МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА СИСТЕМНОГО АНАЛІЗУ ТА УПРАВЛІННЯ
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи

магістра
(освітньо-кваліфікаційний рівень)

спеціальності *Системний аналіз*
на тему: *«Удосконалення алгоритму ідентифікації небажаного вторгнення в локальну мережу підприємства в умовах недостатньої інформації із застосуванням технологій OLAP»*

Виконавець: _____
(підпис)

Керівники	**Прізвище, ініціали**	**Оцінка**	**Підпис**
роботи	*Мінєєв О.С.*	82	
розділів:			
Інформаційно-аналітичний розділ	*Мінєєв О.С.*	82	
Спеціальний розділ	*Мінєєв О.С.*	82	
Рецензент	*Корнієнко В.І.*	82	
Нормоконтроль	*Хом'як Т.В.*	82	

Дніпро, 2023

ЗАТВЕРДЖЕНО:

завідувач кафедри

Системного аналізу та управління
(повна назва)

_____ *к. т. н., доц. Т.А. Желдак*
(підпис) (прізвище, ініціали)

„ ___ ” _____ 2023 р.

ЗАВДАННЯ

на кваліфікаційну роботу

магістра
(освітньо-кваліфікаційний рівень)

студенту групи 124м-22-1 Педану Антону Миколайовичу

Тема кваліфікаційної роботи *«Удосконалення алгоритму ідентифікації небажаного вторгнення в локальну мережу підприємства в умовах недостатньої інформації із застосуванням технологій OLAP»*

затверджена наказом ректора НТУ «Дніпровська політехніка» від *“ ”* *2023 р. №*

Розділ	Зміст завдання	Термін виконання
Інформаційно-теоретичний розділ	*На основі аналізу структури початкових даних, типів даних та ступеню їх інформативності запропонувати структуру сховища даних, метрики та виміри, звернення за якими до нього приймаються.* *Визначити методи дослідження, програмне середовище та шляхи розв’язання задачі*	
Спеціальний розділ	*Розробити методологічні підходи до класифікації з’єднань на нормальні та небажані вторгнення, відібрати набір значущих ознак нормального з’єднання.* *Розробити стійкий метод ідентифікації нормальних з’єднань та класифікації небажаних вторгнень*	

Завдання видав _____ _____
(підпис) (прізвище, ініціали)

Завдання прийняв до виконання _____ _____
(підпис) (прізвище, ініціали)

Дата видачі завдання: *01 вересня 2023 р.*

Термін подання кваліфікаційної роботи до ДЕК: *23 грудня 2023 р.*

РЕФЕРАТ

Пояснювальна записка: 94 с., 38 рис., 22 табл., 34 джерел, 2 додатки.

Захист корпоративних комп'ютерних мереж від небажаного вторгнення – актуальна науково-практична проблема, для вирішення якої необхідно вирішення цілого ряду наукових задач, зокрема ідентифікації серед вхідних з'єднань таких, що несуть небезпеку для мережі.

Мета кваліфікаційної роботи: удосконалення алгоритму ідентифікації небажаного вторгнення в локальну мережу підприємства.

Об'єкт дослідження: система захисту інформації локальної комп'ютерної мережі підприємства від небажаного вторгнення.

Предмет дослідження: алгоритми класифікації небезпечних комп'ютерних з'єднань як багатопараметричних слабо визначених об'єктів з використанням технологій OLAP.

В *інформаційно-аналітичному розділі* проведено аналіз структури початкових даних, типів даних та ступеню їх інформативності; запропоновано структуру сховища даних, перелік метрик та вимірів, звернення за якими до нього приймаються; визначити методи дослідження, програмне середовище та шляхи розв'язання задачі.

В *спеціальному розділі* роботи розроблено новий методологічний підхід до класифікації з'єднань на нормальні та небажані вторгнення на основі агломераційної класифікації; розроблено стійкий метод ідентифікації нормальних з'єднань та класифікації небажаних вторгнень; проведено експериментальну перевірку роботи алгоритму.

Практична цінність отриманих в роботі результатів полягає у розробці та реалізації стійкого алгоритму ідентифікації спроб несанкціонованого вторгнення у локальну мережу підприємства з використанням засобів інтелектуального аналізу даних, який ефективно протидіє зовнішнім атакам. Згаданий алгоритм не просто ідентифікує вид відомої мережевої атаки, а й вірно реагує на атаки нових типів.

КЛЮЧОВІ СЛОВА: ЗАХИСТ ІНФОРМАЦІЇ, МЕРЕЖЕВА АТАКА, КЛАСИФІКАЦІЯ, РЕГРЕСІЯ, БАГАТОВИМІРНІ ДАНІ, OLAP, КУБ, СЕГМЕНТ, МОДЕЛЬ, WEKA.

THE ABSTRACT

Explanatory note: 94 pages, 38 figures, 22 tables, 34 sources, 2 appendices.

Protection of corporate computer networks from unwanted intrusion is an urgent scientific and practical problem, the solution of which requires the solution of several scientific problems, in particular, the identification among incoming connections of those that pose a danger to the network.

The purpose of the thesis: improvement of the identification algorithm of unwanted intrusion into the local network of the enterprise.

The object of the study: the system of protecting the information of the local computer network of the enterprise against unwanted intrusion.

The subject of research: algorithms for the classification of dangerous computer connections as multi-parameter weakly defined objects using OLAP technologies.

In the informational and analytical section, an analysis of the structure of initial data, types of data and the degree of their informativeness was carried out; the structure of the data repository, the list of metrics and measurements, applications for which are accepted; determine research methods, software environment and ways to solve the problem.

In a special section of the work, a new methodological approach to the classification of connections into normal and undesirable invasions based on agglomeration classification was developed; a stable method for identifying normal connections and classifying unwanted intrusions has been developed; an experimental check of the algorithm was carried out.

The practical value of the results obtained in the work consists in the development and implementation of a stable algorithm for identifying attempts of unauthorized intrusion into the local network of the enterprise using the means of intelligent data analysis, which effectively counteracts external attacks. The mentioned algorithm not only identifies the type of known network attack, but also correctly reacts to new types of attacks.

KEYWORDS: INFORMATION SECURITY, NETWORK ATTACK, CLASSIFICATION, REGRESSION, MULTIDIMENSIONAL DATA, OLAP, CUBE, SEGMENT, MODEL, WEKA.

ЗМІСТ

Стор.

ВСТУП 7
1 ІНФОРМАЦІЙНО-ТЕОРЕТИЧНИЙ РОЗДІЛ 10
1.1 Захист комп'ютерних мереж від несанкціонованих внутрішніх та зовнішніх вторгнень. 10
1.1.1. Інформаційна безпека 10
1.1.2. Види загроз 12
1.1.3. Протоколи роботи комп'ютерної мережі 15
1.1.4. Відомі шляхи захисту комп'ютерних мереж 16
1.1.5. Сучасний стан проблеми 17
1.2. Опис проблемної області та задачі, що вирішується 19
1.3. Постановка задачі ідентифікації 23
1.3.1. Задача класифікації 23
1.3.2. Математична постановка задачі 25
1.3.3. Задача кластеризації 26
1.4. Аналіз методів вирішення задач класифікації 29
1.4.1. Логістична регресія 29
1.4.2. ROC-аналіз 31
1.4.3. Штучні нейронні мережі. Карти з самоорганізацією 35
1.5. Огляд технології OLAP у сучасних системах підтримки прийняття рішень. 40
1.6. Програмне забезпечення для аналізу даних WEKA 51
1.7. Висновки до розділу. Постановка задач дослідження 53
2 СПЕЦІАЛЬНИЙ РОЗДІЛ 56
2.1 Формування структури сховища даних 56
2.2 Інтелектуальний аналіз вихідних даних 63
2.3 Побудова класифікаційної моделі ідентифікації небажаного вторгнення на основі логістичної регресії 70

2.4 Класифікація видів мережевих вторгнень з використанням мережі Кохонена 77
2.5 Експериментальні дослідження запропонованої методики 82
2.6 Висновки до розділу..... 88
ВИСНОВКИ..... 95
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 100
ДОДАТКИ..... 104
Додаток А Матеріали кваліфікаційної роботи..... 104
Додаток Б Відгук наукового керівника роботи..... 105

ВСТУП

Задача ідентифікації (процесів, систем) або побудова математичної моделі за результатами спостережень посідає одне з головних місць у сучасній теорії управління і прийнятті рішень у різних сферах: техніці, економіці, біології та в ін. Розв'язання задачі ідентифікації є обов'язковим етапом для наступного прийняття рішень або формування керуючого впливу.

Захист корпоративних комп'ютерних мереж від небажаного вторгнення – актуальна науково-практична проблема, що стає ще більш поширеною з розвитком апаратної і програмної бази сучасних комп'ютерів, їх продуктивності та ступеневого збільшення пропускної здатності каналів інформації.

На сьогодні кількість видів як внутрішніх так і зовнішніх загроз для корпоративної комп'ютерної мережі настільки різноманітне, а кількість з'єднань мережі настільки велика, що для забезпечення контролю безпеки мереж необхідно вирішення актуальної наукової задачі вдосконалення алгоритмів ідентифікації серед вхідного потоку з'єднань таких, що несуть небезпеку чи можуть бути небажаними. Саме вирішенню цієї задачі присвячена дана робота.

Об'єктом дослідження в роботі є система захисту інформації локальної комп'ютерної мережі підприємства від небажаного вторгнення.

Предметом дослідження є алгоритми класифікації небезпечних комп'ютерних з'єднань, як багатопараметричних слабо визначених об'єктів, з використанням технологій OLAP.

Мета кваліфікаційної роботи – удосконалення алгоритму ідентифікації небажаного вторгнення в локальну мережу підприємства.

В роботі використовуються наступні *наукові методи*: агломеративна кластеризації – для кластеризації з'єднань, описаних символьними параметрами; багатовимірна логістична регресія – для побудови безпосередньо моделі ідентифікації та нейронні мережі Кохонена – для візуалізації багатовимірного представлення кластерів з'єднань.

Для досягнення загальної мети дослідження в роботі поставлені й вирішені наступні *наукові та практичні задачі*:

- на основі аналізу структури початкових даних, типів даних та ступеню їх інформативності запропоновано структуру сховища даних, метрики та виміри, звернення за якими до нього приймаються;

- визначені методи дослідження, програмне середовище та шляхи розв'язання задачі;

- розроблено методологічні підходи до класифікації з'єднань на нормальні та небажані вторгнення, а також класифікації останніх за видами вторгнень;

- розроблено стійкий алгоритм ідентифікації нормальних з'єднань та класифікації небажаних вторгнень;

- здійснено експериментальну перевірку розробленого алгоритму ідентифікації на тестовій виборці.

На захист виноситься наступне ***наукове положення***:

Відокремлення категорійних параметрів опису складних слабко структурованих об'єктів від числових дозволяє шляхом застосування на першому етапі агломераційної кластеризації, а потім на кожному з кластерів багатовимірної логістичної регресії дозволяє з надійністю не нижче 0,99 ідентифікувати ці об'єкти як елементи одного з наперед відомих класів.

Наукова новизна отриманих у роботі результатів полягає в наступному:

- обґрунтовано застосування методів агломеративної кластеризації та багатовимірної логістичної регресії для класифікації видів з'єднань, що встановлюються локальною комп'ютерною мережею із глобальною мережею;

- систематизовано множину ознак потенційно небезпечних комп'ютерних з'єднань для ідентифікації типу несанкціонованого вторгнення;

- покращена ефективність алгоритму класифікації типів небезпечних комп'ютерних з'єднань для запобігання несанкціонованого вторгнення з використанням технологій OLAP;

Практична цінність результатів полягає у розробці та реалізації стійко-

го алгоритму ідентифікації спроб несанкціонованого вторгнення у локальну мережу підприємства з використанням засобів інтелектуального аналізу даних, який би ефективно протидіяв зовнішнім атакам. Згаданий алгоритм не просто ідентифікує вид відомої мережевої атаки, а й вірно реагує на атаки нових типів.

Результати кваліфікаційної роботи магістра, а саме алгоритм роботи програмного забезпечення, ідентифікує кожне нове з'єднання комп'ютерної мережі та визначає можливий тип вторгнення у реальному часі може бути запропонований системним адміністраторам та фахівцям з безпеки корпоративних мереж.

Економічний ефект від реалізації результатів магістерської роботи досягається за рахунок усунення втрат від недобросовісної конкуренції, зменшення ймовірності викрадення та пошкодження цінної інформації, що є комерційною таємницею та необхідна для життєдіяльності підприємства. Витрати на створення чи вдосконалення системи захисту від небажаних електронних вторгнень не порівняні з можливими наслідками від пошкодження чи розповсюдження цієї інформації.

1 ІНФОРМАЦІЙНО-ТЕОРЕТИЧНИЙ РОЗДІЛ

1.1. Захист комп’ютерних мереж від несанкціонованих внутрішніх та зовнішніх вторгнень.

1.1.1. Інформаційна безпека.

Говорячи про інформаційну безпеку, ми говоримо, в першу чергу, про захист та доступність інформації, якої володіємо, інформації яка становить комерційну таємницю, інформації, яку не хочемо втратити, або щоб вона стала надбанням «чужих» очей. Сфера загроз інформаційної безпеки постійно змінюється, перетворюється, підлаштовується під методи протидії.

Останнім часом стрімко поширюються зловмисні мережі (ботнети), підвищується інтелектуальність мережевих атак, зростає організована кіберзлочинність та шпигунство, з використанням всесвітньої мережі Інтернет, а також з використанням зростаючої мережі мобільних систем. Що в свою чергу, призводять до нових форм додаткових загроз, більш витонченим способам витоку інформації, у тому числі інсайдерських атак (витік інформації через своїх співробітників), і це далеко не повний перелік різноманіття і складності реальних загроз, до яких схильні інформаційні мережі підприємств.

На даний момент мережеві комунікації набувають основоположну роль при веденні бізнесу та отриманні інформації будь-якої сучасної компанії. Для того щоб гарантувати конфіденційність, цілісність і доступність даних і системних ресурсів, необхідно захищати свої інформаційні ресурси, комунікації.

Враховуючи важливість комерційної інформації для компаній і можливі проблеми вразі її часткової втрати або витоку - IT безпека є однією з ключових задач для будь-якого бізнесу.

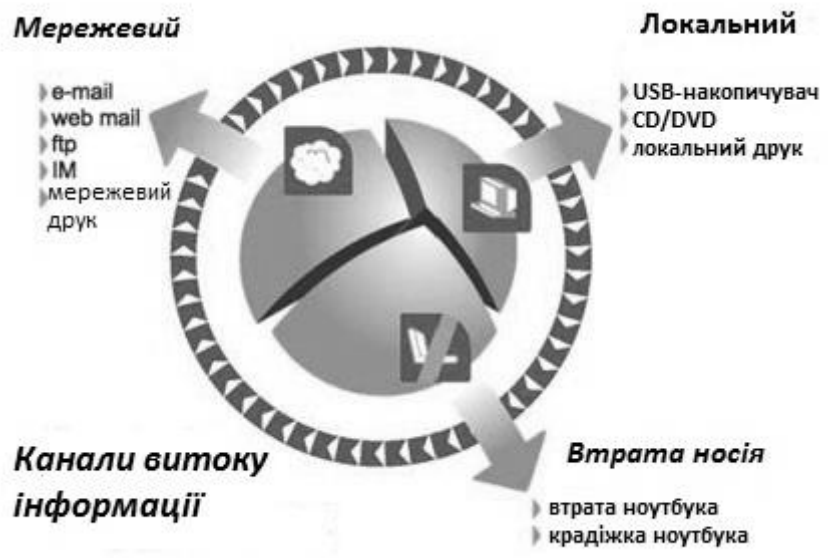


Рис. 1.1 Основні шляхи витоку інформації

Таким чином, захист периметра корпоративної мережі поділяється на дві задачі:

1. Захист зовнішнього периметра, пов'язана з мережевими загрозами і загрозами під час взаємодії з зовнішніми ресурсами.

- Мережеві атаки, спрямовані на недоступність інформаційних ресурсів (наприклад, Web-серверів, сервісів електронної пошти і т.д.) - атаки класу DoS і DDoS;

- компрометація інформаційних ресурсів та ескалація привілеїв - як з боку інсайдерів, так і зовнішніх зловмисників, як з метою використання ваших ресурсів,так і з метою нанесення збитку;

- дії шкідливого програмного коду (віруси, мережеві черв'яки, трояни, програми-шпигуни і т.д.);

- різні мережеві атаки на додатки;

2. Захист внутрішнього периметра, пов'язана з загрозами витоку конфіденційної інформації шляхом викрадення даних інсайдерами або шкідливими програмами.

- Витік конфіденційної інформації і викрадення даних через зовнішні носії;

- дії шкідливого програмного коду (віруси, мережеві черв'яки, трояни, програми-шпигуни і т.д.);

- вилучення робочих станцій і серверів;

1.1.2. Види загроз.

Був час, коли для локальних мереж більшості компаній зовнішні загрози не представляли серйозної небезпеки. Мережі були самодостатніми. Для вторгнення ззовні потрібно було якимось чином звернутися до модему зсередини мережі або фізично підключитися до мережного кабелю. Зараз більшість локальних мереж підключені до Internet, і це радикально змінює ситуацію. Якщо до мережі є доступ із зовнішнього світу, значить, вторгнення звідти можна очікувати в будь-який момент.

Порушення зовнішньої безпеки може проявлятися в різних формах [1]:

- несанкціоноване використання паролів і ключів.

- атаки DoS (Denial of Service).

- заміна IP-адреси.

- комп'ютерні віруси і черв'яки.

- програми виду "троянський кінь".

Несанкціоноване використання паролів і ключів. Пароль - це послідовність букв, цифр та інших символів, за допомогою якої система перевіряє, чи дійсно це той самий користувач, який має цей обліковий запис і право доступу до ресурсів. Ключ - це число, або код, який використовується системою для перевірки цілісності каналу комунікації.

Паролі та ключі - це заходи безпеки, розроблені для запобігання несанкціонованого доступу користувачів до ресурсів мережі. Безпека паролів - найважливіша частина всієї стратегії мережевої безпеки.

Атаки DoS. Атаки DoS (Denial of Service - відмова в обслуговуванні) можуть виконуватися кількома різними способами, але всі мають за мету порушити нормальну роботу комп’ютера, який зазнав атаку.

Атаки DoS не призводять до краху комп'ютера, вони призначені лише для розриву мережевого з'єднання або приведення його у непрацездатний стан. Ці атаки наповнюють мережу безкорисними пакетами або імітують мережеві проблеми, які призводять до розриву з'єднань.

Відомі такі найбільш поширені форми атак DoS:

- на основі протоколу ICMP;
- атаки Smurf;
- Ping of Death;
- атаки SYN.

Протокол ICMP. Це умисне переповнення системи пакетами ICMP, в нормальному режимі призначеними для перевірки правильності повідомлень і виявлення помилок в інформації, яка передається по Internet. Команда ping зазвичай використовується для передачі пакетів ICMP з метою визначення, чи підключений до мережі даний комп'ютер (ідентифікований IP-адресою або ім'ям хоста, яке перетворюється в IP-адресу).

Команда ping посилає повідомлення ICMP Echo Request і очікує від комп'ютера, що перевіряється повідомлення ICMP Echo Reply. Якщо пакети посилати за IP-адресою безперервно, то сервер не може впоратися з їх потоком, його продуктивність зменшується, і в решті решт він відключається через перевищення часу очікування відповіді.

Атаки Smurf. Являє собою потік пакетів ICMP, що зачіпає всі служби провайдера або весь сегмент мережі. Повідомлення ICMP надсилаються за широкомовною адресою і генерують відповіді всіх комп'ютерів підмережі. Коли провайдер атакований зазначеним методом, знижується продуктивність усіх з'єднань, у наслідок чого всі користувачі відключаються.

Ping Of Death. У цьому більш витонченому різновиді атаки DoS використовується обмеження на довжину MTU (Maximum Transmission Unit). Конкретна величина MTU залежить від типу середовища та мережевої архітектури. Якщо довжина переданого пакета перевищує MTU, то він повинен бути розбитий на кілька менших пакетів, які потім будуть зібрані в приймаючому комп'ютері.

Довжина пакета IP, в якому інкапсульована відповідь відгуку ICMP, обмежена 65 535 октетами (октет - це 8 біт даних). Атакуючий, звісно, знає це. Він посилає пакет, у якому довжина поля даних відповіді відгуку ICMP перевищує припустиму кількість октетів. Комп’ютер, який приймає, намагаючись зібрати такий пакет, зазнає крах.

