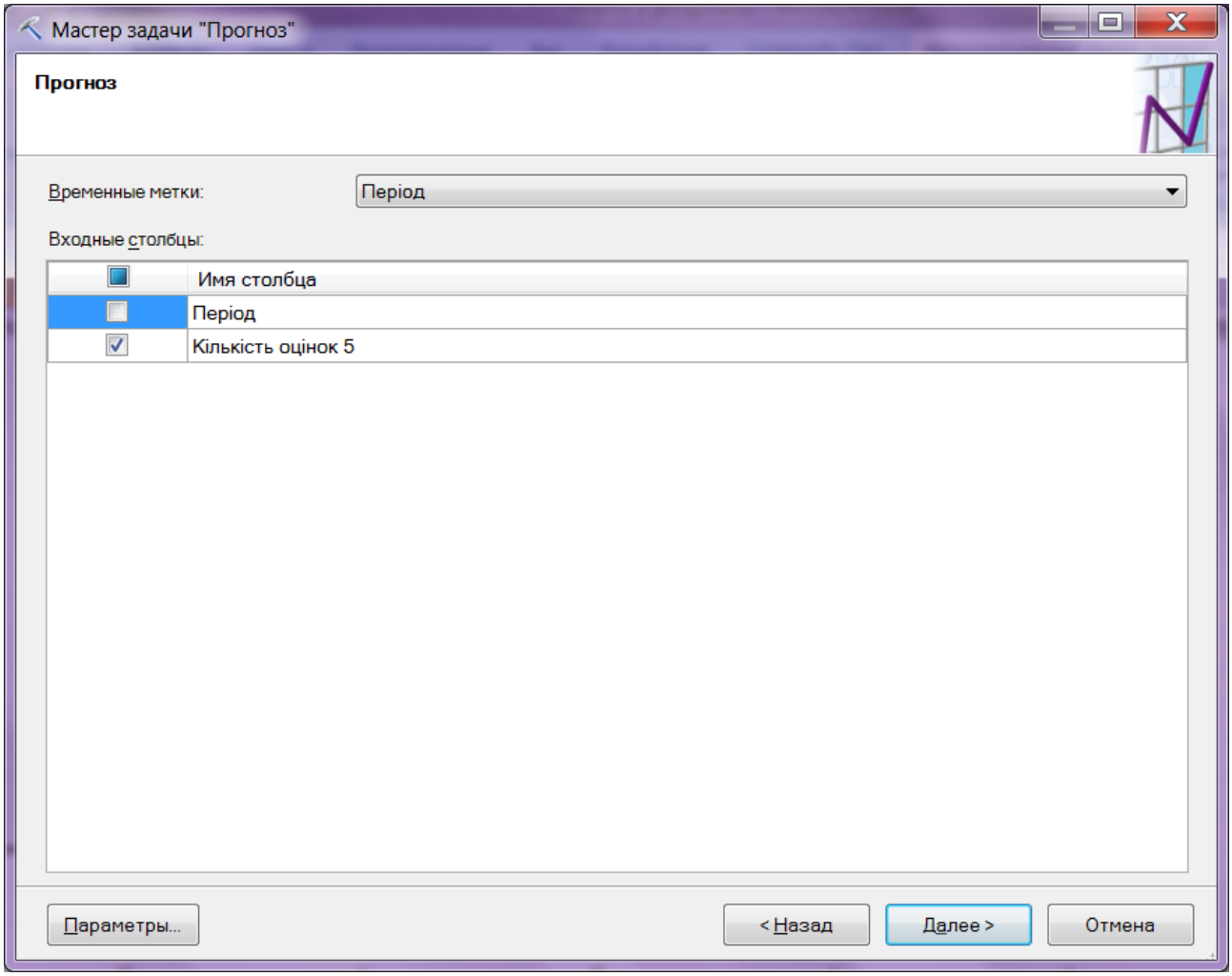


Рис. 3.10. Модель прогнозування оцінок "відмінно" методом часових рядів

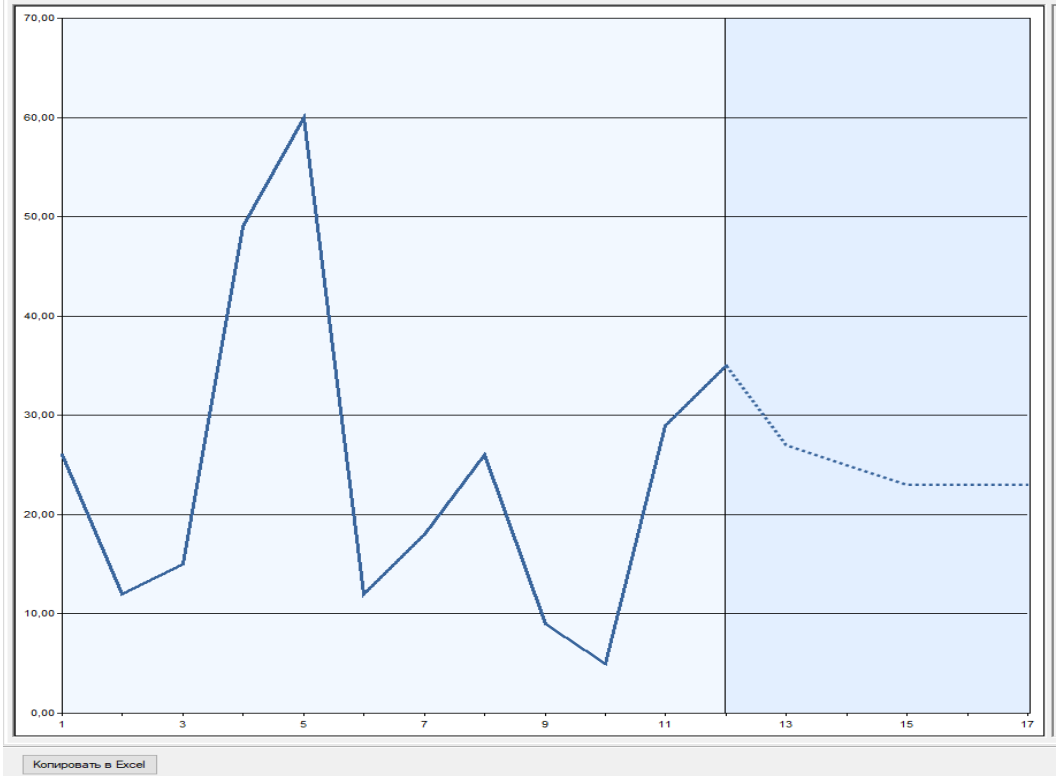


Рис. 3.11 Прогноз кількості оцінок "відмінно" з дисципліни "Вища математика"

В результаті прогнозування можемо передбачити, що в наступному місяці з дисципліни "Вища математика" студенти спец. 122 "Комп'ютерні науки" отримають 25 оцінок відмінно.

3.5. Висновок до розділу 3.

В розділі 3 наведений приклад реалізації аналізу освітніх даних засобами MS Analysis Services.

Джерелом аналітичних даних є сховище даних згенероване в MS SQL Server. Його структура враховує всю необхідну інформацію для реалізації задач, зазначених у таблиці 2.1 розділу 2.

В процесі реалізації задач аналізу використані алгоритми кластеризації, дерев рішень та прогнозування методом часових рядів.

Висновок

В кваліфікаційній роботі здійснено дослідження способів застосування методів Data Mining в задачах інформаційної підтримки діяльності ВНЗ.

1. Досліджено освітній процес ВНЗ на прикладі Національного університету харчових технологій.

2. Описано задачі аналізу успішності навчання студентів ВНЗ, які можна вирішити застосувавши технологію Data Mining.

3. Спроектовано та реалізовано реляційне сховище даних для проведення аналізу успішності навчання студентів ВНЗ.

3. Досліджено та адаптовано алгоритми Data Mining для вирішення задач аналізу навчання студентів ВНЗ.

4. Проаналізовано успішність навчання студентів НУХТ з використанням методів кластерного аналізу, дерев рішень, часових рядів технології Data Mining засобами MS Analysis Services.

Використання методів інтелектуального аналізу освітніх даних (Educational Data Mining) дозволяє спрогнозувати успішність навчання студентів, виявити асоціації, шаблони і тенденції, які сприятимуть поліпшенню освітніх процесів ВНЗ.