Атаки SYN. Для розриву з'єднання атакуючий може використовувати послідовність узгодження TCP. Атакуючий запускає досить велику кількість запитів сеансу (зазвичай використовуючи для цього підроблену IP-адресу). Комп'ютер, що приймає, ставить ці запити в чергу. Переповнюючи таким чином чергу і підтримуючи її постійно заповненою, атакуючий може повністю блокувати обробку запитів і установку сеансів.

Заміна IP-адреси. Заміна IP-адреси виконується шляхом зміни заголовка пакету, що передається, після чого пакет виглядає так, ніби його передав комп'ютер, що має підставлену адресу. Сама по собі заміна адреси не є формою атаки, це лише метод, який використовується для отримання доступу до мережевих комп'ютерів з метою крадіжки або руйнування даних.

Комп'ютерні віруси і черв'яки. Комп'ютерний вірус - це програма (або фрагмент програми), здатна без відома або згоди користувача розмножуватися і поширюватися на інші програми та комп'ютери шляхом копіювання свого коду в файли, які зберігаються в системі.

Черв’як - це один з різновидів шкідливого вірусу, що розмножується і пошкоджує файли, які зберігаються в комп'ютері. Черв’яки часто поширюються у вкладеннях до електронних повідомлень або в HTML-сторінках, які містять сценарії.

Троянський кінь - це програма, яка впроваджується в комп'ютер під виглядом іншої програми для отримання інформації.

Внутрішні загрози. У практиці корпоративних мереж відомо безліч випадків крадіжки даних, їх зловмисного використання або руйнування не без допомоги службовців самої компанії.

Причини внутрішніх загроз безпеки мережі:

- промислове шпигунство;
- внутрішні інтриги;
- незадоволені службовці (або колишні службовці);
- випадкові порушення.

Більш детально не будемо зупинятись на внутрішніх загрозах.

1.1.3. Протоколи роботи комп’ютерної мережі.

Протокол передачі даних - стандарт, що описує правила взаємодії функціональних блоків при передачі даних.

TCP. TCP являє собою орієнтований на з'єднання протокол наскрізної доставки з гарантією, призначений для використання в багаторівневій ієрархії протоколів, що підтримує різнорідні мережеві додатки. TCP забезпечує надійний обмін даними між парами процесів на вузлах, підключених до різних, але пов'язаних між собою мереж. Протокол виходить з незначного числа передумов про надійність комунікаційних протоколів, розташованих нижче рівня TCP. Протокол TCP передбачає, що він може використовувати простий і потенційно ненадійний сервіс доставки дейтаграм протоколів нижчих рівнів. Взагалі протокол TCP повинен працювати в широкому класі комунікаційних систем, що мають електричні з'єднання з мережами комутації пакетів (packet-switched) або каналів (circuit-switched) [2].

UDP. Протокол передачі дейтаграм, призначений для підтримки режиму обміну дейтаграмами на основі комутації пакетів в середовищі пов'язаних між собою комп'ютерних мереж. Це протокол передбачає використання IP в якості протоколу нижчого рівня.

Протокол UDP забезпечує прикладним програмам процедури для передачі повідомлень іншим додаткам з мінімальним сервісом. Протокол орієнтований на транзакції, не гарантує доставки повідомлень і не запобігає появі дублікатів [2].

ICMP. Протокол забезпечує передачу інформації про виникнення проблем у комунікаційному середовищі. Повідомлення ICMP містить інформацію про помилки при обробці дейтаграм. Повідомлення ICMP передаються з використанням базових заголовків IP. Кожне ICMP-повідомлення інкапсулюється безпосередньо в межах одного IP-пакета, і, таким чином, як і UDP, ICMP є ненадійним протоколом [2]

1.1.4. Відомі шляхи захисту комп’ютерних мереж.

Згідно зі статистикою CIS, кожна друга організація протягом 2009 - 2010 років зафіксувала різні атаки на свої інформаційні ресурси, а 45,6% з них зазнали цілеспрямованого нападу. При цьому Статистика 2009 - 2010 року: 64% респондентів зазнають вірусних атак, 29% - DoS-атак, 14% - мережевих вторгнень.

Для повноцінного захисту системи від атак різного роду необхідно застосовувати цілий комплекс програмного забезпечення: антивірус, мережевий екран, система виявлення атак, і т.д. Саме тому сучасна тенденція на ринку засобів забезпечення безпеки - інтеграція основних функцій антивірусу, мережевого екрану, системи виявлення атак та ін., необхідних персональному користувачеві, в рамках одного продукту. Тим більше, що подібне рішення дозволяє використовувати методи пошуку вірусів в мережевому трафіку або методи виявлення атак на локальних даних.

На сьогоднішній день в області захисту комп'ютерної інформації (інформації, яка зберігається, обробляється і передається за допомогою комп'ютерних систем) склалася наступна ситуація - сучасні комерційні антивірусні програми не в змозі забезпечити належний рівень захисту комп'ютерних систем від шкідливих програм і мережевих атак. Методи і алгоритми, що використовуються в них мають ряд істотних недоліків. Так, найбільш точний на сьогоднішній день метод виявлення шкідливих програм, сигнатурний метод, ґрунтується на сигнатурному аналізі (пошуку унікальних послідовностей символів у файлі, який перевіряється), і добре себе зарекомендував при виявленні вже відомих комп'юте-

рних вірусів (здатний виявляти до 90 - 95% відомих шкідливих програм), абсолютно не придатний для виявлення нових, раніше невідомих шкідливих програм і мережевих атак. На те, щоб виявити нову шкідливу програму, створити для неї сигнатуру і оновити антивірусні бази, потрібен якийсь час, найчастіше досить тривалий (від декількох годин до декількох десятків годин). Весь цей час, комп'ютери в усьому світі залишаються практично не захищеними перед обличчям нової загрози. Як показує практика, саме нові, раніше невідомі комп'ютерні віруси є причиною глобальних інформаційних епідемій і призводять до величезних фінансових та моральних збитків (в 2001 році вірус CodeRed заразив близько 390 000 комп'ютерів за 13 годин, в 2003 році вірус Slammer заразив близько 75 000 комп'ютерів за 10 хвилин, в 2007 році вірус Storm заразив близько 25 мільйонів комп'ютерів) [3].

Для того, щоб забезпечити захист комп'ютерних систем від невідомих шкідливих програм були розроблені різні евристичні (не точні) методи. Такі методи здатні з певною часткою ймовірності винести рішення про шкідливість програми. Однак, існуючі евристичні алгоритми, що застосовуються сьогодні для виявлення невідомих шкідливих програм, характеризуються високим рівнем виникнення помилок першого та другого родів (класифікація легітимних програм як шкідливих і навпаки) і низьким рівнем, менше 60%, виявлення невідомих комп'ютерних вірусів, що ускладнює їх застосування в сучасних антивірусних системах.

1.1.5. Сучасний стан проблеми.

Здійснивши дослідження ситуації, яка склалась в останній час в області здійснення атак [4], маємо констатувати стан сталого зростання актуальності проблеми.:

Кількість переданих за 1 секунду пакетів (PPS) збільшилася в чотири рази в порівнянні з четвертим кварталом 2021 року, таким чином ми спостерігаємо значне зростання всіляких DDoS-атак.

Як заявила компанія Prolexic Technologies, із загального числа атак приблизно 24% складають атаки SYN flood, 22% - ICMP flood і 19% - UDP flood, що означає зміну в тактиці атак.

Поширеність SYN flood і збільшення PPS говорить про те, що відбулася зміна у стратегії - атаки стали менш складними, але надзвичайно небезпечними.

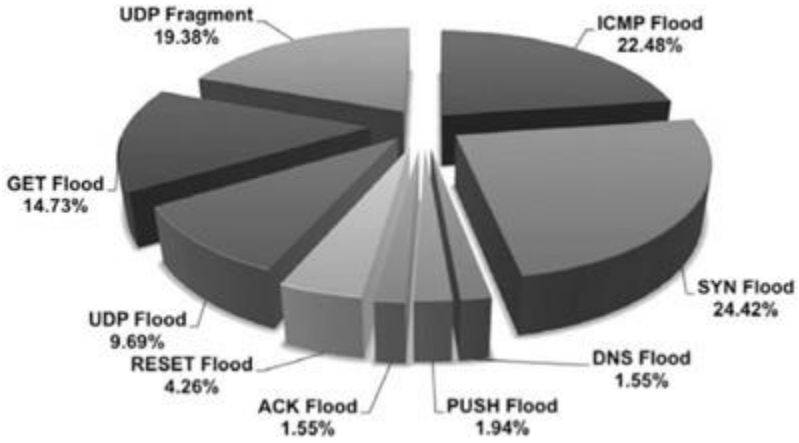


Рис. 1.2 Частки атак різних типів

Британська компанія NCC Group, що спеціалізується в області інформаційного захисту, опублікувала Top 10 країн, з території яких здійснюються хакерські атаки [5].

Географічний рейтинг хак-плацдармів очолюють США і Китай, на частку яких у минулому році припадало 22 і 16% спроб злому відповідно. За сукупністю ці напади щорічно обходяться світовій економіці в 43 млрд. дол. Третє місце з великим відривом займає Росія з показником 3,6%; річний збиток від дій російських хакерів складає близько 4 млрд. дол. П'ятірку лідерів замикають Бразилія (3,5%) та Італія (3,1%).

Примітно, що половину позицій в Top 10 займають країни Західної Європи. З території Італії, Голландії, Франції, Данії і Німеччини було зроблено майже 200 млн. непрошених вторгнень, що обійшлися жертвам в 16 млрд. дол Британія опинилася за межами ведучої «десятки», зайнявши 15-е місце. Проте, за оцінками дослідників, діяльність британських хакерів заподіює значний збиток; в минулому році його розміри склали 2 млрд. дол.

Рейтинг NCC був складений на основі даних про несанкціоновані вторгнення, представлених добровольцями в рамках проекту DSHield (автор - американський інститут SANS). Підсумкова статистика враховує всі спроби злому, як успішні, так і провальні.

За даними Prolexic Technologies, в січні-березні кількість DDoS-атак в фінансовому секторі збільшилося майже в 3 рази. Захисні рішення компанії ідентифікували і нейтралізували 1,1 трлн. шкідливих пакетів - на 3000% більше, ніж у попередньому кварталі.

Тим не менш, сукупний підсумок залишився практично незмінним, хоча на 25% перевищує загальний показник річної давності. Середня тривалість DDoS-атак продовжує знижуватися: рік тому вона становила 65 годин, в IV кварталі - 34 години, в новому році 28,5 годин. У той же час потужність цих ударів зростає: згідно зі статистикою Prolexic, за квартал середньостатистична швидкість DDoS - трафіку збільшилася з 5,2 до 6,1 Гбіт / с (півроку тому цей показник становив лише 1,5 Гбіт / с). Основним джерелом DDoS-атак, як і раніше є Китай, хоча американські та російські плацдарми збільшили свою активність.

При проведенні DDoS-атак зловмисники, як і раніше вважають за краще використовувати протоколи 3 і 4 рівня (мережеві і транспортні), проте кількість атак на додатки продовжує збільшуватися. За квартал число DDoS прикладного рівня зросло на 6%, за рік - на 25%. Техніка UDP Flood поступово сходить зі сцени, а GET Flood підвищує свій рейтинг. Основна маса DDoS-атак в даний час проводиться за типом SYN Flood.

1.2. Опис проблемної області та задачі, що вирішується.

Програмне забезпечення для виявлення мережевих вторгнень захищає комп'ютерну мережу від несанкціонованого доступу, включаючи можливих інсайдерів (членів даної організації). Задача навчання датчика вторгнення полягає

в побудові моделі прогнозування (наприклад, класифікатор), що здатна розрізняти «погане» з'єднання, так зване вторгнення або напад, і «хороше» - нормальне з'єднання.

У 1998 році була підготовлена DARPA – програма для оцінки виявлення вторгнень. Вона знаходилась під управлінням компанією MIT Lincoln Labs. Мета полягала в огляді та оцінці досліджень в області виявлення вторгнень. Був представлений стандартний набір даних, що підлягають аудиту. Він включав в себе широкий спектр вторгнень, змодельованих в середовищі військової мережі. В1999 році на конкурсі виявлення вторгнень KDD використовувалася версія цього набору даних.

Lincoln Labs створили умови для отримання сирих даних TCP протягом дев'яти тижнів для локальних мереж (LAN), що імітують типові локальні мережі повітряних військ в США. Lincoln Labs діяли так, ніби мережа насправді належала військам, однак вони засипали її численними атаками.

Підготовлені тренувальні вихідні дані важили близько чотирьох гігабайт. Вони були представлені у вигляді стислих двійкових даних TCP, які отримані наприкінці семи тижнів мережевого трафіку. Дані були перероблені приблизно в п'ять мільйонів записів з’єднань. Подібним чином, тестові дані, зібрані за два тижні, дали близько двох мільйонів записів з’єднань [6].

З'єднання представляє собою послідовність пакетів TCP, передача яких починається і закінчується в якийсь чітко визначений час. В цей час дані переходять від IP-адреси джерела до цільової IP-адреси за деяким чітко визначеним протоколом. Кожне з'єднання позначається як або нормальне, або як атака тільки з одним конкретним типом атаки. Запис кожного з'єднання складається близько зі 100 байт.

Атаки можна розділити на чотири основні категорії:

- DOS: відмова в обслуговуванні, наприклад, SYN - флуд;

- R2L: несанкціонований доступ з віддаленого комп'ютера, наприклад, вгадування пароля;

- U2R: несанкціонований доступ до локального пріоритетного суперкори-

стувача (корінь). Наприклад, різні атаки типу «буфер переповнений»;

- зондування: спостереження і розвідка, наприклад, сканування портів.

Важливо зазначити, що тестові дані розподілені не з тією ж імовірністю, що й тренувальні дані, і включають в себе певні спеціальні типи атак, які не містяться в тренувальних даних. Це робить задачу більш реалістичною. Деякі експерти з вторгнень вважають, що більшість нових атак є різновидами вже відомих атак, і знання "почерку" відомих атак може бути достатньо, щоб вловити їх нові різновиди. Набори даних містять загалом 24 типи тренувальних атак, а також 14 додаткових видів, що містяться лише в тестових даних.

Ознаки-функції для розпізнавання атак

Stolfo і співавтори визначили ряд ознак (функцій) більш високого рівня, які допомагають відрізнити нормальне з'єднання від атаки. Існує кілька типів отриманих функцій.

Функції типу «один хост» аналізують тільки ті зв'язки протягом останніх двох секунд, які мають один хост призначення в якості поточного з'єднання, і обчислюють статистику, пов'язану з протоколом поведінки, службами і т.д.

Аналогічні функції «однієї служби» розглядають тільки ті зв'язки протягом останніх двох секунд, які мають таку само службу, що і поточне з'єднання.

Функції типу «один хост» і «одна служба» разом називаються часовими трафік - функціями записів з’єднань.

Деякі зондовані атаки сканують хости (або порти), використовуючи набагато більший проміжок часу, ніж дві секунди, наприклад, раз на хвилину. Тому, записи з'єднань були також сортовані за хостами призначення, і функції були побудовані з використанням вікна на 100 з’єднань, що підключені до одного хосту, а не часового вікна. Це надає безліч так званих хост - функцій трафіку.

На відміну від більшості DOS - атак і атак зондування, з'являється також безліч непослідовних моделей, які часто зустрічаються в записах атак R2L і U2R. Це відбувається тому, що атаки DOS і зондування включають в себе декілька з'єднань на деяких хостах (або на одному) за дуже короткий період часу, а атаки R2L і U2R криються в порціях даних і зазвичай атакують тільки за одне

з'єднання.

Таким чином, розробка корисних алгоритмів для отримання неструктурованих порцій даних автоматично стає відкритим питанням для досліджень. Stolfo і співавтори використовують знання доменів, щоб додавати нові функції, спрямовані на пошук підозрілої поведінки (змісту) в порціях даних, наприклад, число невдалих спроб авторизації. Ці функції називаються контент-функціями, або функціями змісту.

Повний список набору функцій, що визначають зв'язок між записами, наведено в трьох таблицях нижче.

Таблиця 1.1

Основні характеристики окремих з’єднань TCP

Назва функції	Опис функції	Тип
Duration	тривалість з'єднання (число секунд)	безперервна
protocol_type	тип протоколу, наприклад, TCP, UDP та ін.	дискретна
service	мережева служба призначення, наприклад, HTTP, Telnet та ін.	дискретна
src_bytes	кількість даних у байтах, що передані від джерела до адресата	безперервна
dst_bytes	кількість даних у байтах, що передані від адресата до джерела	безперервна
flag	стан з’єднання - нормальне або помилка	дискретна
land	1, якщо з'єднання з/на той само хост/порт, інакше 0	дискретна
wrong_fragment	число «невірних» фрагментів	безперервна
urgent	ряд термінових/екстрених пакетів	безперервна

Таблиця 1.2

Контент-функції всередині з’єднань, запропонованих доменом

Назва функції	Опис функції	Тип
hot	число «гарячих» (екстремальних, небезпечних) показників	безперервна
num_failed_logins	число невдалих спроб авторизації	безперервна
logged_in	1 у випадку успішної авторизації, інакше 0	дискретна
num_compromised	число «скомпрометованих» умов	безперервна
root_shell	1, якщо кореневий запис оболонки існує, 0 в ін. випадку	дискретна
su_attempted	1, при виклику команди «su root»; 0 в іншому випадку	дискретна
num_root	число «кореневих» доступів	безперервна
num_file_creations	число операцій створення файлів	безперервна
num_shells	число викликів оболонки	безперервна
num_access_files	кількість операцій доступу до файлів контролю	безперервна
num_outbound_cmds	кількість вихідних команд у сесії FTP	безперервна
is_hot_login	1, якщо логін належить «гарячому» списку; 0 в ін. випадку	дискретна
is_guest_login	1, якщо логін є гостьовим; 0 в іншому випадку	дискретна

Таблиця 1.3

Трафік - функції, які використовують двох секундне вікно

Назва функції	Опис функції	Тип
count	число підключень до того ж хосту, що і поточне з'єднання, протягом останніх двох секунд	безперервна
Примітка: наступні функції відносяться до підключень до одного хосту		
serror_rate	% підключень, які мають SYN помилки	безперервна
rerror_rate	% підключень, які мають REJ помилки	безперервна
same_srv_rate	% підключень до однієї служби	безперервна
diff_srv_rate	% підключень до різних служб	безперервна
srv_count	число підключень до тієї ж служби, що і поточне з'єднання, протягом останніх двох секунд	безперервна
Примітка: наступні функції відносяться до підключень до однієї служби		
srv_serror_rate	% підключень, які мають SYN помилки	безперервна
srv_rerror_rate	% підключень, які мають REJ помилки	безперервна
srv_diff_host_rate	% з'єднань з різними хостами	безперервна

1.3 Постановка задачі ідентифікації.

1.3.1. Задача класифікації.

Класифікація є найбільш простою задачею і водночас задачею Data Mining, яка найбільш часто розв'язується. Через поширеність задач класифікації необхідно чітке розуміння суті цього поняття.

Задача класифікації - формалізована задача, в якій є множина об'єктів (ситуацій), які розділені деяким чином на класи. Визначена кінцева множина об'єктів, для яких відомо, до яких класів вони відносяться. Ця множина називається вибіркою. Класова приналежність інших об'єктів не відома. Потрібно побудувати алгоритм, що здатен класифікувати довільний об'єкт з вихідної множини.

Класифікувати об'єкт - значить, вказати номер (або найменування) класу, до якого відноситься даний об'єкт.

Класифікація вимагає дотримання наступних правил:

- у кожному акті поділу необхідно застосовувати тільки одну підставу;

- поділ має бути пропорційним, тобто загальний обсяг видових понять повинен дорівнювати обсягу діленого родового поняття;

- члени поділу повинні взаємно виключати один одного, їх обсяги не по-

винні перехрещуватися;

- поділ має бути послідовним.

Розрізняють:

- допоміжну (штучну) класифікацію, яка проводиться за зовнішньою ознакою і служить для надання множині предметів (процесів, явищ) потрібного порядку;

- природну класифікацію, яка проводиться за істотними ознаками, що характеризують внутрішню спільність предметів і явищ. Вона є результатом і важливим засобом наукового дослідження, тому що передбачає і закріплює результати вивчення закономірностей об'єктів, які класифікуються.

В залежності від обраних ознак, їх поєднання і процедури поділу понять класифікація може бути:

- простою - поділ родового поняття тільки за ознакою і лише один раз до розкриття всіх видів. Прикладом такої класифікації є дихотомія, при якій членами поділу бувають тільки два поняття, кожне з яких суперечить іншому (тобто дотримується принцип: "А і не А");

- складною - застосовується для поділу одного поняття з різних підстав та синтезу таких простих поділів в єдине ціле. Прикладом такої класифікації є періодична система хімічних елементів.

Під класифікацією будемо розуміти віднесення об'єктів (спостережень, подій) до одного з наперед відомих класів.

Класифікація відноситься до стратегії навчання з вчителем (supervised learning), яке також називають контрольованим або керованим навчанням.

Задачею класифікації часто називають передбачення категоріальної залежної змінної (тобто залежної змінної, яка є категорією) на основі вибірки безперервних і / або категоріальних змінних.

Класифікація може бути одномірною (за однією ознакою) і багатомірною (за двома і більше ознаками).

Багатомірна класифікація була розроблена біологами при вирішенні проблем дискримінації для класифікування організмів. Однією з перших робіт,

присвячених цьому напрямку, вважають роботу Р. Фішера (1930 р.), в якій організми поділялися на підвиди залежно від результатів вимірювань їх фізичних параметрів. Біологія була і залишається найбільш затребуваним і зручним середовищем для розробки багатомірних методів класифікації.

Мета процесу класифікації полягає в тому, щоб побудувати модель, яка використовує прогнозуючі атрибути в якості вхідних параметрів і отримує значення залежного атрибуту. Процес класифікації полягає в розбитті множини об'єктів на класи за певним критерієм.

Класифікатором називається деяка сутність, яка визначає, якому з визначених класів належить об'єкт за вектором ознак.

Для проведення класифікації за допомогою математичних методів необхідно мати формальний опис об'єкта, яким можна оперувати, використовуючи математичний апарат класифікації. Таким описом у нашому випадку виступає база даних.

Кожен об'єкт (запис бази даних) несе інформацію про деяку властивість об'єкта.

Набір вихідних даних (або вибірку даних) розбивають на дві множини: навчальну і тестову.

Навчальна множина (training set) - множина, яка включає дані, що використовуються для навчання (конструювання) моделі. Така множина містить вхідні та вихідні (цільові) значення прикладів. Вихідні значення призначені для навчання моделі.

Тестова (test set) множина також містить вхідні і вихідні значення прикладів. Тут вихідні значення використовуються для перевірки працездатності моделі.

1.3.2. Математична постановка задачі.

Нехай X- множина описів об'єктів, Y – множина номерів (або найменувань) класів. Існує невідома цільова залежність – відображення $y^*: X \to Y$, значення якої відомі лише на об'єктах кінцевої навчальної вибірки

$X^m = \{(x_1, y_1), \ldots, (x_m, y_m)\}$. Потрібно побудувати алгоритм $a: X \to Y$, здатний класифікувати довільний об'єкт $x \epsilon X$.

Для класифікації використовуються різні методи. Основні з них:

- класифікація за допомогою дерев рішень;
- байєсівська (наївна) класифікація;
- класифікація за допомогою штучних нейронних мереж;
- класифікація методом опорних векторів;
- статистичні методи, зокрема, лінійна регресія;
- класифікація за допомогою методу найближчого сусіда;
- класифікація CBR-методом;
- класифікація за допомогою генетичних алгоритмів.

Деякі алгоритми для вирішення задач класифікації комбінують навчання з учителем з навчанням без учителя, наприклад, одна з версій нейронних мереж Кохонена - мережі векторного квантування, які навчаються з учителем.

1.3.3. Задача кластеризації.

Задача кластеризації схожа з задачею класифікації, є її логічним продовженням, але її відмінність в тому, що класи досліджуваного набору даних заздалегідь не визначені.

Синонімами терміна "кластеризація" є "автоматична класифікація", "навчання без вчителя" і "таксономія".

Кластеризація призначена для розбиття сукупності об'єктів на однорідні групи (кластери або класи). Якщо дані вибірки представити як точки в просторі ознак, то задача кластеризації зводиться до визначення "згущень точок".

Кластеризація є описовою процедурою, вона не робить жодних статистичних висновків, але дає можливість провести розвідувальний аналіз і вивчити "структуру даних".

Кластер можна охарактеризувати як групу об'єктів, що мають спільні властивості.

Питання, яке задається аналітиками при вирішенні багатьох задач, поля-

гає в тому, як організувати дані в наглядні структури, тобто розгорнути таксономії. У таблиці 1.4 наведено порівняння деяких параметрів задач класифікації та кластеризації.

Таблиця 1.4

Порівняння задач класифікації та кластеризації

Характеристика	Класифікація	Кластеризація
Контрольованість навчання	Контрольоване навчання	Неконтрольоване навчання
Стратегія	Навчання з учителем	Навчання без вчителя
Наявність мітки класу	Навчальна множина супроводжується міткою, що вказує клас, до якого належить спостереження	Мітки класу навчальної множини невідомі
Підстава для класифікації	Нові дані класифікуються на підставі навчальної множини	Дана множина даних з метою встановлення існування класів або кластерів даних

На рис. 1.3 схематично представлені задачі класифікації і кластеризації.

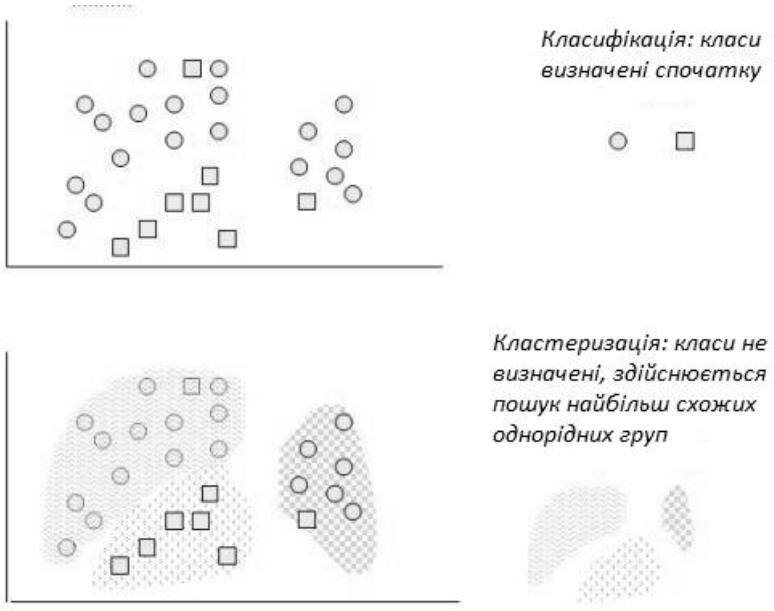


Рис.1.3 Порівняння задач класифікації та кластеризації

Кластери можуть не перетинатися, так звані ексклюзивні (non-overlapping, exclusive), і перетинатися (overlapping). Схематичне зображення кластерів, які перетинаються і які не перетинаються наведено на рис. 1.4.

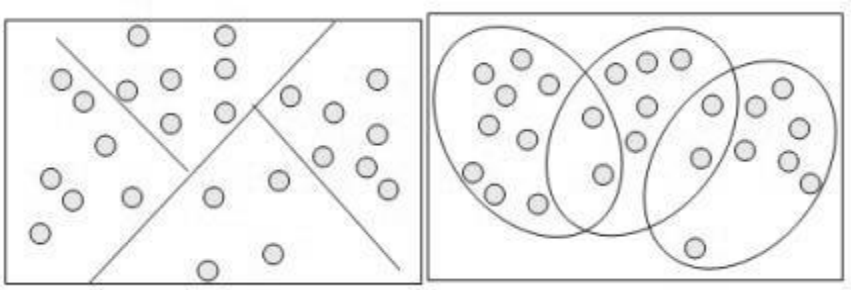


Рис. 1.4. Кластери, які перетинаються і які не перетинаються

Оцінка якості кластеризації може бути проведена на основі наступних процедур:

- ручна перевірка;

- встановлення контрольних точок і перевірка на отриманих кластерах;

- визначення стабільності кластеризації шляхом додавання в модель нових змінних;

- створення і порівняння кластерів з використанням різних методів. Різні методи кластеризації можуть створювати різні кластери, і це є нормальним явищем. Проте створення схожих кластерів різними методами вказує на правильність кластеризації.

Процес кластеризації залежить від обраного методу і майже завжди є ітеративним. Він може включати безліч експериментів з вибору різноманітних параметрів, наприклад, міри відстані, типу стандартизації змінних, кількості кластерів і т.д. Кінцевою метою кластеризації є отримання змістовних відомостей про структуру досліджуваних даних. Отримані результати потребують подальшої інтерпретації, дослідження та вивчення властивостей і характеристик об'єктів для можливості точного опису сформованих кластерів.

Найбільш поширені методи розв'язання задачі кластеризації:

- метод k-середніх (працює тільки з числовими атрибутами),

- ієрархічний кластерний аналіз (працює також з символьними атрибутами),

- метод SOM.

Складністю кластеризації є необхідність її оцінки.

1.4 Аналіз методів вирішення задач класифікації та кластеризації

1.4.1. Логістична регресія.

Логістична регресія - корисний класичний інструмент для рішення задачі регресії та класифікації.

Логістична регресія - це різновид множинної регресії, загальне призначення якої полягає в аналізі зв'язку між кількома незалежними змінними (званими також регресорами або предикторами) і залежною змінною. Бінарна логістична регресія, як випливає з назви, застосовується у разі, коли залежна змінна є бінарною (тобто може приймати тільки два значення). Іншими словами, за допомогою логістичної регресії можна оцінювати ймовірність того, що подія наступить для конкретного випробуваного.

Для передбачення безперервної змінної зі значеннями на відрізку [0,1] при будь-яких значеннях незалежних змінних застосуємо наступне регресійне рівняння (логіт-перетворення):

$$P = \frac{1}{1+e^{-y}}, \tag{1.1}$$

де P - ймовірність того, що станеться подія, яка цікавить;

y - стандартне рівняння регресії.

Залежність, що зв'язує ймовірність події та величину y, показана на наступному графіку (рис. 1.5)

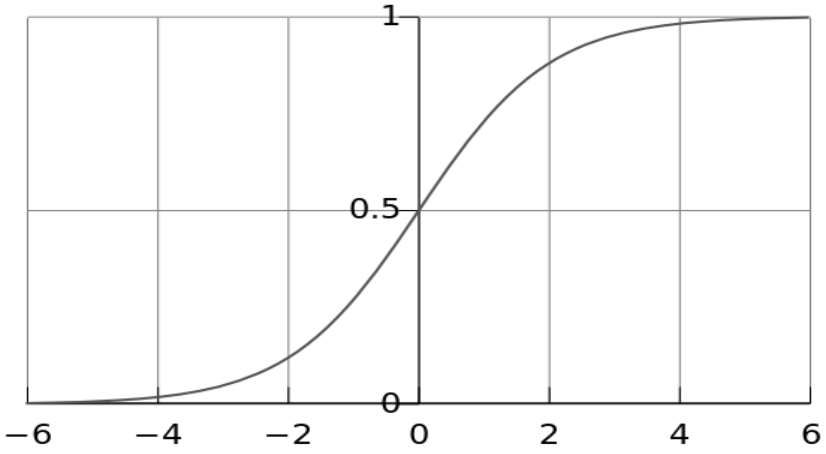


Рис. 1.5 - Логістична крива

Існує кілька способів знаходження коефіцієнтів логістичної регресії. На практиці часто використовують метод максимальної правдоподібності. Основу метода складає функція правдоподібності (likehood function), що виражає щільність імовірності (ймовірність) спільної появи результатів вибірки $Y_1, Y_2, \ldots, Y_k$: $L(Y_1, Y_2, \ldots, Y_k; \theta) = p(Y_1; \theta) \ldots p(Y_k; \theta)$. Згідно методу максимальної правдоподібності в якості оцінки невідомого параметра приймається таке значення $\Theta = \Theta(Y_1, \ldots, Y_k)$, яке максимізує функцію *L*.

Знаходження оцінки спрощується, якщо максимізувати не саму функцію *L*, а натуральний логарифм *ln(L)*, оскільки максимум обох функцій досягається при одному і тому ж значенні θ:

$$L^*(Y;\theta) = ln\left(L(Y;\theta)\right) \to max \tag{1.2}$$

У разі бінарної незалежної змінної, яку ми маємо в логістичній регресії, викладки можна продовжити наступним чином. Позначимо через P_i ймовірність появи одиниці: $P_i = Prob(Y_i = 1)$. Ця ймовірність буде залежати від $X_i W$, де X_i - рядок матриці регресорів, *W* - вектор коефіцієнтів регресії:

$$P_i = F(X_i W),\ F(z) = \frac{1}{1+e^{-z}} \tag{1.3}$$

Логарифмічна функція правдоподібності дорівнює:

$$L^* = \sum_{i \in I1} \ln P_i(W) + \sum_{i \in I0} \ln\left(1 - P_i(W)\right) == \sum_{i=1}^{k} \left[Y_i \ln P_i(W) + (1 - Y_i) \ln\left(1 - P_i(W)\right) \right], \tag{1.4}$$

де *I0, I1* - множини спостережень, для яких $Y_i = 0$ і $Y_i = 1$ відповідно.

Можна показати, що градієнт *g* і гессіан *H* функції правдоподібності дорівнюють відповідно

$$g = -\sum_i (Y_i - P_i) X_i\ , \tag{1.5}$$

$$H = -\sum_i P_i (1 - P_i) X_i^T X_i \le 0\,. \tag{1.6}$$

Гессіан усюди негативно визначений, тому логарифмічна функція правдоподібності всюди увігнута. Для пошуку максимуму можна використовувати метод Ньютона, який тут буде завжди сходитися (виконана умова збіжності ме-

тоду):

$$W_{t+1} = W_t - \left(H\left(W_t\right)\right)^{-1} g_t\left(W_t\right) = W_t - W_t . \quad (1.7)$$

Насправді, логістичну регресію можна представити у вигляді одношарової нейронної мережі з сигмоїдальною функцією активації, ваги якої є коефіцієнти логістичної регресії, а вага поляризації - константа регресійного рівняння (рис. 1.6).

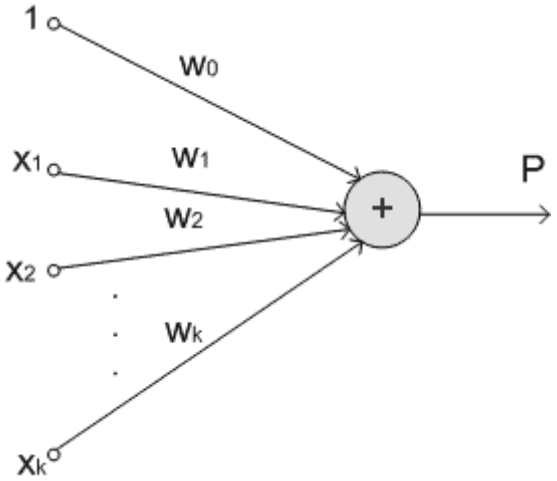


Рис.1.6 - Представлення логістичної регресії у вигляді нейронної мережі

1.5.2. ROC-аналіз.

ROC-крива (Receiver Operator Characteristic) - крива, яка найбільш часто використовується для представлення результатів бінарної класифікації в машинному навчанні. Назва прийшла з систем обробки сигналів. Оскільки класів два, один з них називається класом з позитивними результатами, другий - з негативними результатами. ROC-крива показує залежність кількості вірно класифікованих позитивних прикладів від кількості невірно класифікованих негативних прикладів. У термінології ROC-аналізу перші називаються істинно позитивним, другі - помилково негативною множиною. При цьому передбачається, що у класифікатора є деякий параметр, варіюючи який, ми будемо отримувати те чи інше розбиття на два класи. Цей параметр часто називають порогом, або точкою відсікання (cut-off value). В залежності від нього будуть виходити різні величини помилок I і II роду.

В логістичній регресії поріг відсікання змінюється від 0 до 1 - це і є роз-

рахункове значення рівняння регресії. Будемо називати його рейтингом.

Для розуміння суті помилок I і II роду розглянемо таблицю зв’язаності (confusion matrix), яка будується на основі результатів класифікації моделлю і фактичною (об'єктивною) приналежністю прикладів до класів.

Таблиця 1.5

Таблиця зв’язаності результатів моделювання

	Фактично	
Модель	позитивно	негативно
позитивно	TP	FP
негативно	FN	TN

TP (True Positives) - вірно класифіковані позитивні приклади (так звані істинно позитивні випадки);

TN (True Negatives) - вірно класифіковані негативні приклади (істинно негативні випадки);

FN (False Negatives) - позитивні приклади, класифіковані як негативні (помилка I роду). Це так званий "помилковий пропуск" - коли подія, яка нас цікавить помилково не виявляється (помилково негативні приклади);

FP (False Positives) - негативні приклади, класифіковані як позитивні (помилка II роду). Це помилкове виявлення, тому що при відсутності події помилково виноситься рішення про її присутність (помилково позитивні випадки).

При аналізі частіше оперують не абсолютними показниками, а відносними - частками (rates), вираженими у відсотках:

Частка істинно позитивних прикладів (True Positives Rate):

$$TPR = TP / (TP + FN) \cdot 100\%$$

Частка помилково позитивних прикладів (False Positives Rate):

$$FPR = FP / (TN + FP) \cdot 100\%$$

Введемо ще два визначення: чутливість та специфічність моделі. Ними визначається об'єктивна цінність будь-якого бінарного класифікатора.

Чутливість (Sensitivity) - це і є частка істинно позитивних випадків:

$$Se = TPR = TP / (TP + FN) \cdot 100\%$$

Специфічність (Specificity) - частка істинно негативних випадків, які були правильно ідентифіковані моделлю:

$Sp = TN / (TN + FP) \cdot 100\%$

Зауважимо, що $FPR = 100\text{-}Sp$.

ROC-крива виходить таким чином:

1. Для кожного значення порога відсікання, яке змінюється від 0 до 1 з кроком *dx* (наприклад, 0.01) розраховуються значення чутливості *Se* і специфічності *Sp*. В якості альтернативи порогом може бути кожне наступне значення прикладу у вибірці.

2. Будується графік залежності: по осі *Y* відкладається чутливість *Se*, по осі *X* - *100% - Sp* (сто відсотків мінус специфічність), або, що те ж саме, *FPR* - частка помилково позитивних випадків.

В результаті вимальовується деяка крива (рис. 1.7).

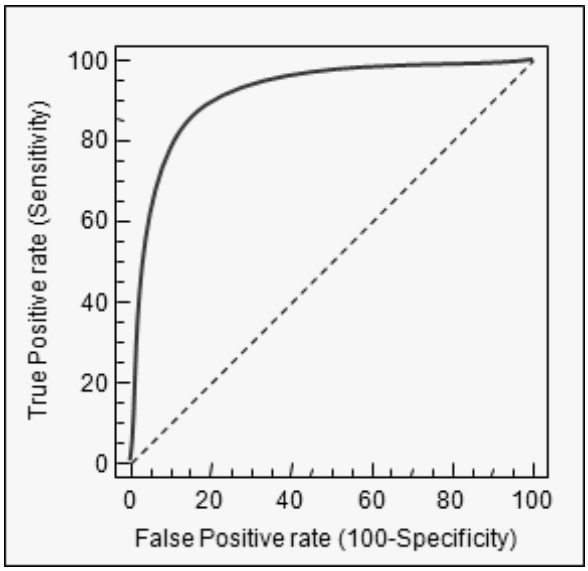


Рис. 1.7 – ROC - крива

Графік часто доповнюють прямою $y = x$.

Для ідеального класифікатора графік ROC-кривої проходить через верхній лівий кут, де частка істинно позитивних випадків становить 100% або 1.0 (ідеальна чутливість), а частка помилково позитивних прикладів дорівнює нулю. Тому чим ближче крива до верхнього лівого кута, тим вище прогнозуюча здатність моделі. Навпаки, чим менше вигин кривої і чим ближче вона розташована до діагональної прямої, тим менш ефективна модель. Діагональна лінія

відповідає "марному" класифікатору, тобто повній нерозрізненості двох класів.

Візуальне порівняння кривих ROC не завжди дозволяє виявити найбільш ефективну модель. Своєрідним методом порівняння ROC-кривих є оцінка площі під кривими. Теоретично вона змінюється від 0 до 1.0, але, оскільки модель завжди характеризується кривою, розташованою вище позитивної діагоналі, то зазвичай говорять про зміни від 0.5 («марний» класифікатор) до 1.0 ("ідеальна" модель). Ця оцінка може бути отримана безпосередньо обчисленням площі під багатогранником, обмеженим праворуч і знизу осями координат і зліва вгорі - експериментально отриманими точками. Чисельний показник площі під кривою називається AUC (Area Under Curve).

З великими допущеннями можна вважати, що чим більше показник AUC, тим кращою прогностичною силою володіє модель. Однак варто знати, що:

- показник AUC призначений скоріше для порівняльного аналізу декількох моделей;

- AUC не містить ніякої інформації про чутливість і специфічність моделі.