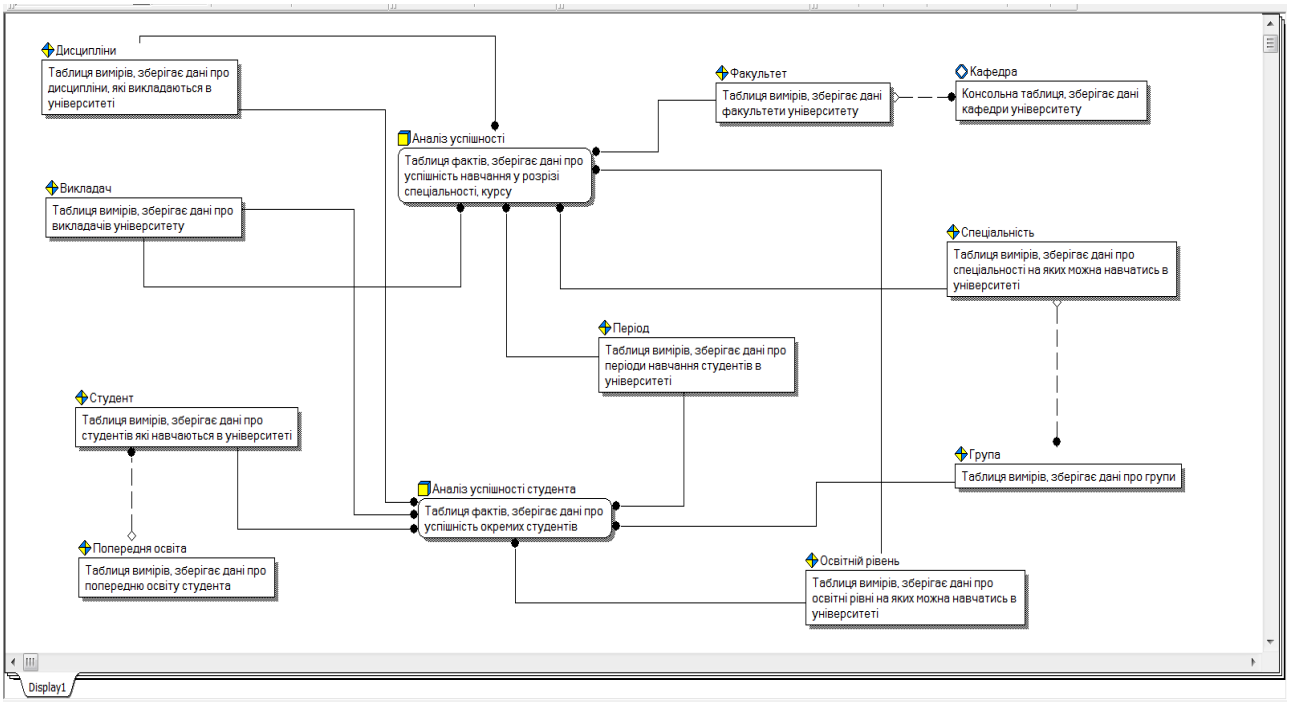
Список використаних джерел

1. Стариков А. Нейронные сети — математический апарат // Режим доступа: http://www.basegroup.ru/library/analysis/neural/math
2. Марченко О.О., Россада Т.В. Актуальні проблеми Data Mining: Навчальний посібник для студентів факультету комп'ютерних наук та кібернетики. — Київ. — 2017. — 150 с.
3. https://coderlessons.com/tutorials/bolshie-dannye-i-analitika/teoriia-khraneniia-dannykh/31-25-luchshikh-instrumentov-dlia-intellektualnogo-analiza-dannykh
4. О.І. Черняк, П.В. Захарченко / Інтелектуальний аналіз даних. Підручник, Київ-2010.
5. О.М. Верес, В.Л. Мельник, Л.Б. Чирун Застосування ms sql server 2005 для побудови інтелектуальної складової інформаційної систем. – Львів – 2008.
6. https://stud.com.ua/121123/informatika/shovischa_danih
7. Goyal, Monika. Applications of Data Mining in Higher Education. International journal of computer science, 2012, 9 (2), p. 113.
8. Програмний інструментарій Education Intelligence як засіб підвищення ефективності креативного навчання / В. Гришачов, Д. Замятін, О. Кебкало, А. Михайлюк, Л. Огнівчук, В. Тарасенко // Міжнародний науковий журнал "Комп'ютинг", 2011, Том 10, Випуск 2 - с. 114- 132.
9. Барсегян А. А. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP [Текст] / А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод. – 2-е изд., перераб. И доп. – СПб. : БХВ-Петербург, 2007. – 384 с.: ил. – CD-ROM.
10. Fayyad, U. Advances in Knowledge Discovery and DataMining / U. Fayyad, G. PiatetskyShapiro, P. Smyth, R. Uthurusamy.– AAAI/MIT Press, 1996.
11. Огнівчук Л.М. Застосування методів інтелектуального аналізу даних для розв'язання освітніх та управлінських завдань