У літературі іноді наводиться така експертна шкала для значень AUC, за якою можна судити про якість моделі:

Таблиця 1.6

Шкала значень AUC

Інтервал AUC	Якість моделі
0.9-1.0	Відмінне
0.8-0.9	Дуже гарне
0.7-0.8	гарне
0.6-0.7	середнє
0.5-0.6	незадовільне

Неможливо одночасно підвищити і чутливість, і специфічність моделі, компроміс знаходиться за допомогою порога відсікання. Поріг відсікання потрібен для того, щоб застосовувати модель на практиці: відносити нові приклади до одного з двох класів. Для визначення оптимального порогу потрібно задати критерій його визначення, тому що в різних задачах присутня своя оптимальна

стратегія. Критеріями вибору порога відсікання можуть виступати:

- вимога мінімальної величини чутливості (специфічності) моделі;
- вимога максимальної сумарної чутливості і специфічності моделі, тобто

$$Cut\_off_0 = \max_k (Se_k + Sp_k) \tag{1.8}$$

- вимога балансу між чутливістю і специфічністю, тобто коли $Se \approx Sp$:

$$Cut\_off_0 = \min_k \left| Se_k - Sp_k \right|. \tag{1.9}$$

1.4.3. Штучні нейронні мережі. Карти з самоорганізацією.

Однією з технологій, що використовуються для вирішення задач обробки і аналізу даних, розпізнавання зображень, класифікації та прогнозування є штучні нейронні мережі.

Нейронна мережа являє собою сукупність нейроподібних елементів, певним чином пов'язаних один з одним і зовнішнім середовищем за допомогою зв'язків, що визначаються ваговими коефіцієнтами. В процесі функціонування мережі здійснюється перетворення вхідного вектора у вихідний, деяка переробка інформації.

Конкретний вид перетворення даних, яке виконується мережею обумовлюється не тільки характеристиками нейроподібних елементів, але й особливостями її архітектури, а саме топологією міжнейронних зв'язків, вибором певних підмножин нейроподібних елементів для введення і виведення інформації, способами навчання мережі, наявністю або відсутністю конкуренції між нейронами, напрямком і способами управління і синхронізації передачі інформації між нейронами.

Карти з самоорганізацією - це один з різновидів нейромережевих алгоритмів. Основною відмінністю даної технології від нейромереж, які навчаються за алгоритмом зворотного поширення, є те, що при навчанні використовується метод навчання без вчителя, тобто результат навчання залежить тільки від структури вхідних даних. При такому навчанні навчальна множина складається лише із значень вхідних змінних, в процесі навчання немає порівняння виходів

нейронів з еталонними значеннями. Можна сказати, що така мережа вчиться розуміти структуру даних.

Ідея мережі Кохонена належить фінському вченому Тойво Кохонену (1982 рік). Основний принцип роботи мереж - введення в правило навчання нейрона інформації щодо його розташування.

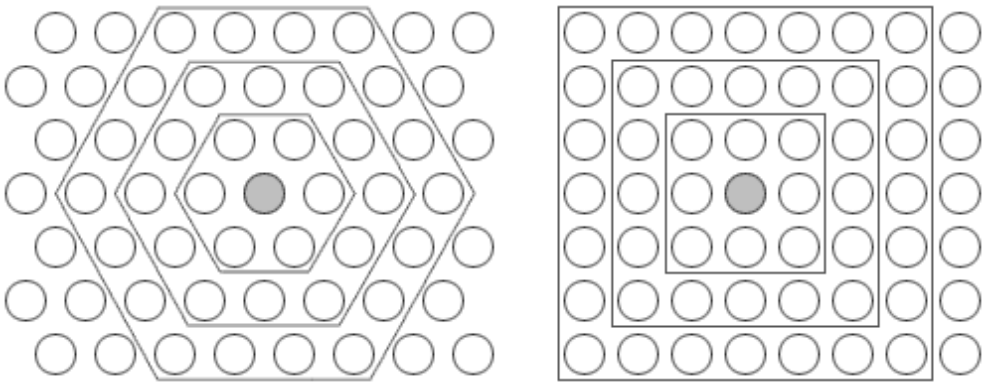
Найбільш поширене застосування мереж Кохонена - рішення задачі класифікації без вчителя, тобто кластеризації. При такій постановці задачі дано набір об'єктів, кожному з яких відповідає рядок таблиці (вектор значень ознак). Потрібно розбити вихідну множину на класи, тобто для кожного об'єкта знайти клас, до якого він належить.

Існують різні постановки задач, що вирішуються застосуванням мереж Кохонена, як наприклад в нашому випадку: розбиття вихідної множини на наперед задану кількість класів.

Технологія карт ознак, що самоорганізуються являє собою набір аналітичних процедур і алгоритмів, що дозволяють перетворити традиційний опис багатьох об'єктів, заданих в багатовимірному ($n > 3$) просторі ознак плоскої бази даних, в двовимірну карту, влаштовану таким чином, що близьким об'єктам в багатовимірному просторі відповідають точки (їхні образи) на карті, що стоять поруч. В результаті важко аналізовані в сукупності багатовимірні об'єкти отримують простий і наглядний вид на двовимірній карті, яка зберігає їх основні властивості (топологію і розподіл в багатовимірному просторі).

SOM передбачає використання впорядкованої структури нейронів. Зазвичай використовуються одно і двовимірні сітки. При цьому кожен нейрон являє собою n-мірний вектор-стовпець $w = [w_1, w_2, \ldots, w_n]^T$, де n визначається розмірністю вихідного простору (розмірністю вхідних векторів).

Зазвичай нейрони розташовуються у вузлах двовимірної сітки з прямокутними або шестикутними клітинками. При цьому, як було сказано вище, нейрони також взаємодіють один з одним. Величина цієї взаємодії визначається відстанню між нейронами на карті. На рис. 1.8 наводиться приклад відстані для шестикутної і чотирикутної сіток.



а) б)

Рис. 1.8 - Відстань між нейронами на карті для шестикутної (а) і чотирикутної (б) сіток.

При цьому легко помітити, що для шестикутної сітки відстань між нейронами більше збігається з евклідовою відстанню, ніж для чотирикутної сітки.

При реалізації алгоритму SOM заздалегідь задається конфігурація сітки (прямокутна або шестикутна), а також кількість нейронів у мережі. Деякі джерела рекомендують використовувати максимально можливу кількість нейронів в карті. При цьому початковий радіус навчання (neighborhood) в значній мірі впливає на здатність узагальнення за допомогою отриманої карти.

Перед початком навчання карти необхідно проініціалізувати вагові коефіцієнти нейронів. Вдало вибраний спосіб ініціалізації може істотно прискорити навчання, і привести до отримання більш якісних результатів.

Мережа Кохонена навчається методом послідовних наближень. В процесі навчання таких мереж на входи подаються дані, але мережа при цьому підлаштовується не під еталонне значення виходу, а під закономірності у вхідних даних. Починається навчання з обраного випадковим чином вихідного розташування центрів. В процесі послідовної подачі на вхід мережі навчальних прикладів визначається найбільш схожий нейрон (той, у якого скалярний добуток ваг і поданого на вхід вектора мінімальний). Цей нейрон оголошується переможцем і є центром при підстроюванні ваг у сусідніх нейронів. Таке правило навчання передбачає "змагальне" навчання з урахуванням відстані нейронів від "нейрона-переможця". Таким чином, якщо позначити нейрон-переможець як *c*, то отримаємо

$$x - w_c = \min_j \{x - w_j\}. \quad (1.10)$$

Навчання при цьому полягає не в мінімізації помилки, а в підстроюванні ваг (внутрішніх параметрів нейронної мережі) для найбільшого збігу з вхідними даними. При цьому вектор, що описує нейрон-переможець і вектора, що описують його сусідів у сітці переміщаються в напрямку вхідного вектора. Це проілюстровано на рис. 1.9 для двовимірного вектора.

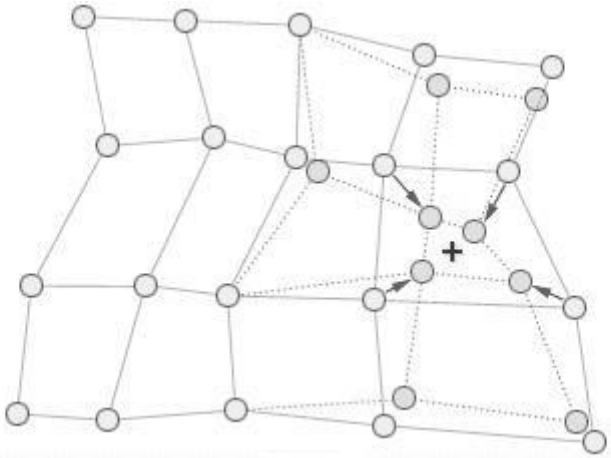


Рис. 1.9 - Підстроювання вагів нейрона переможця і його сусідів.

Координати вхідного вектора відзначені хрестом, вид сітки після модифікації відображений штриховими лініями.

При цьому для модифікації вагових коефіцієнтів використовується формула:

$$w_i(t+1) = w_i(t) + h_{ci} * \left[x(t) - w(t) \right], \quad (1.11)$$

де *t* позначає номер епохи (дискретний час). При цьому вектор *x(t)* вибирається випадково з навчальної вибірки на ітерації *t*. Функція *h(t)* називається функцією сусідства нейронів. Ця функція являє собою функцію, що не зростає від часу і відстані між нейроном-переможцем і сусідніми нейронами в сітці. Ця функція розбивається на дві частини: власне функцію відстані і функцію швидкості навчання від часу.

Зазвичай застосовується одна з двох функцій від відстані: проста константа

$$h(d,t)=\begin{cases} const, d \le \sigma(t) \\ 0, d > \sigma(t) \end{cases}, \quad (1.12)$$

або Гаусова функція

$$h(d,t)=e^{-\frac{d^2}{2\sigma^2(t)}}. \quad (1.13)$$

При цьому кращий результат виходить при використанні гауссовой функції відстані. При цьому є спадною функцією від часу. Часто цю величину називають радіусом навчання, який вибирається досить великим на початковому етапі навчання і поступово зменшується так, що в кінцевому випадку навчається один нейрон-переможець. Найбільш часто використовується функція лінійно спадна від часу.

Розглянемо тепер функцію швидкості навчання *a(t)*. Ця функція також являє собою функцію спадну від часу. Найбільш часто використовуються два варіанти цієї функції: лінійна і обернено пропорційна часу виду $a(t)=\frac{A}{t+B}$, де *A* і *B* це константи. Застосування цієї функції призводить до того, що всі вектори з навчальної вибірки вносять приблизно рівний внесок у результат навчання.

Навчання складається з двох основних фаз: на первісному етапі вибирається досить велике значення швидкості навчання і радіуса навчання, що дозволяє розташувати вектора нейронів відповідно до розподілу прикладів у вибірці, а потім здійснюється точне підстроювання вагів, коли значення параметрів швидкості навчання багато менше початкових.

Основний ітераційний алгоритм Кохонена послідовно проходить ряд епох, на кожній з яких обробляється один приклад з навчальної вибірки. Вхідні сигнали послідовно пред'являються мережі, при цьому бажані вихідні сигнали не визначаються. Після пред'явлення достатнього числа вхідних векторів синаптичні ваги мережі стають здатні визначити кластери. Ваги організуються так, що топологічно близькі вузли чутливі до схожих вхідних сигналів.

Отриману карту можна використовувати як засіб візуалізації при аналізі даних. В результаті навчання карта Кохонена класифікує вхідні приклади на кластери (групи схожих прикладів) і візуально відображає багатовимірні вхідні

дані на площині нейронів.

Відображення кластерів. Кластером буде група векторів, відстань між якими всередині цієї групи менше, ніж відстань до сусідніх груп. Структура кластерів при використанні алгоритму SOM може бути відображена шляхом візуалізації відстані між опорними векторами (ваговими коефіцієнтами нейронів). При використанні цього методу найчастіше використовується уніфікована матриця відстаней (*u-matrix*). Обчислюється відстань між вектором ваг нейрона в сітці і його найближчими сусідами. Потім ці значення використовуються для визначення кольору, яким цей вузол буде зображений.

1.5. Огляд технології OLAP у сучасних системах підтримки прийняття рішень.

До теперішнього часу в багатьох організаціях накопичені значні обсяги даних, на основі яких є можливість вирішення різних аналітичних і управлінських задач. Проблеми збереження та обробки аналітичної інформації стають все більш актуальними і привертають увагу фахівців і фірм, що працюють в області інформаційних технологій. Це призвело до формування повноцінного ринку технологій бізнес-аналізу.

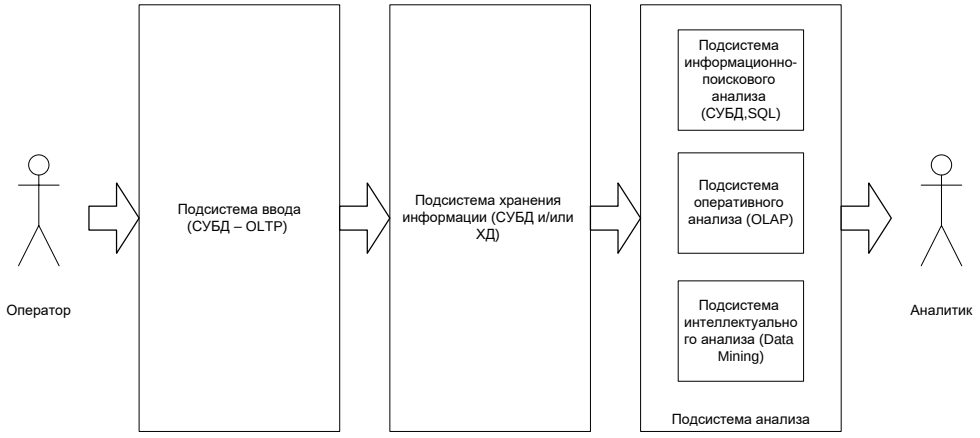


Рис. 1.10 – Архітектура СППР

Збір та зберігання інформації, а також вирішення задач інформаційно-

пошукового запиту ефективно реалізуються засобами систем керування базами даних (СКБД). У OLTP (Online Transaction Processing) - підсистемах реалізується транзакційна обробка даних. Безпосередньо OLTP - системи не підходять для повноцінного аналізу інформації через суперечливість вимог, що пред'являються до OLTP - систем і СППР.

Для надання необхідної для прийняття рішень інформації зазвичай доводиться збирати дані з декількох транзакційних баз даних різної структури і змісту. Основна проблема при цьому полягає в неузгодженості та суперечливості цих баз - джерел, відсутності єдиного логічного погляду на корпоративні дані.

Тому для об'єднання в одній системі OLTP і СППР для реалізації підсистеми зберігання використовується концепція сховищ даних (СД). В основі концепції СД знаходиться ідея розділення даних, що використовуються для оперативної обробки і для вирішення задач аналізу, що дозволяє оптимізувати структури зберігання. СД дозволяє інтегрувати раніше роз'єднані деталізовані дані, що містяться в історичних архівах, накопичуваних в традиційних OLTP - системах, що надходять із зовнішніх джерел, в єдину базу даних, здійснюючи їх попереднє узгодження і, можливо, агрегацію.

Підсистема аналізу може бути побудована на основі:

- підсистеми інформаційно-пошукового аналізу на базі реляційних СКБД і статичних запитів з використанням мови SQL;

- підсистеми оперативного аналізу. Для реалізації таких підсистем застосовується технологія оперативної аналітичної обробки даних OLAP, що використовує концепцію багатовимірного представлення даних;

- підсистеми інтелектуального аналізу, що реалізують методи і алгоритми Data Mining.

Прийняти будь-яке управлінське рішення неможливо не володіючи необхідною для цього інформацією, зазвичай кількісною. Для цього необхідне створення сховищ даних (Data warehouses), тобто процес збору, відсіювання та попередньої обробки даних з метою надання результуючої інформації користувачам для статистичного аналізу (а нерідко і створення аналітичних звітів).

Ральф Кімбол (Ralph Kimball), один з авторів концепції сховищ даних, описував сховище даних як "місце, де люди можуть отримати доступ до своїх даних" [25]. Він же сформулював і основні вимоги до сховищ даних [26]:

- підтримка високої швидкості отримання даних зі сховища;
- підтримка внутрішньої несуперечності даних;
- можливість отримання і порівняння так званих зрізів даних (slice and dice);
- наявність зручних утиліт перегляду даних у сховищі;
- повнота і достовірність даних, що зберігаються;
- підтримка якісного процесу поповнення даних.

Задовольняти всім перерахованим вимогам у рамках одного і того ж продукту часто не вдається. Тому для реалізації сховищ даних зазвичай використовується кілька продуктів, одні з яких є власне засоби зберігання даних, інші - засоби їх отримання і перегляду, треті - засоби їх поповнення і т.д.

Типове сховище даних, як правило, відрізняється від звичайної реляційної бази даних. *По-перше*, звичайні бази даних призначені для того, щоб допомогти користувачам виконувати повсякденну роботу, тоді як сховища даних призначені для прийняття рішень. Наприклад, продаж товару і виписування рахунку здійснюються з використанням бази даних, яка призначена для опрацювання транзакцій, а аналіз динаміки продажів за декілька років, що дозволяє спланувати роботу з постачальниками, - за допомогою сховища даних. *По-друге*, звичайні бази даних постійно змінюються в процесі роботи користувачів, а сховище даних відносно стабільне: дані у ньому зазвичай оновлюються за розкладом (наприклад, щотижня, щодня або щогодини - залежно від потреб). В ідеалі процес поповнення являє собою просто додавання нових даних за певний період часу без зміни попередньої інформації, що вже знаходиться у сховищі. І *по-третє*, звичайні бази даних найчастіше є джерелом даних, що потрапляють у сховище. Крім того, сховище може поповнюватися за рахунок зовнішніх джерел, наприклад статистичних звітів.

OLAP (Online Analytical Processing) – технологія оперативної аналітичної

обробки даних, що використовує методи і засоби для збору, зберігання та аналізу багатовимірних даних з метою підтримки процесів прийняття рішень.

Основне призначення OLAP-систем – підтримка аналітичної діяльності, довільних запитів користувачів - аналітиків. Мета OLAP - аналізу – перевірка гіпотез,що виникають.

Системи підтримки прийняття рішень зазвичай володіють засобами надання користувачу агрегатних даних для різних вибірок з вихідного набору в зручному для сприйняття та аналізу вигляді. Як правило, такі агрегатні функції утворюють багатовимірний (і, отже, нереляційний) набір даних (нерідко званий гіперкубом або метакубом), осі якого містять параметри, а клітинки - залежні від них агрегатні дані - причому зберігатися такі дані можуть і в реляційних таблицях, але в даному випадку ми говоримо про логічну організацію даних, а не про фізичну реалізацію їх зберігання). Уздовж кожної осі дані можуть бути організовані у вигляді ієрархії, що представляє різні рівні їх деталізації. Завдяки такій моделі даних користувачі можуть формулювати складні запити, генерувати звіти, отримувати підмножини даних.

Концепція OLAP була описана в 1993 році Едгаром Коддом, відомим дослідником баз даних і автором реляційної моделі даних [27]. У 1995 році на основі вимог, викладених Коддом, був сформульований так званий тест FASMI (Fast Analysis of Shared Multidimensional Information - швидкий аналіз колективної багатовимірної інформації), що включає наступні вимоги до засобів для багатовимірного аналізу:

- надання користувачу результатів аналізу за прийнятний час (звичайно не більше 5 с), нехай навіть ціною менш детального аналізу;

- можливість здійснення будь-якого логічного та статистичного аналізу, характерного для даного застосування, і його збереження в доступному для кінцевого користувача вигляді;

- багатокористувацький доступ до даних з підтримкою відповідних механізмів блокувань і засобів авторизованого доступу;

- багатовимірне концептуальне уявлення даних, включаючи повну підтримку ієрархій та множинних ієрархій (це - ключова вимога OLAP);

- можливість звертатися до будь-якої потрібної інформації незалежно від її обсягу та місця зберігання.

Слід зазначити, що OLAP-функціональність може бути реалізована різними способами, починаючи з найпростіших засобів аналізу даних в офісному застосуванні і закінчуючи розподіленими аналітичними системами, що засновані на серверних продуктах.

В OLAP користувач отримує природну, інтуїтивно зрозумілу модель даних, представлену у вигляді багатовимірних кубів (Cubes).

У процесі аналізу даних часто виникає необхідність побудови залежностей між різними параметрами, число яких може бути значним.

Під виміром будемо розуміти послідовність значень одного з параметрів, що аналізуються. Наприклад, для параметра «час» це - послідовність днів, місяців, кварталів, років.

Можливість аналізу залежностей між різними параметрами передбачає можливість подання даних у вигляді багатовимірної моделі - гіперкуба (Рис. 1.11), або OLAP-куба.

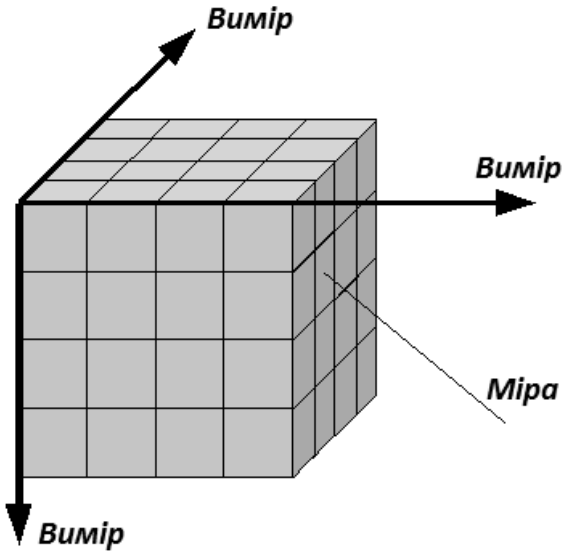


Рис. 1.11 – Гіперкуб

Осями куба є виміри, за якими відкладають параметри, пов'язані з предметною областю, що аналізується, наприклад, назви товарів і назви місяців року. На перетині осей вимірів розташовуються дані, які кількісно характеризують факти, що аналізуються - міри, наприклад, обсяги продажів, виражені в одиницях продукції.

У найпростішому випадку двовимірного куба виходить таблиця, що показує значення рівнів продажів по товарах і місяцях. В клітинках OLAP-куба можуть міститися не тільки суми, а й результати виконання інших агрегатних функцій мови SQL, таких як *MIN, MAX, AVG, COUNT*, а в деяких випадках - і інших (дисперсії, середньоквадратичного відхилення і т.д.).

Подальше ускладнення моделі даних можливо за декількома напрямками:

- збільшення числа вимірів - дані про продажі не тільки по місяцях і товарах, а й по регіонах. У цьому випадку куб стає тривимірним;

- ускладнення вмісту клітинки - наприклад, нас може цікавити не тільки рівень продажів, але і чистий прибуток або залишок на складі. В цьому випадку в клітинці буде кілька значень;

- введення ієрархії в межах одного виміру - загальне поняття «час» пов'язане з ієрархією значень: рік складається з кварталів, квартал з місяців і т.д.

Кожен з вимірів OLAP - куба може бути представлений у вигляді ієрархічної структури. Наприклад, вимір «Регіон» може мати такі рівні ієрархії: «країна - область - місто – район». Деякі виміри можуть мати кілька рівнів ієрархічного подання, наприклад вимір «час» - подання «рік - квартал - місяць - день» та подання «рік - тиждень - день». Так само в рамках виміру «Географія» можна ввести рівні «Країна», «Регіон», «Область» і «Місто».

Над гіперкубом можуть виконуватися такі операції:

Зріз (рис 1.12) - формується підмножина багатовимірного масиву даних, яка відповідає єдиному значенню одного або декількох елементів вимірів, що не входять в цю підмножину.

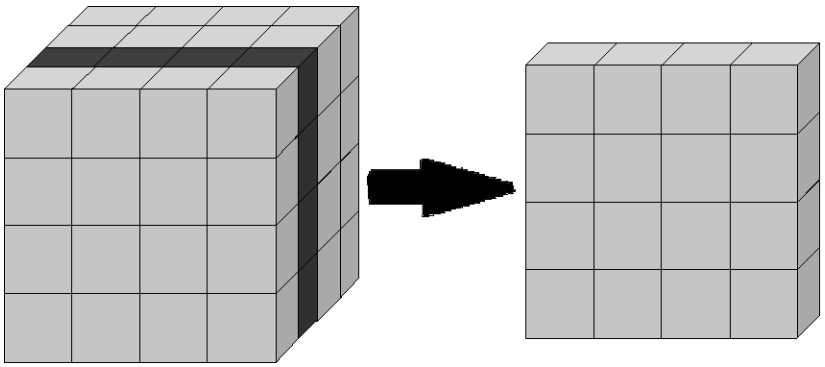


Рис. 1.12 – Зріз

Обертання (рис 1.13) - зміна розташування вимірів, представлених у звіті або на сторінці, що відображається. Наприклад, операція обертання може полягати в перестановці місцями рядків і стовпців таблиці.

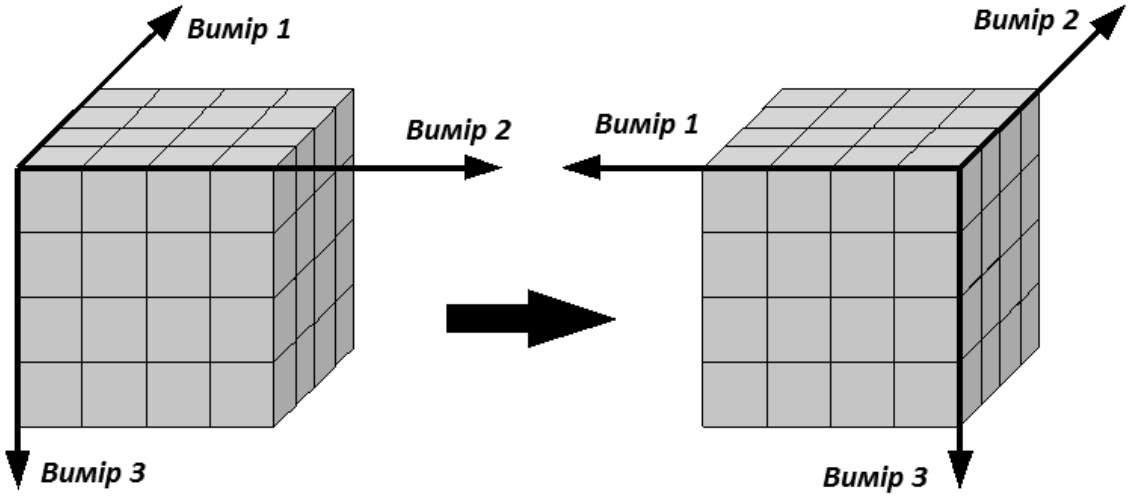


Рис. 1.13 – Обертання

Консолідація (рис 1.14) і *деталізація* (рис 1.15) - операції, які визначають перехід вгору за напрямком від детального подання даних до агрегованого і навпаки, відповідно. Напрямок деталізації (узагальнення) може бути задано як за ієрархією окремих вимірів, так і згідно іншим відношенням, встановленим в рамках вимірів або між вимірами.

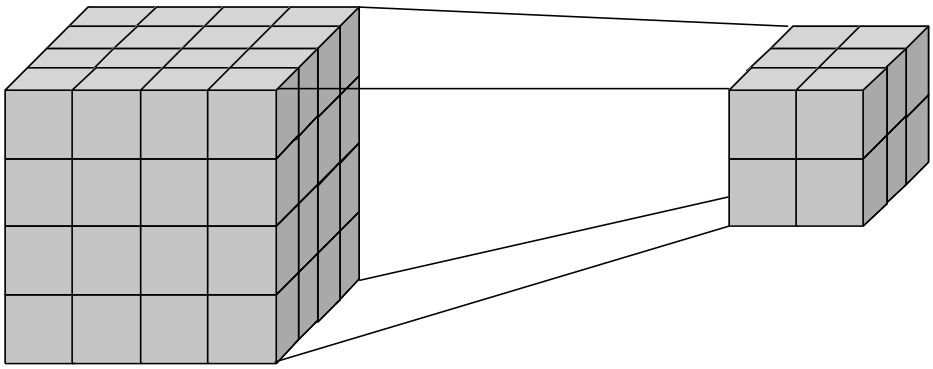


Рис. 1.14 – Консолідація

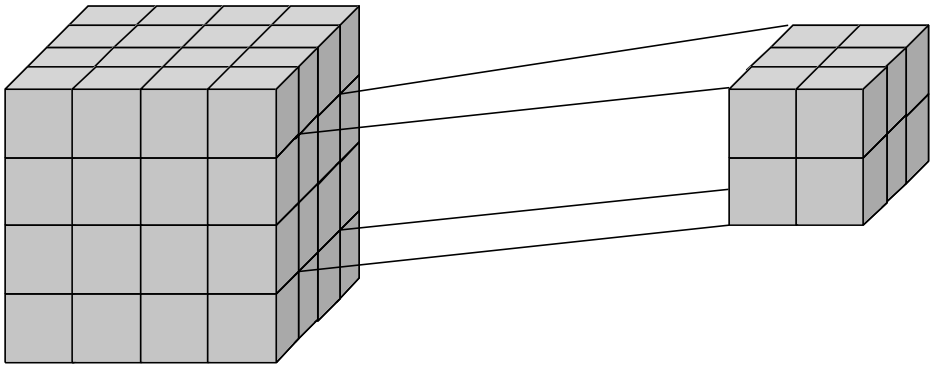


Рис. 1.15 – Деталізація

Наприклад, якщо при аналізі даних про продажі в Північній Америці виконати операцію деталізації для виміру «Регіон», то будуть відображені такі елементи, як «Канада», «Східні штати США» і «Західні штати США». В результаті подальшої деталізації елемента «Канада» будуть відображені елементи «Торонто», «Ванкувер» і т.д.

У загальному вигляді архітектура корпоративної OLAP - системи описується схемою з трьома виділеними рівнями (рис. 1.16):

- отримання, перетворення і завантаження даних;
- зберігання даних;
- аналіз даних.

Рис. 1.16 Архітектура корпоративної OLAP-системи

Дані надходять із різних внутрішніх OLTP-систем, від підлеглих структур, від зовнішніх організацій відповідно до встановленого регламенту, форм і макетів звітності. Вся ця інформація перевіряється, узгоджується, перетворюється і поміщається в сховище і вітрини даних. Після цього користувачі за допомогою спеціалізованих інструментальних засобів отримують необхідну їм інформацію для побудови табличного і графічного подання, прогнозування, моделювання та виконання інших аналітичних задач.

В багатовимірних сховищах даних містяться агрегатні дані різного ступеня деталізації, наприклад, обсяги продажів по днях, місяцях, роках, за категоріями товарів і т.п. Мета зберігання агрегатних даних - скоротити час виконання запитів, оскільки в більшості випадків для аналізу та прогнозів цікаві не дета-

льні, а сумарні дані. Тому при створенні багатовимірної бази даних завжди обчислюються і зберігаються деякі агрегатні дані.

Зазначимо, що збереження всіх агрегатних даних не завжди виправдано. Справа в тому, що при додаванні нових вимірів обсяг даних, що складають куб, зростає експоненційно (іноді говорять про «вибухове зростання» обсягу даних). Якщо говорити більш точно, ступінь зростання обсягу агрегатних даних залежить від кількості вимірів куба і членів вимірів на різних рівнях ієрархій цих вимірів. Для вирішення проблеми «вибухового зростання» застосовуються різноманітні схеми, які дозволяють при обчисленні далеко не всіх можливих агрегатних даних досягти прийнятної швидкості виконання запитів.

Як вихідні, так і агрегатні дані можуть зберігатися або в реляційних, або в багатовимірних структурах. Тому на сьогодні застосовуються три способи зберігання даних:

- MOLAP (Multidimensional OLAP) - вихідні і агрегатні дані зберігаються в багатовимірній базі даних. Зберігання даних в багатовимірних структурах дозволяє маніпулювати даними як багатовимірним масивом, завдяки чому швидкість обчислення агрегатних значень однакова для будь-якого з вимірів. Проте в цьому випадку багатовимірна база даних виявляється надлишковою, так як багатовимірні дані повністю містять вихідні реляційні дані.

- ROLAP (Relational OLAP) - вихідні дані залишаються в тій же реляційній базі даних, де вони спочатку і знаходилися. Агрегатні ж дані поміщають в спеціально створені для їх зберігання службові таблиці в тій же базі даних.

- HOLAP (Hybrid OLAP) - вихідні дані залишаються в тій же реляційній базі даних, де вони спочатку знаходилися, а агрегатні дані зберігаються в багатовимірній базі даних.

Деякі OLAP-засоби підтримують зберігання даних лише в реляційних структурах, деякі - тільки в багатовимірних. Однак більшість сучасних серверних OLAP-засобів підтримують всі три способи зберігання даних. Вибір способу зберігання залежить від обсягу і структури вихідних даних, вимог до швидкості виконання запитів та частоти оновлення OLAP-кубів.

Багатовимірний аналіз даних може бути проведений за допомогою різних засобів, які умовно можна розділити на клієнтські і серверні OLAP-засоби.

Клієнтські OLAP-засоби являють собою програми, які здійснюють обчислення агрегатних даних (сум, середніх величин, максимальних або мінімальних значень) та їх відображення, при цьому самі агрегатні дані містяться в кеші всередині адресного простору такого OLAP-засобу.

Якщо вихідні дані містяться в настільній СКБД, обчислення агрегатних даних проводиться самим OLAP-засобом. Якщо ж джерело вихідних даних – серверна СКБД, багато з клієнтських OLAP-засобів посилають на сервер *SQL*-запити, що містять оператор *GROUP BY*, і в результаті отримують агрегатні дані, обчислені на сервері.

Переваги застосування серверних OLAP- засобів в порівнянні з клієнтськими OLAP-засобами подібні з перевагами застосування серверних СКБД порівняно з настільними: у разі застосування серверних засобів обчислення і зберігання агрегатних даних відбувається на сервері, а клієнтський додаток отримує лише результати запитів до них, що дозволяє в загальному випадку знизити мережевий трафік, час виконання запитів і вимоги до ресурсів, що споживаються клієнтським додатком. Зазначимо, що засоби аналізу і обробки даних масштабу підприємства, як правило, базуються саме на серверних OLAP-засобах, наприклад, таких як Microsoft SQL Server 2008 R2 Analysis Services та ін.

Подібно до того, як *SQL* (Structured Query Language - мова структурованих запитів) являє собою мову створення запитів для отримання даних з реляційних баз даних, *MDX* (Multi-Dimensional eXpressions - мова багатовимірних виразів) є мовою запитів, яка використовується для отримання даних з багатовимірних баз даних. Точніше, *MDX* використовується для запиту даних з баз даних OLAP за допомогою Analysis Services і підтримує два особливих режими. При використанні в якості виразів *MDX* дозволяє визначати багатовимірні об'єкти і дані для обчислення значень, а також керувати ними. Як мова запитів він використовується для отримання даних з баз даних Analysis Services. Спочатку *MDX* був розроблений компанією Microsoft і був введений разом з Analysis

Services 7.0 у 1998 році. Використання мови *MDX* не обмежено авторськими правами на продукт Analysis Services. Ця мова використовується для отримання інформації з баз даних OLAP; він заснований на стандартах галузі. Мова є частиною специфікації OLEDB для OLAP, що фінансується Microsoft, він підтримується також багатьма іншими провайдерами OLAP, включаючи Intelligence Server компанії Microstrategy, Essbase Server компанії Hyperion і Enterprise BI Server від SAS.

1.6. Програмне забезпечення для аналізу даних WEKA.

WEKA - відкритий програмний продукт, що розвивається світовим науковим співтовариством, ПЗ написано на мові Java в університеті Вайкато (Нова Зеландія), поширюється під ліцензією GNU General Public License version 21 (GPLv2) і надає користувачеві можливість передобробки даних, рішення задач класифікації, регресії, кластеризації і пошуку асоціативних правил, а також візуалізації даних і результатів. Програма дуже проста в освоєнні (мабуть, має самий інтуїтивний інтерфейс серед усіх програм такого типу), безкоштовна і може бути доповнена новими алгоритмами, засобами передобробки і візуалізації даних.

Weka дозволяє виконувати такі задачі аналізу даних, як:

- підготовка даних - попередня обробка - preprocessing,
- відбір ознак - feature selection,
- кластеризація,
- класифікація,
- пошук асоціативних правил,
- регресійний аналіз,
- візуалізація результатів.

Weka добре підходить для розробки нових підходів в машинному навчанні, оскільки Weka - набір засобів візуалізації і бібліотека алгоритмів машинного

навчання для вирішення задач інтелектуального аналізу даних (data mining) та прогнозування, з графічною користувальницької оболонкою для доступу до них. Система дозволяє безпосередньо застосовувати алгоритми до вибірок даних, а також викликати алгоритми з програм на мові Java. На сьогоднішній день це найкраща Open source бібліотека для Data Mining. Користувачами Weka є дослідники в області машинного навчання і прикладних наук. Вона також широко використовується в навчальних цілях.

Передбачається, що вихідні дані представлені у вигляді матриці ознакових описів об'єктів. Weka надає доступ до SQL-баз через Java Database Connectivity (JDBC) і в якості вихідних даних може приймати результат SQL-запиту. Можливість обробки множини пов'язаних таблиць не підтримується, але існують утиліти для перетворення таких даних в одну таблицю, яку можна завантажити в Weka.

Weka має користувальницький інтерфейс Explorer, але та ж функціональність доступна через компонентний інтерфейс Knowledge Flow і з командного рядка. Є окремий додаток Experimenter для порівняння прогнозуючої здібності алгоритмів машинного навчання на заданому наборі задач.

Explorer має декілька панелей.

Панель передобробки Preprocess panel дозволяє імпортувати дані з бази, CSV файлу і т.д., і застосовувати до них алгоритми фільтрації, наприклад, перекладати кількісні ознаки в дискретні, видаляти об'єкти і ознаки за заданим критерієм.

Панель класифікації Classify panel дозволяє застосовувати алгоритми класифікації і регресії (в Weka вони не розрізняються і називаються classifiers) до вибірки даних, оцінювати прогнозуючу здатність алгоритмів, візуалізувати помилкові прогнози, ROC-криві, і сам алгоритм, якщо це можливо (зокрема, дерева рішень).

Панель пошуку асоціативних правил Associate panel вирішує задачу виявлення всіх значущих взаємозв'язків між ознаками.

Панель кластеризації Cluster panel дає доступ до алгоритму k-середніх,

EM-алгоритму для суміші гауссіанов та іншим.

Панель відбору ознак Select attributes panel дає доступ до методів відбору ознак.

Панель візуалізації Visualize будує матрицю графіків розкиду (scatter plot matrix), дозволяє вибирати і збільшувати графіки, і т.д.

Weka надає прямий доступ до бібліотеки реалізованих у ній алгоритмів. Це дозволяє легко використовувати вже реалізовані алгоритми з інших систем, реалізованих на Java. Наприклад, ці алгоритми можна викликати з MATLAB.

Для використання Weka з систем, реалізованих на інших платформах, можливий виклик алгоритмів через інтерфейс командного рядка.

1.7. Висновки до розділу. Постановка завдання.

Резюмуючи викладене в інформаційно-теоретичному розділі, можна зробити наступні висновки:

1. Останнім часом стрімко поширюються зловмисні мережі, підвищується інтелектуальність мережевих атак, зростає організована кіберзлочинність та шпигунство з використанням всесвітньої мережі Інтернет, а також з використанням зростаючої мережі мобільних систем. Що в свою чергу, призводять до нових форм додаткових загроз, більш витонченим способам витоку інформації, у тому числі інсайдерських атак, і це далеко не повний перелік різноманіття і складності реальних загроз, до яких схильні інформаційні мережі підприємств. Тому задача класифікації мережевих з’єднань на підприємстві чи в установі є актуальною і потребує застосування сучасних алгоритмів.

2. Кількість переданих за 1 секунду пакетів (PPS) в 1 кварталі 2012 року збільшилася в чотири рази в порівнянні з четвертим кварталом 2011 року. Такого порядку темпи росту спостерігаються останні 5-6 років. Це обумовлено підвищенням пропускної здатності каналів та збільшенням потужності зловмисних мереж. Для аналізу інформації про з’єднання, що надходить у кількості 500 000

– 1 000 000 записів на тиждень, необхідно використовувати сучасні сховища даних. Сховища даних повинні забезпечувати високу швидкість отримання даних, можливість отримання і порівняння так званих зрізів даних, а також несуперечність, повноту і достовірність даних.

3. OLAP є ключовим компонентом побудови та застосування сховищ даних. Ця технологія заснована на побудові багатовимірних наборів даних - OLAP-кубів, осі якого містять параметри, а клітинки - залежні від них агрегатні дані. Програми з OLAP-функціональністю повинні надавати користувачеві результати аналізу за прийнятний час, здійснювати логічний і статистичний аналіз, підтримувати багатокористувацький доступ до даних, здійснювати багатовимірне концептуальне подання даних і мати можливість звертатися до будь-якої потрібної інформації.