12. Харкянен О.В. Інтелектуальний аналіз даних [Електронний ресурс] : навчальний посібник / О.В. Харкянен О.В., О.М. М’якшило, С.В. Грибков, – К.: НУХТ, 2019 – 170 с. : іл.
13. В. Р. Вергун. Характеристика методів розв'язання задачі класифікації в інтелектуальному аналізі даних навчальних програм -Національний університет "Львівська політехніка", м. Львів, Україна, 2019
14. Офіційний сайт Національного університету харчових технологій. Режим доступу: https://nuft.edu.ua/
15. Офіційний сайт ПП "Політек-СОФТ". Режим доступу: http://www.politek-soft.kiev.ua/

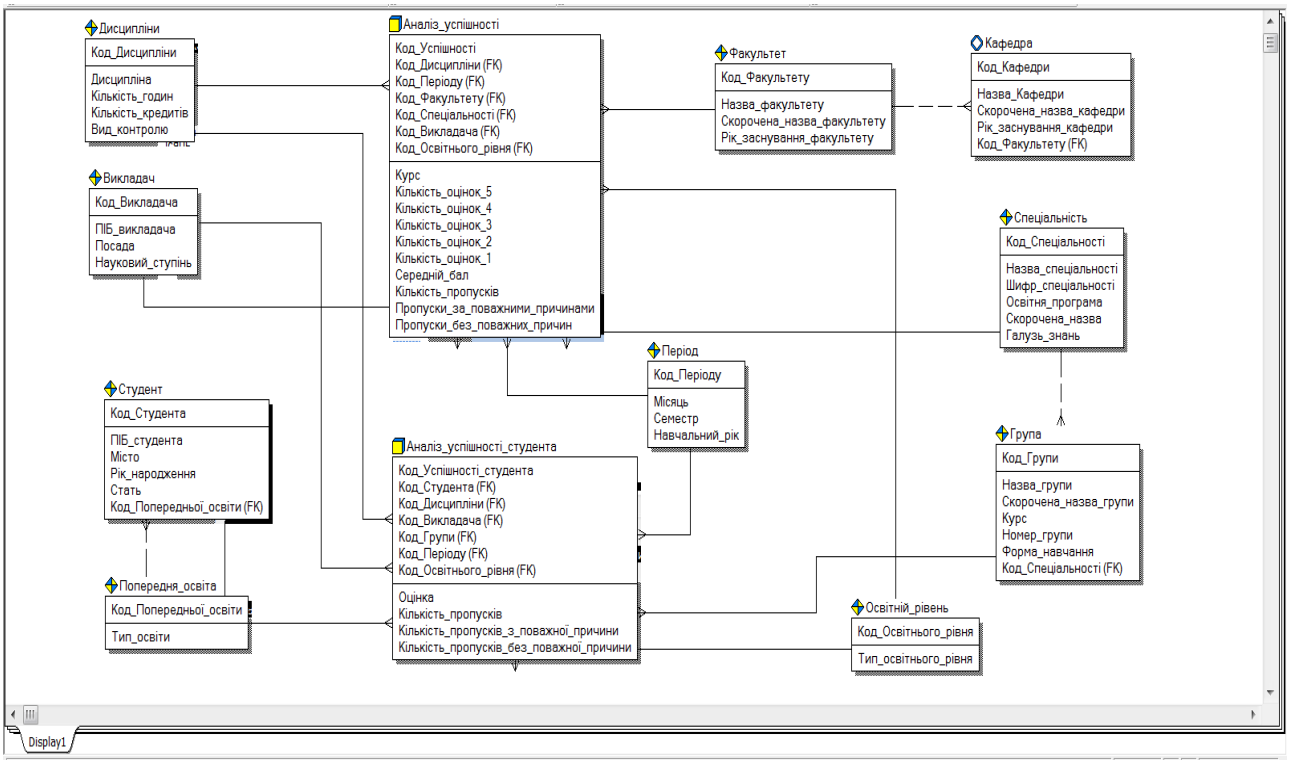
Додаток А.

Схема сховища даних на рівні визначень



Додаток Б.

Схема сховища даних на Dimensional рівні



Додаток В.

Схема сховища даних в MS SQL Server