4. У якості вихідних даних для роботи були розглянуті відомості про мережеву активність локальної мережі, підготовлені керуючою компанією MIT Lincoln Labs за звітами системних адміністраторів підприємства за дев’ять тижнів 2009 року (набір поділений на близько 5 мільйонів записів навчальної вибірки та близько 2 мільйонів записів тестової вибірки). Близько 20% з’єднань – нормальні. Решта – спроби несанкціонованих вторгнень різних видів (23 види).

5. Кожне з’єднання описано в базі вхідних даних 42-ма параметрами символьного, логічного та дійсного типів даних. Про наявність апріорного зв’язку між тим чи іншим параметром та видом з’єднання не відомо.

6. Обрана для вирішення проблема ідентифікації типу з’єднання пов’язана з побудовою моделі класифікації типів з’єднань на основі відомих параметрів.

7. В ході аналізу предметної області було з’ясовано, що задача ідентифікації типів з’єднань найкраще вирішується із застосуванням методу агломеративної кластеризації (для кластеризації з’єднань, описаних символьними параметрами) та багатовимірної логістичної регресії для побудови безпосередньо моделі ідентифікації.

8. У якості інструментарію для розв’язання проблеми було запропоновано

Weka - набір засобів візуалізації і бібліотека алгоритмів машинного навчання для вирішення задач інтелектуального аналізу даних (data mining) та прогнозування.

9. Підбиваючи підсумки, слід відзначити, що:

Об'єктом дослідження у кваліфікаційній роботі магістра є система захисту інформації локальної комп'ютерної мережі підприємства від небажаного вторгнення.

Водночас *предметом дослідження* є алгоритми класифікації небезпечних комп'ютерних з'єднань як багатопараметричних слабо визначених об'єктів з використанням технологій OLAP.

Мета дослідження – удосконалення алгоритму ідентифікації небажаного вторгнення в локальну мережу підприємства.

Для досягнення зазначеної мети необхідно вирішити наступні *науково-практичні задачі*:

- на основі аналізу структури початкових даних, типів даних та ступеню їх інформативності запропонувати структуру сховища даних, метрик та вимірів, звернення за якими до нього приймаються;
- визначити методи дослідження, програмне середовище та шляхи розв'язання задачі;
- розробити методологічні підходи до класифікації з'єднань на нормальні та небажані вторгнення, а також класифікації останніх за видами вторгнень;
- розробити та удосконалити стійкий алгоритм ідентифікації нормальних з'єднань та класифікації небажаних вторгнень;
- здійснити експериментальну перевірку розробленого алгоритму ідентифікації на тестовій виборці;
- розглянути інженерно-технічні заходи з охорони праці та обґрунтувати ергономічні параметри робочого місця користувача ПК.

СПЕЦІАЛЬНИЙ РОЗДІЛ

2.1. Формування структури сховища даних.

В інструменті SQL Server Management Studio (SSMS) створимо таблицю, яка буде містити вихідні дані. Для створення таблиці необхідно вибрати в контекстному меню гілки «Таблиці» вибраної бази даних пункт «Створити таблицю» (рис 2.1).

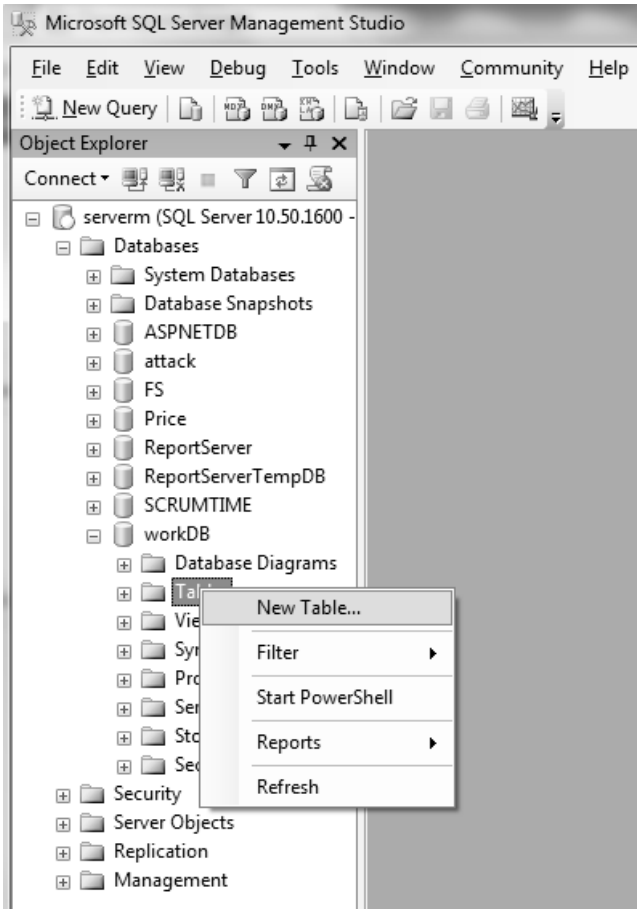


Рис. 2.1 – Створення таблиці

Назвемо таблицю kddcup. Далі необхідно ввести назви полів та типи даних. Існуючі типи даних в SQL Server приведені в таблиці 2.1:

Таблиця 2.1

Існуючі типи даних в SQL Server

bit	char	date	money	CLR
binary	nchar	datetime	smallmoney	hierarchyid
varbinary	varchar	datetime2	timestamp	rowversion
float	nvarchar	time	cursor	uniqueidentifier
real	numeric	smalldatetime	table	xml (Transact-SQL)
int, bigint, smallint і tinyint (Transact-SQL)	decimal	datetimeoffset	sql_variant	

Таблиця 2.2

Типи точних числових даних, що використовують цілі значення

Тип даних	Діапазон	Сховище
bigint	від -2^63 (-9 223 372 036 854 775 808) до 2^63-1 (9 223 372 036 854 775 807)	8 байт
int	від -2^31 (-2 147 483 648) до 2^31-1 (2 147 483 647)	4 байта
smallint	від -2^15 (-32 768) до 2^15-1 (32 767)	2 байти
tinyint	від 0 до 255	1 байт

Типи числових даних з фіксованими точністю і масштабом:

Decimal $\left[\left(p\left[,\, s\right]\right)\right]$ і numeric $\left[\left(p\left[,\, s\right]\right)\right]$.

При використанні максимальної точності числа можуть приймати значення в діапазоні від $-10^{38}+1$ до $10^{38}-1$.

Таблиця 2.3

Типи даних, що представляють грошові (валютні) значення

Тип даних	Діапазон	Сховище
money	від -922 337 203 685 477,5808 до 922 337 203 685 477,5807	8 байт
smallmoney	від -214 748,3648 до 214 748,3647	4 байта

Таблиця 2.4

Типи приблизних числових даних, що використовуються для числових даних з плаваючою комою

Тип даних	Діапазон	Сховище
float	-1,79E+308 — -2,23E-308, 0 і 2,23E-308 — 1,79E+308	Залежить від n
real	-3,40E+38 — -1,18E-38, 0 і 1,18E-38 — 3,40E+38	4 байта

binary [(n)] – двійкові дані фіксованої довжини розміром в *n* байт, де *n* – значення від 1 до 8000. Розмір зберігання становить *n* байт.

varbinary [(n|max)] – двійкові дані змінної довжини. *n* може мати значення від 1 до 8000; *max* означає максимальну довжину зберігання, яка становить 2^31-1 байт. Розмір зберігання – це фактична довжина введених даних плюс 2 байти.

Date описує дату. *Datetime* визначає дату, включає час дня з частками секунди в 24 – годинному форматі.

Функції *time, datetime2* і *datetimeoffset* дають велику точність секунд. Функція *datetimeoffset* підтримує часові пояси для додатків, розгорнутих по всьому світу.

nchar [(n)] - символьні дані в Юнікоді довжиною в *n* символів. Аргумент *n* повинен мати значення від 1 до 4000. Розмір сховища вдвічі більше *n* байт.

nvarchar [(n|max)] - символьні дані в Юнікоді змінної довжини. Аргумент *n* може приймати значення від 1 до 4000. Аргумент *max* вказує, що максимальний розмір сховища дорівнює 2^31-1 байт. Розмір сховища в байтах вдвічі більше числа введених символів + 2 байти.

Bit - цілочисельний тип даних, який може приймати значення 1, 0 або NULL.

char [(n)] - символьні дані фіксованої довжини, не в Юнікоді, з довжиною *n* байт. Значення *n* повинно бути в інтервалі від 1 до 8000. Розмір сховища дорівнює *n* байт.

varchar [(n|max)] - символьні дані змінної довжини, не в Юнікоді. *n* може мати значення від 1 до 8000. *max* означає, що максимальний розмір сховища дорівнює 2^31-1 байт. Розмір сховища дорівнює фактичній довжині даних плюс два байти.

Smalldatetime - визначає дату, що поєднується з часом дня. Час представлено в 24 - годинному форматі з секундами, завжди рівними нулю (: 00), без часток секунд.

Маємо таку таблицю (рис. 2.2):

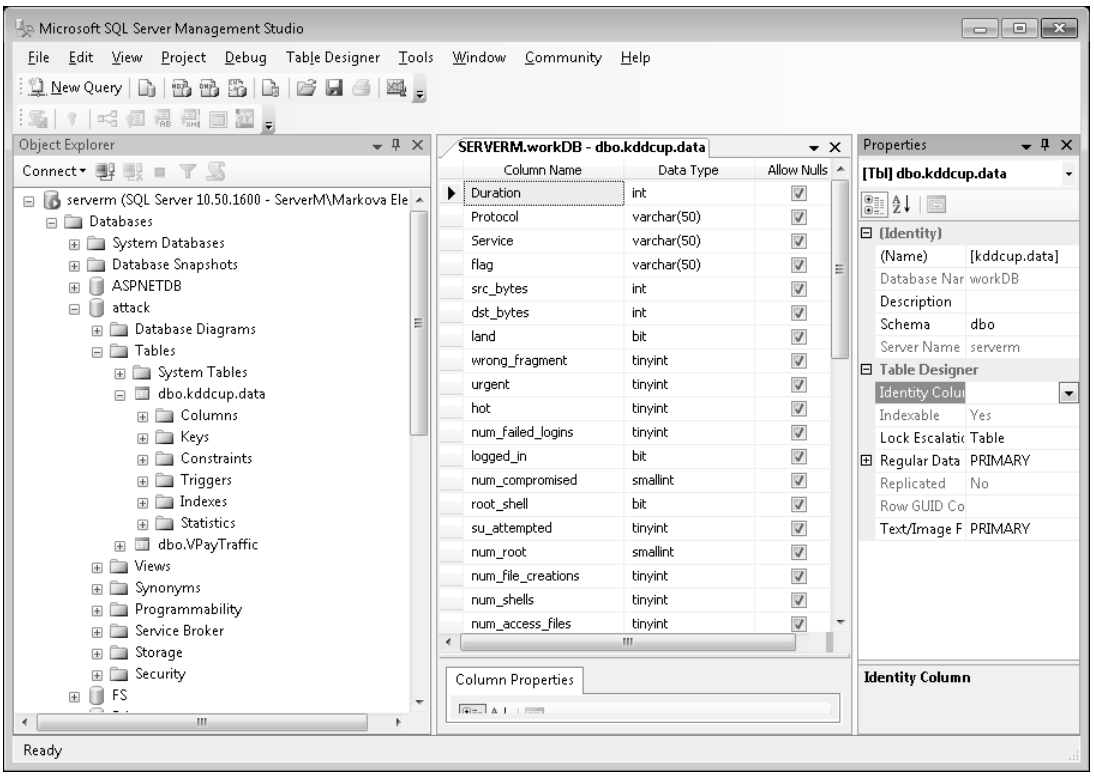


Рис. 2.2 – Поля та типи даних таблиці kddcup

Тепер необхідно імпортувати дані в цю таблицю.

Майстер імпорту та експорту SQL Server використовується для копіювання та перетворення даних між джерелами даних, які підтримуються, та місцями призначення імпорту або експорту.

Призначення майстра імпорту та експорту SQL Server полягає в копіюванні даних з вихідного розташування в цільове. Цей майстер може також створити цільову базу даних і цільові таблиці.

У середовищі SQL Server Management Studio підключаємось до сервера serverm типу Database Engine, розгортаємо бази даних, у контекстному меню обраної бази даних обираємо пункт Задачі – Імпорт даних.

Імпорт та експорт даних за допомогою майстра імпорту та експорту SQL Server включає наступні кроки.

1. Запускаємо майстер імпорту та експорту служб SQL Server.

2. На відповідних сторінках майстра вибираємо джерело даних і цільове призначення даних. Доступні такі джерела даних, як постачальники даних .NET Framework, постачальники OLE DB, власні клієнти-постачальники служб SQL

Server, ADO.NET, Microsoft Office Excel, Microsoft Office Access, а також джерело неструктурованих файлів. Обираємо джерело неструктурованих файлів. Доступні такі призначення, як постачальники даних .NET Framework, постачальники OLE DB, власний клієнт SQL Server, Excel, Access і призначення «неструктурований файл». Обираємо власний клієнт SQL Server. Задаємо параметри типу призначення.

3. Обираємо таблицю kddcup в якості таблиці, в яку будемо імпортувати дані.

Маємо заповнену таблицю kddcup, в якій знаходяться вихідні дані. Тепер переходимо до побудови аналітичного куба.

SQL Server Business Intelligence Development Studio (BI Dev Studio) - інструмент, призначений для розробки повноцінних систем бізнес-аналізу на основі Analysis Services, Reporting Services і Integration Services.

Програма BI Dev Studio інтегрується в оболонку Visual Studio, що дозволяє створювати додаткові типи проектів для SSAS.

Для створення проекту служб SSAS в середовищі BI Dev Studio необхідно виконати наступні кроки.

У меню «Файл» Visual Studio обираємо команду «Створити», потім вибираємо пункт «Проект». У діалоговому вікні «Новий проект» на панелі «Типи проектів» вибираємо значення «Проекти бізнес-аналітики», а на панелі «Шаблони» вказуємо «Проект служб SSAS». Змінюємо ім'я проекту на CubeDiplom.

Після створення проекту служб Analysis Services робота з проектом починається з визначення одного або декількох джерел даних, які будуть використовуватися в цьому проекті. Для визначення джерела даних потрібно задати рядок з’єднання, яке буде використане для підключення до цього джерела даних.

У браузері рішень у контекстному меню елементу «Джерела даних» вибираємо команду «Створити джерело даних». Визначаємо джерело даних на основі нового з'єднання. У списку «Постачальник» вибираємо «Власний постачальник даних OLE DB \ SQL Server Native Client 10.0».

В текстовому полі Ім'я сервера вводимо serverm. У списку «Виберіть або

введіть ім'я бази даних» вибираємо «WorkDB». Перевіряємо з'єднання з базою даних. На сторінці «Завершення роботи майстра» вводимо ім'я Attack.

Після визначення джерел даних, що використовуються в проекті служб Analysis Services, на наступному етапі визначаємо представлення джерела даних для цього проекту.

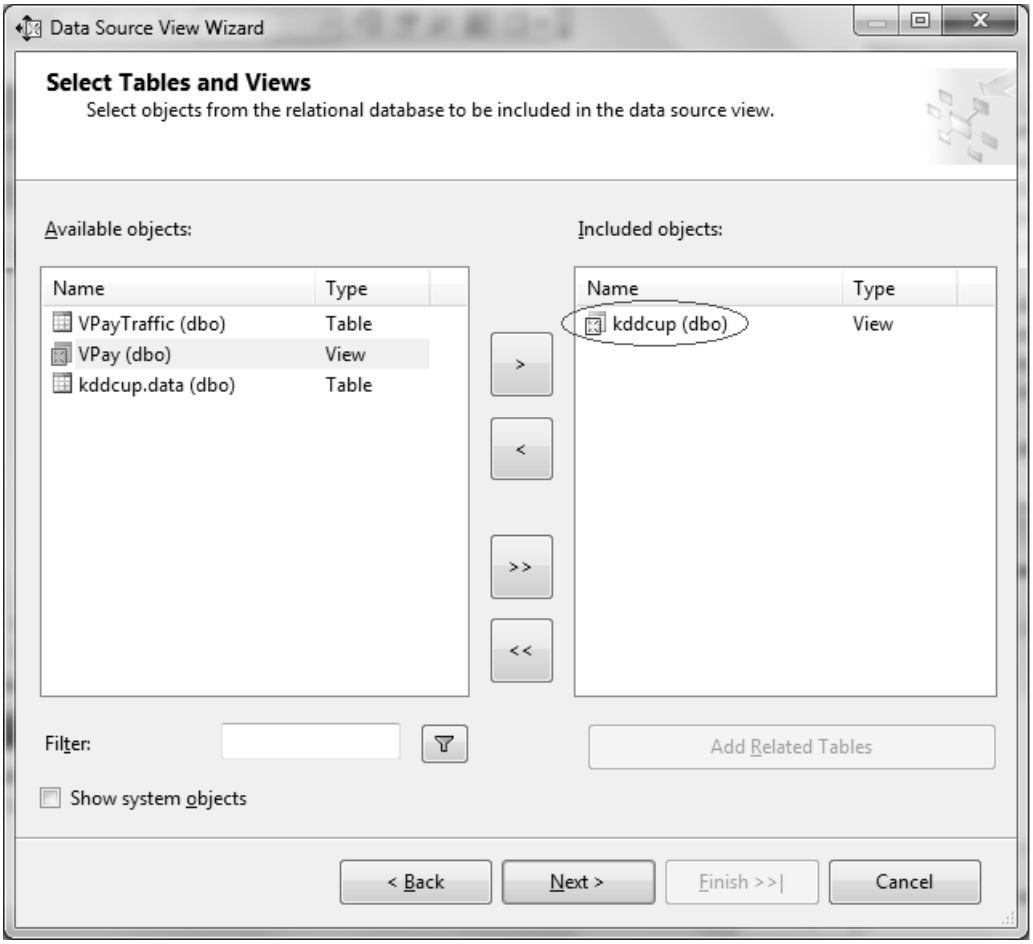


Рис. 2.3 – Вибір представлення

Представлення джерела даних є окремим єдиним представленням метаданих із зазначених таблиць і представлень, що визначаються джерелом даних для проекту. Зберігання метаданих в представленні джерела даних дозволяє працювати з метаданими в процесі розробки, не встановлюючи з'єднань з базовими джерелами даних. У браузері рішень у контекстному меню папки «Представлення джерел даних вибираємо пункт «Створити представлення джерела даних». На сторінці «Вибір джерела даних» у групі «Джерело реляційних даних» обрано джерело даних «Attack». На сторінці «Вибір таблиць і представ-

лень» можна вибрати таблиці та представлення зі списку об'єктів, доступних у вибраному джерелі даних. Можна встановити для цього списку фільтр, який допоможе відібрати потрібні таблиці й представлення.

У списку «Доступні об'єкти» вибираємо kddcup (dbo). У полі «Ім'я» вводимо Attack. Таким чином ми визначили представлення джерела даних Attack.

Після визначення представлення джерела даних в проекті служб Microsoft Analysis Services можна визначити вихідний куб служб Analysis Services.

Крім того, можна визначити куб і його виміри за один прохід за допомогою майстра кубів. Також можна визначити один або кілька вимірів, а потім за допомогою майстра кубів визначити куб, в якому вони будуть використовуватися. Розробку складного рішення зазвичай починають з визначення вимірів.

За допомогою майстра вимірів створимо вимір Protocol.

У браузері рішень у контекстному меню вузла «Виміри» вибираємо команду «Створити вимір». На сторінці «Вибір методу створення» вибираємо параметр «Використовувати існуючу таблицю». На сторінці «Визначення вихідних відомостей» вибрано представлення джерела даних Attack. У списку «Основна таблиця» обрана таблиця «kddcup».

На сторінці «Вибір атрибутів виміру» встановлюємо прапорець для атрибута Protocol. Завершуєм роботу майстра.

У браузері рішень в проекті «CubeDiplom» в папці «Виміри» з'явиться вимір «Protocol».

Аналогічно створимо виміри Service, Flag і Results. Вікно Solution Explorer набуде вигляд наведеного на рисунку 2.4. Зверніть увагу, що всі включені нами виміри знаходяться в папці Dimensions.

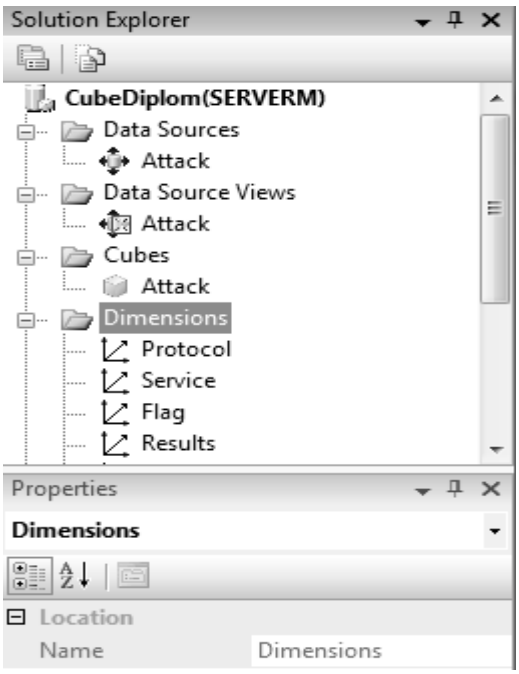


Рис. 2.4 – Виміри

Визначення куба і його властивостей. Майстер кубів допомагає визначити для куба групи мір та вимірювання. Далі за допомогою майстра кубів буде побудований куб. У браузері рішень в контекстному меню вузла «Куби» виберемо команду «Створити куб». На сторінці «Вибір методу створення» вибрано «Використовувати існуючі таблиці». На сторінці «Вибір таблиць груп мір» вибрано представлення джерела даних Attack у якості таблиці групи мір kddcup.

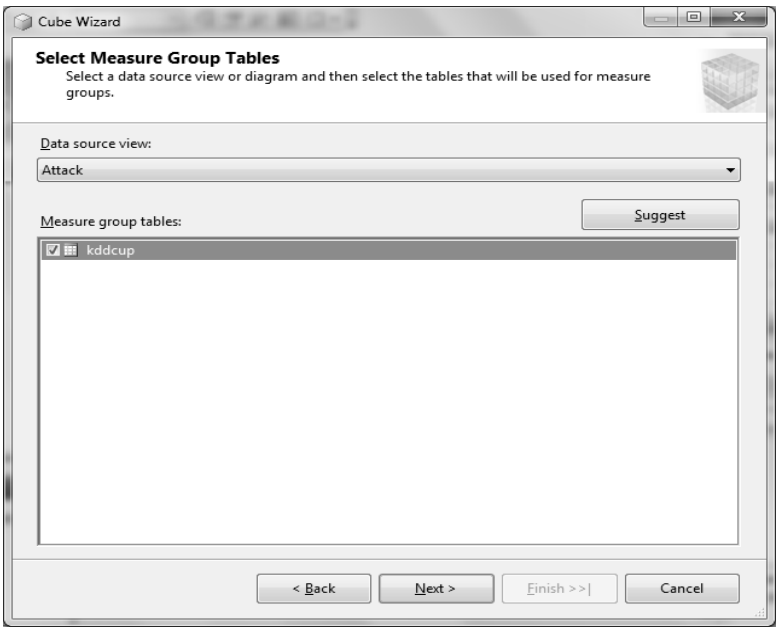


Рис. 2.5 – Вибір таблиць груп мір

На сторінці «Вибір мір» вибираємо Kddcupe Count.

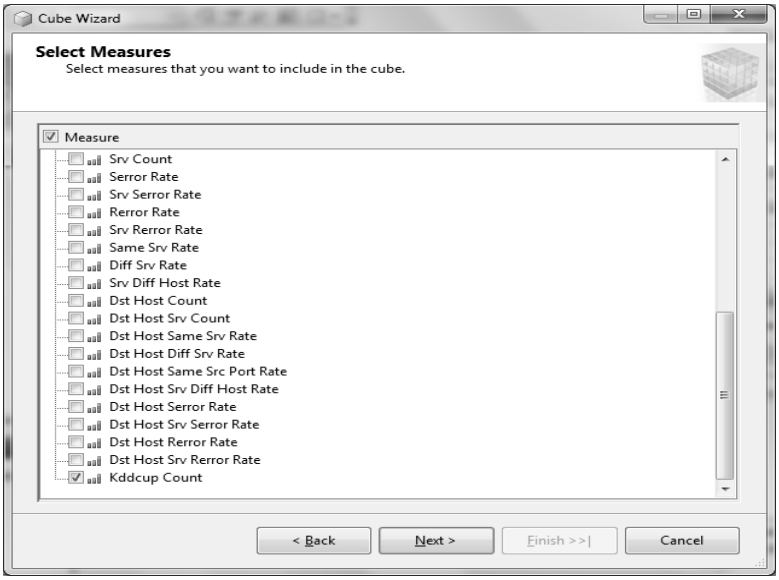


Рис. 2.6 – Вибір мір

На сторінці «Вибір існуючих вимірів» виберемо раніше створені виміри. На сторінці «Завершення роботи майстра» змінимо ім'я куба на «Attack».

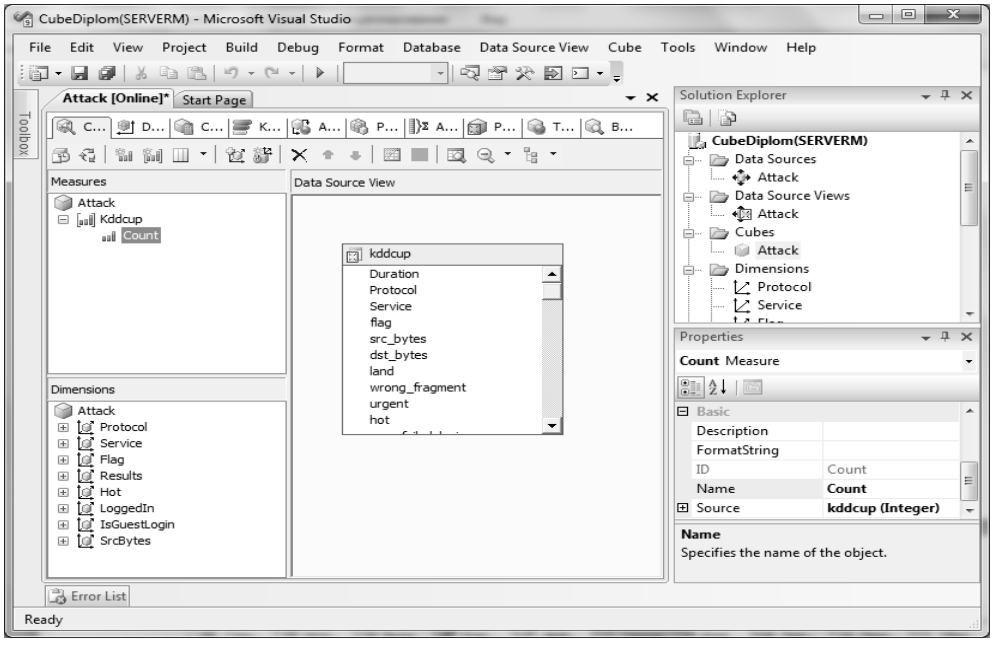


Рис. 2.7 – Конструктор кубів

В області «Міри» вкладки «Структура куба» конструктора кубів розкриємо групу мір «kddcupe». Змінимо міру count:

Рис. 2.8 – Визначення міри

Міра count показує кількість записів (рядків).

Додамо міри, які розраховуються: у вкладці Calculations конструктора кубів створимо новий обчислюваний член NormalPackets. Визначимо його як ([Measures].[Count],[Results].&[normal.]). Таким чином дана міра показує кількість нормальних з'єднань.

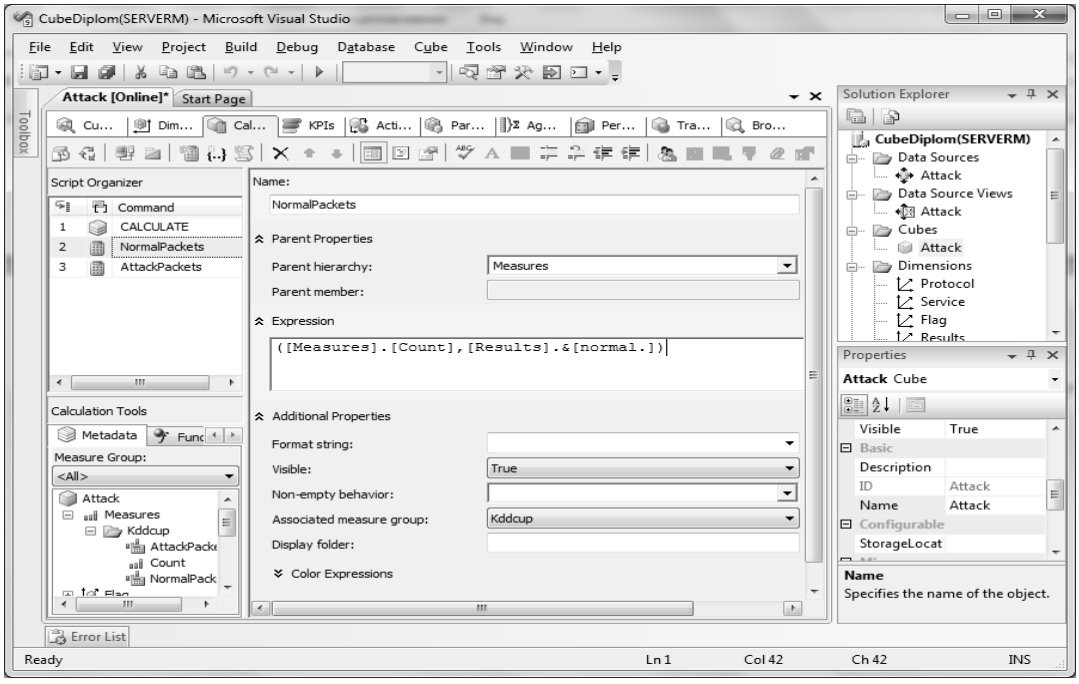


Рис. 2.9 – Створення обчислюваного члену NormalPackets

Аналогічно створимо обчислюваний член AttackPackets і визначимо його як ([Measures].[Count] - NormalPackets).

Таким чином цей обчислюваний член показує кількість зєднань, які є атаками.

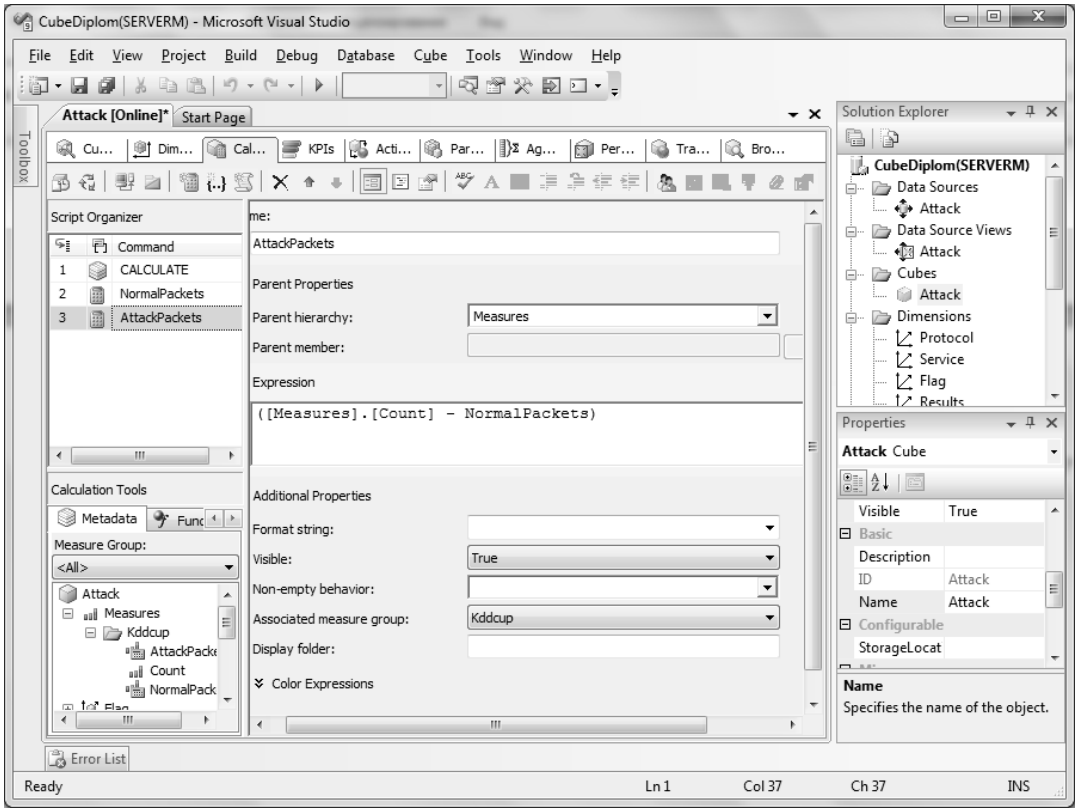


Рис. 2.10 – Створення обчислюваного члену AttackPackets

У майстрі кубів була визначена єдина секція для куба з використанням режиму зберігання результатів багатовимірної інтерактивної аналітичної обробки даних (MOLAP) без статистичних виразів. Для обробки MOLAP всі дані кінцевого рівня і всі статистичні вирази зберігаються в кубі, щоб забезпечити максимальну продуктивність. Статистичні вирази представляють собою попередньо обчислені зведені дані, які містять відповіді на ще не поставлені запитання, що дозволяє скоротити час до отримання відповіді на запит. На вкладці «Секції» можна визначати додаткові секції, параметри зберігання і настройки зворотного запису.

Щоб переглянути куб і дані виміри для об'єктів куба Attack проекту CubeDiplom, необхідно розгорнути проект на зазначеному екземплярі служб

Analysis Services, а потім виконати обробку куба і його вимірів. У процесі розгортання проекту служб Analysis Services в екземплярі служб Analysis Services створюються ті об'єкти, які були визначені. У процесі обробки об'єктів в екземплярі служб Analysis Services відбувається копіювання даних з базових джерел даних в об'єкти куба.

Після розгортання куба дані куба відображаються на вкладці «Браузер» конструктора кубів, а дані вимірів відображаються на вкладці «Браузер» конструктора вимірів.

Необхідно вказати, який зріз багатовимірного куба ми хочемо переглянути. Для цього слід вибрати виміри, які будуть відкладені по осях куба. Перетягнемо вибраний вимір Results в центральну частину вікна, в прямокутник «Перетягніть сюди поля стовпців». Далі виберемо міру count і перетягнемо її в прямокутник «Перетягніть сюди поля підсумків або деталей».

У результаті отримуємо кількість з'єднань кожного типу.

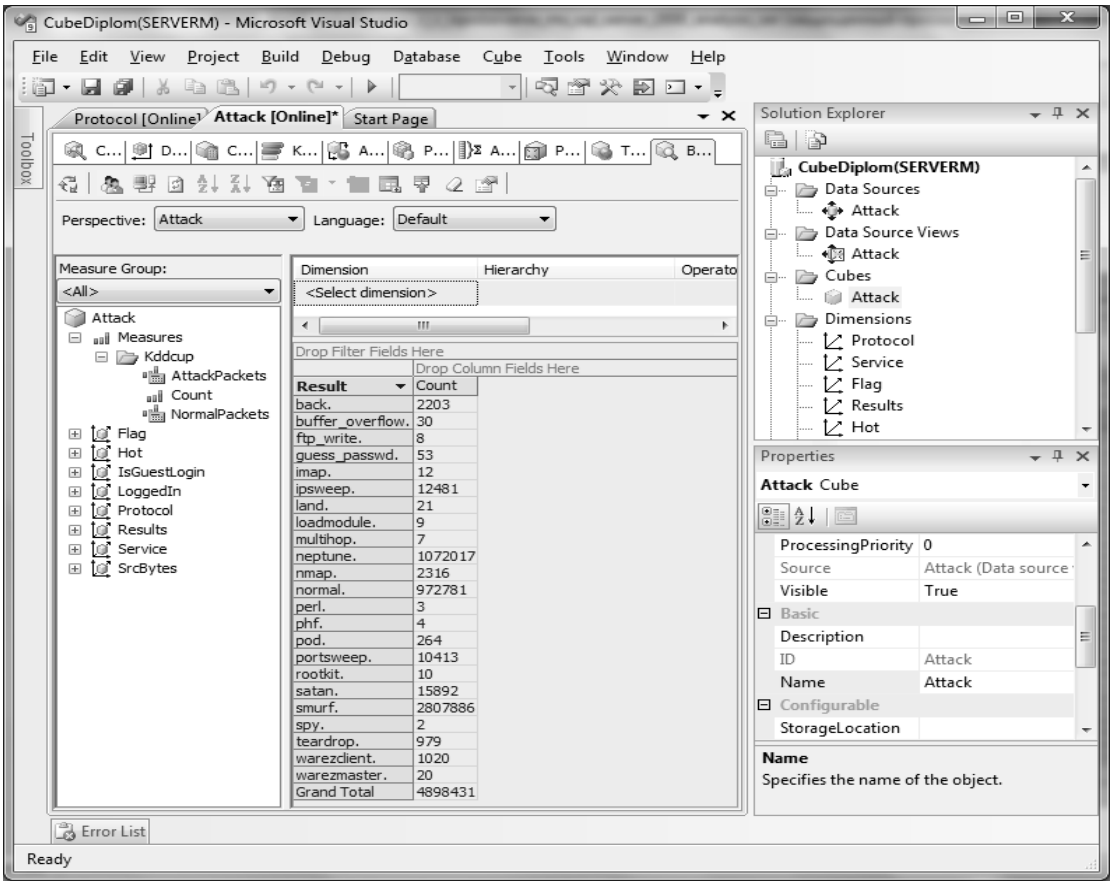


Рис. 2.11 – Конструктор кубів

2.2 Інтелектуальний аналіз вихідних даних

SQL Server Management Studio (SSMS) - інструмент, призначений для адміністраторів баз даних, що дозволяє управляти багатовимірними об'єктами,створеними розробниками баз даних. SSMS дозволяє адмініструвати Analysis Services, SQL Server, Reporting Services і Integration Services в єдиній консолі, яка об'єднує функціональність управління, редагування запитів і настроювання продуктивності.

За допомогою даного інструменту і *MDX* - мови запитів до багатовимірних даних, виділимо так звані сегменти – множини випадків з'єднань за символьними ознаками, які не перетинаються.

Лістинг запиту та його результати наведені на рисунку 2.12.

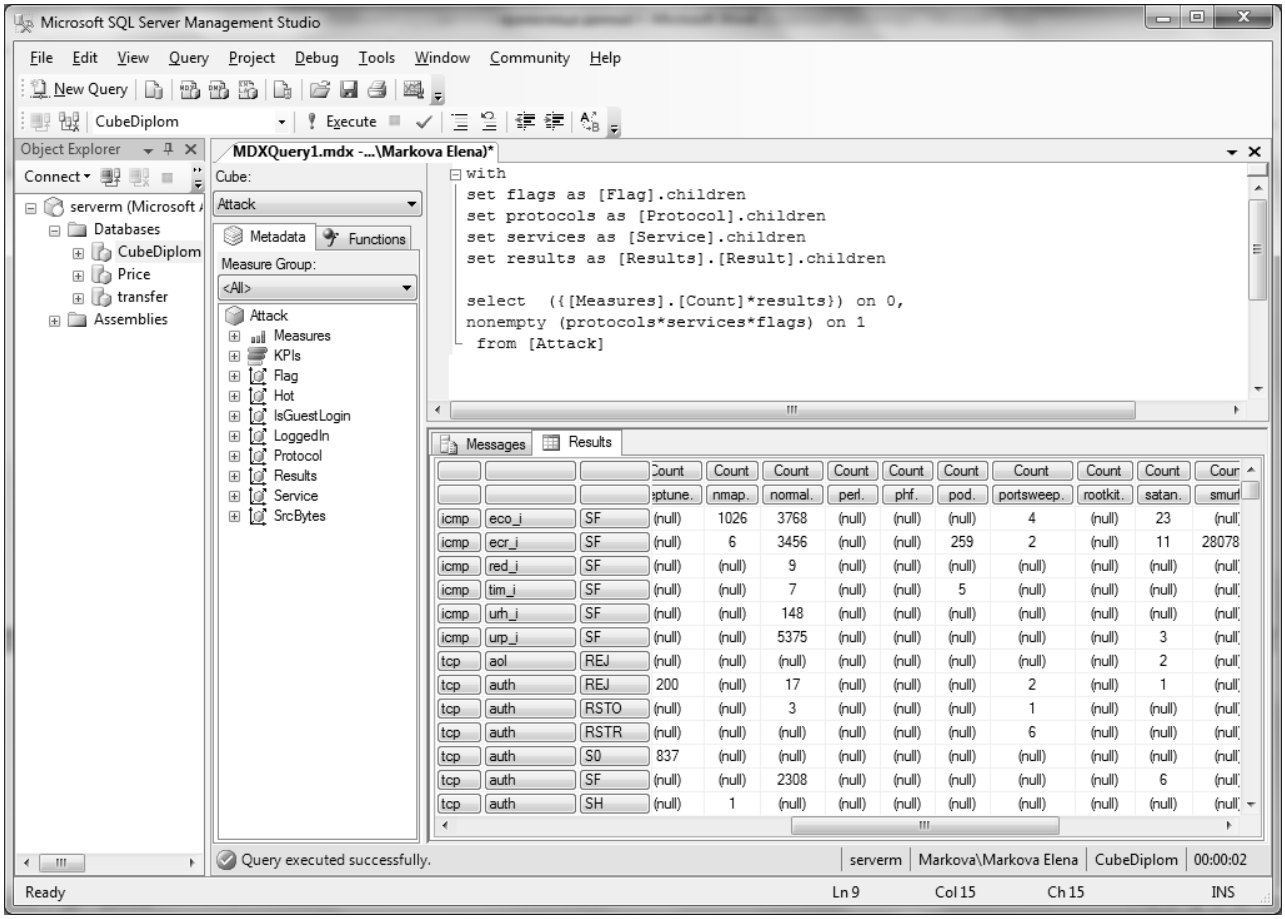


Рис. 2.12 – Результат виконання запиту

Таким чином ми виділили сегменти за допомогою перетину атрибутів Protocol, Service і Flag і отримали для кожного сегмента кількість з'єднань кож-

ного типу. Кількість унікальних значень параметру Protocol дорівнює 3, значень параметру Service – 70 та кількість значень параметру Flag – 11. Таким чином при перетині цих ознак ми маємо отримати 2310 сегментів.

Але деякі зі сполучень неможливі з точки зору комп'ютерних мереж, тож більшість кластерів виявляються порожніми, тобто в них не потрапляє жодного з'єднання. За допомогою функції nonempty мови MDX ми виділили тільки не порожні сегменти. Їх кількість 346.

Тепер серед цих сегментів виключимо ті, в які потрапляє лише один вид з'єднання, тобто це з'єднання можна ідентифікувати за допомогою правила. Таких сегментів 210. Наведемо деякі правила для ідентифікації таких з'єднань:

(Protocol = icmp) & (Service = urh_i) & (Flag = SF) → (Type =normal)

(Protocol = tcp) & (Service = auth) & (Flag = S0) → (Type =neptune)

(Protocol = tcp) & (Service = other) & (Flag = RSTR) → (Type =portsweep)

(Protocol = tcp) and (Service = private) & (Flag = SH) → (Type =nmap)

(Protocol = udp) & (Service = ntp_u) & (Flag = SF) → (Type = normal)

Фрагмент карти сегментів, які залишились для подальшого аналізу наведено на рис. 2.13.

A	B	C	D	E	F	G	H	I	J	K	L	M	N
Protocol	icmp	tcp	tcp	tcp	tcp	udp	udp	udp	tcp	tcp	icmp	tcp	icmp
Service	ecr_i	private	http	private	smtp	private	domain_u	other	http	ftp_data	eco_i	other	urp_i
Flag	SF	S0	SF	REJ	SF	SF	SF	SF	REJ	SF	SF	REJ	SF
back.	0	0	2105	0	0	0	0	0	0	0	0	0	0
buffer_overflow.	0	0	0	0	0	0	0	0	0	8	0	0	0
ftp_write.	0	0	0	0	0	0	0	0	0	4	0	0	0
guess_passwd.	0	0	0	0	0	0	0	0	0	0	0	0	0
imap.	0	0	0	0	0	0	0	0	0	0	0	0	0
ipsweep.	40	0	1	702	11	0	0	0	12	0	11517	0	0
land.	0	0	0	0	0	0	0	0	0	0	0	0	0
loadmodule.	0	0	0	0	0	0	0	0	0	3	0	0	0
multihop.	0	0	0	0	0	0	0	0	0	3	0	0	0
neptune.	0	819730	0	192790	1	0	0	0	200	0	0	0	0
nmap.	6	0	0	0	0	250	0	0	0	0	1026	0	0
normal.	3456	0	564647	1	95084	73848	57773	55891	53297	38000	3768	141	5375
perl.	0	0	0	0	0	0	0	0	0	0	0	0	0
phf.	0	0	4	0	0	0	0	0	0	0	0	0	0
pod.	259	0	0	0	0	0	0	0	0	0	0	0	0
portsweep.	2	150	0	2221	2	0	0	0	3	0	4	2	0
rootkit.	0	0	0	0	0	0	0	3	0	1	0	0	0
satan.	11	169	1	1532	13	1438	9	261	5	0	23	10627	3
smurf.	2807886	0	0	0	0	0	0	0	0	0	0	0	0
spy.	0	0	0	0	0	0	0	0	0	0	0	0	0
teardrop.	0	0	0	0	0	979	0	0	0	0	0	0	0
warezclient.	0	0	0	0	0	0	0	0	0	706	0	0	0
warezmaster.	0	0	0	0	0	0	0	0	0	18	0	0	0
Total	2811660	820049	566758	197246	95111	76515	57782	56155	53517	38743	16338	10770	5378

Рис. 2.13 – Фрагмент карти сегментів, що містять кілька типів з'єднань

Наприклад, якщо якесь з'єднання ми можемо однести до сегменту *[Protocol = icmp] & [Service = erc_i] & [Flag = SF],* то ми бачимо, що це з'єднання може бути нормальним (normal) або такими атаками: ipsweep, nmap, pod, portsweep, satan чи smurf.

Крім символьних ознак Protocol, Service і Flag, кожне з'єднання характеризується параметрами, які приймають дійсні числові значення. Ці числові ознаки і будемо використовувати для подальшої ідентифікації за допомогою обраних методів.

2.3 Побудова класифікаційної моделі ідентифікації небажаного вторгнення на основі логістичної регресії

Мультіномінальна логістична регресія є варіантом логістичної регресії, при якій залежна змінна не є дихотомічною, як при бінарній логістичній регресії, а має більше двох категорій. У той час як, при бінарній логістичній регресії незалежна змінна може мати інтервальну шкалу, то мультіномінальна логістична регресія придатна тільки для категоріальних незалежних змінних, причому має значення, чи належать вони до шкали найменувань або до порядкової шкали. Звичайно ж, не виключається можливість завдання як коваріат змінних, що мають інтервальну шкалу.

Якщо цільова змінна не є дихотомічною і всі предикторські змінні дискретні, то в цьому випадку ми маємо мультіномінальну таблицю випадків, що називається логлінійною моделлю. Логлінійна модель має 2 недоліки:

1. У неї набагато більше параметрів і багато з них не представляють інтересу. Логлінійна модель описує спільний розподіл всіх змінних, в той час як логістична модель описує тільки умовні розподіли цільової змінної по відношенню до предикторських.

2. Логлінійна модель набагато більш складна в інтерпретації. В логлінійній моделі, ефект предиктора *X* на цільову змінну *Y* описується як *XY*-асоціація. В логістичній моделі ефект *X* на *Y* абсолютний.

Для побудови мультіноміальної логістичної моделі будемо користуватись набором засобів візуалізації і бібліотекою алгоритмів машинного навчання для вирішення задач інтелектуального аналізу даних (data mining) Weka.

Припустимо є *k* класів для *n* випадків з *m* атрибутів, матриця параметрів B матиме розмірність *m*(k-1)*. Виберемо один із класів і назвемо його базовим, нехай для визначеності це буде останній клас *k*.

Ймовірність ідентифікації класу як класу *j* (за винятком базового класу) буде наступною

$$P_j\left(X_i\right)=\frac{exp(X_iB_j)}{\sum_{l=1}^{k-1}exp(X_iB_l)+1} \tag{2.1}$$

Ймовірність базового класу

$$P_k\left(X_i\right)=1-\sum_{j=1}^{k-1}P_j\left(X_i\right)=\frac{1}{\sum_{j=1}^{k-1}exp(X_iB_j)+1} \tag{2.2}$$

Від'ємна мультіноміальна функція правдоподібності буде наступною:

$$L=-\sum_{i=1..n}\sum_{j=1..(k-1)}Y_{ij}\ln\left(P_j\left(X_i\right)\right)+\left(1-\sum_{j=1..(k-1)}Y_{ij}\right)\times\ln\left(1-\sum_{j=1..(k-1)}P_j\left(X_i\right)\right)+ridge\times(B^2) \tag{2.3}$$

де *ridge* - деякий коефіцієнт, який вибирається перед побудовою моделі.

Задача зводиться до знаходження матриці *B*, для якої *L* мінімальна. Для пошуку оптимальних значень *m×(k-1)* змінних використовується квазі-ньютонівський метод, що закладений у програму.

Хоча оригінальна логістична регресія не має справу з вагами випадків, цей алгоритм трохи змінений, аби мати можливість поводитись з вагами випадків.

Зауважимо, що в WEKA є можливість замінювати пропущені значення, а номінальні ознаки можуть бути перетворені в числа.

Запускаємо інструмент для аналізу Weka.

Рис. 2.14 – Інтерфейс Weka

Через Explorer\Панель передобробки (Preprocess panel) імпортуємо дані з .txt файлу. Зауважимо, що текстові файли приготовлені заздалегідь для кожного сегмента. У кожному файлі знаходяться спостереження, які відносяться до одного певного сегменту. Спостереження в різних файлах не перетинаються. Наприклад, імпортуємо дані, які належать до сегменту [Protocol = icmp] та [Service = erc_i] та [Flag = SF].

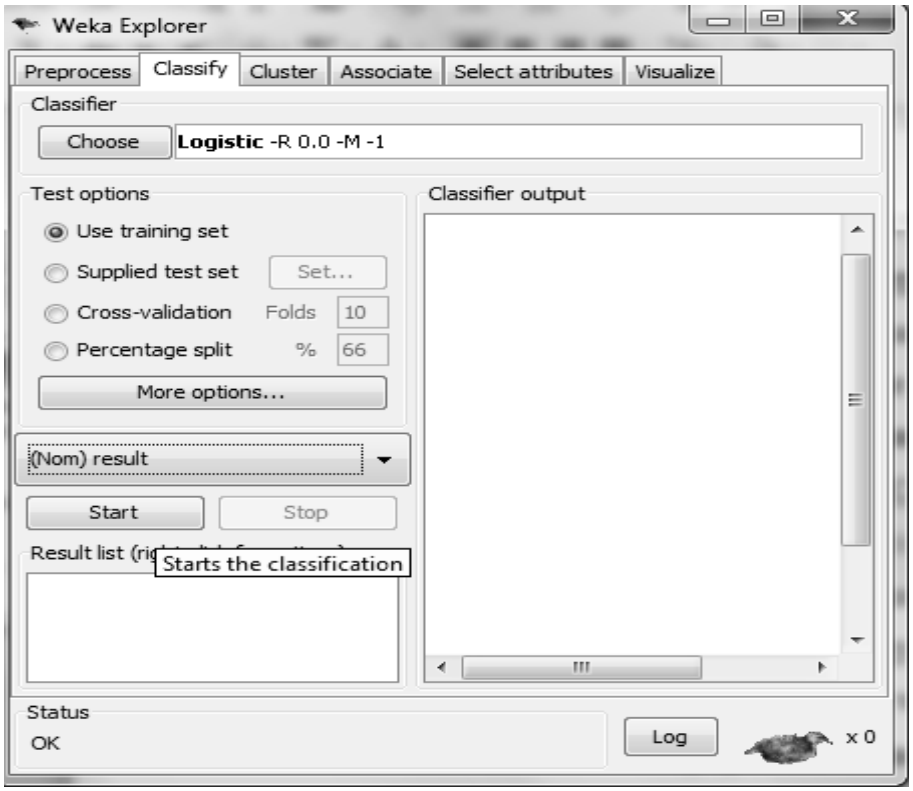


Рис. 2.15 – Панель класифікації

На панелі класифікації (Classify panel) обираємо необхідний метод, його параметри та ознаку, яка буде задавати клас (у нашому випадку це result). Запускаємо процес класифікації. У результаті навчання класифікатора отримали таку модель:

Спостереження: 2811660

Атрибути: 17

```
result
src_bytes
wrong_fragment
count
srv_count
serror_rate
same_srv_rate
diff_srv_rate
srv_diff_host_rate
dst_host_count
dst_host_srv_count
dst_host_same_srv_rate
dst_host_diff_srv_rate
dst_host_same_src_port_rate
dst_host_srv_diff_host_rate
dst_host_serror_rate
dst_host_rerror_rate
```

Логістична регресія з параметром *ridge = 0*

Таблиця 2.5

Коефіцієнти логістичної моделі

Variable	normal.	smurf.	pod.	nmap.	satan.	ipsweep.
src_bytes	4.2197	4.4525	1.2638	-0.8574	2.714	0.2514
wrong_fragment	-1149.7	-1309.8	5920.456	-8635.98	-2157.99	3634.265
count	-12.9346	2.2268	0.297	-2.9483	0.2687	-0.9039
srv_count	3.3873	-7.1429	-1.183	-2.3313	-3.3719	-3.2684
serror_rate	1436.964	-1750.08	-3044.15	803.3014	-1917.55	728.1108
same_srv_rate	-1436.94	1749.979	3044.149	-803.302	1917.551	-728.111
diff_srv_rate	23709.28	-28876.3	-50228.5	13254.49	-31639.6	12013.83
srv_diff_host_rate	-3259.2	-4469.64	1088.645	3219.767	-3520.05	812.5953
dst_host_count	-15.3112	-14.5841	-16.4187	9.1307	-14.0866	67.1201
dst_host_srv_count	21.1235	20.0058	20.6337	-3.1063	18.8708	-61.455
dst_host_same_srv_rate	-11827.5	37815.51	-12005.9	842.6783	-12169.8	-744.959
dst_host_diff_srv_rate	-1639.49	-1219.31	-1229.74	4361.927	77.2599	15512.11
dst_host_same_src_port_rate	6121.806	-43256	5258.329	679.7233	6774.016	26349.57
dst_host_srv_diff_host_rate	102319.6	96446.01	103549.5	64829.68	-37687.4	-316148
dst_host_serror_rate	-2416.68	-2371.57	-3053.81	-1228.54	-2768.2	-989.851
dst_host_rerror_rate	-465.844	-939.26	-1557.35	-1946.55	-12728.1	8696.59
Intercept	7067.173	3591.517	2387.237	-713.295	3428.674	-24879.2

Час, необхідний для створення моделі:13201,57 секунд.

Оцінка на навчальній виборці:

Правильно класифіковані об’єкти: 2811652 99,9997 %

Неправильно класифіковані об’єкти: 8 0,0003 %

Надійність моделі: 0,9989.

Середня абсолютна похибка: $4{,}7*10^{-7}$.

Корінь із середньої квадратичної похибки: 0,0007.

Загальна кількість об’єктів (спостережень): 2811660.

Таблиця 2.6

Результати ROC- аналізу по класах

	TRUE	FALSE	Точність	Чутливість	F-статистика	Площа ROC- кривої	Клас
	1	0	1	1	1	1	normal.
	1	0	1	1	1	1	smurf.
	1	0	1	1	1	1	pod.
	0.333	0	0.667	0.333	0.444	1	nmap.
	1	0	0.917	1	0.957	1	satan.
	0.975	0	0.867	0.975	0.918	1	ipsweep.
	0	1	0	0	0	1	portsweep.
Зважене середнє	1	0	1	1	1	1	

Таблиця 2.7

Матриця зв’язаності

normal	smurf	pod	nmap	satan	ipsweep	portsweep	← реальний тип	↓ розпізнано
3455	0	0	0	1	0	0	|	normal
0	2807886	0	0	0	0	0	|	smurf
0	0	259	0	0	0	0	|	pod
0	0	0	2	0	4	0	|	nmap
0	0	0	0	11	0	0	|	satan
0	0	0	1	0	39	0	|	ipsweep
0	0	0	0	0	2	0	|	portsweep

Тепер аналогічно отримаємо модель для іншого сегменту, наприклад, *[Protocol = udp] & [Service = private] & [Flag = SF]*.

Спостереження: 76515

Атрибути: 18

```
result
Duration
src_bytes
dst_bytes
wrong_fragment
count
srv_count
serror_rate
rerror_rate
same_srv_rate
diff_srv_rate
dst_host_count
dst_host_srv_count
dst_host_same_srv_rate
dst_host_diff_srv_rate
dst_host_same_src_port_rate
dst_host_serror_rate
dst_host_rerror_rate
```

Логістична регресія з параметром *ridge = 0*

Таблиця 2.8

Коефіцієнти логістичної моделі

Variable	teardrop.	satan.	nmap.
Duration	-0.037	-0.057	-7.5355
src_bytes	-1.3688	-1.7571	0.3027
dst_bytes	0.1707	-0.061	-0.0012
wrong_fragment	39.0768	-68.3356	-96.2403
count	-1.2138	-0.2605	-11.351
srv_count	1.5563	2.7395	13.5394
serror_rate	434.6762	217.9543	360.2853
rerror_rate	5299.424	-2400.73	-13265.3
same_srv_rate	27.6604	18.775	536.3654
diff_srv_rate	-194.923	8.5318	236.0322
dst_host_count	0.2742	0.1596	0.0268
dst_host_srv_count	0.1488	0.257	-0.1684
dst_host_same_srv_rate	-35.9989	12.8245	-146.874
dst_host_diff_srv_rate	18.8328	53.3302	-47.8181
dst_host_same_src_port_rate	-4.9131	81.0726	136.3385
dst_host_serror_rate	64.5436	118.0239	-572.939
dst_host_rerror_rate	-24.1633	36.8448	-136.388
Intercept	-4.2086	18.7301	-564.935

Час, необхідний для створення моделі: 67,09 секунд.

Оцінка на навчальній виборці:

Правильно класифіковані об’єкти: 76507 99,9895 %

Неправильно класифіковані об’єкти: 8 0,0105 %

Надійність моделі: 0,9985

Середня абсолютна похибка: 0,0001

Корінь із середньоквадратичної похибки: 0,0062

Загальна кількість об’єктів (спостережень): 76515

Таблиця 2.9

Детальна точність по класах

	TRUE	FALSE	Точність	Чутливість	F-статистика	Площа ROC-квивої	Клас
	1	0	1	1	1	1	teardrop.
	1	0	1	1	1	1	satan.
	0.996	0	0.973	0.996	0.984	1	nmap.
	1	0	1	1	1	1	normal.
Зважене середнє	1	0	1	1	1	1	

Таблиця 2.10

Матриця зв’язаності

teardrop	satan	nmap	normal	←розпізнано	↓ реальний тип
979	0	0	0	|	teardrop
0	1438	0	0	|	satan
0	0	249	1	|	nmap
0	0	7	73841	|	normal

Аналізуючи результати навчання моделей на визначених попередньо кластерах, наведені у таблицях 2.5 – 2.10, можна зробити наступні висновки:

– коефіцієнти мультіномінальної багатовимірної регресії приймають доволі великі значення, тобто отримані моделі чутливі до зміни вхідних даних;

– для кожного з кластерів можна отримати математичну модель не більше як з 24 параметрів, яка з надійністю не менше 0,99 дозволяє вірно ідентифікувати вид з’єднання, в тому числі відрізнити нормальні з’єднання від небажаних;

– чим більше об’єктів попадає в кластер, тим більше часу потрібно для побудови моделі й визначення її коефіцієнтів, втім цей час витрачається один раз і в майбутньому за вже налаштованою моделлю можна ідентифікувати вхідне з’єднання в реальному масштабі часу;

– якість роботи моделі на прикладах (кількість помилок) не залежить від кількості прикладів даного типу в кластері, тобто модель однаково якісно розпізнає й атаку, якої у даному кластері майже 3 мільйони прикладів, й атаки

яких лише десятки прикладів.

Останнє спостереження дає можливість стверджувати, що запропонована в роботі методика ідентифікації мережевих з’єднань, на відміну від методів, побудованих на статистичних оцінках, є стійкою і нечутливою до тиску на коефіцієнти прикладів певного типу, кількість яких значно більша, ніж в інших.

2.4 Класифікація видів мережевих вторгнень з використанням мережі Кохонена

Карти, які самоорганізуються, представляють собою потужний аналітичний інструмент, який об'єднує в собі дві основні парадигми аналізу - кластеризацію і проеціювання, тобто візуалізацію багатовимірних даних на площині. Мережа Кохонена розпізнає кластери в багатовимірних навчальних даних і відносить усі дані до тих чи інших кластерів, використовуючи алгоритм проеціювання зі збереженням топологічної подоби. При цьому ті елементи вибірки, які перебувають у відносній близькості у вихідному багатомірному просторі, виявляються поруч і в просторі з більш низькою розмірністю.

Для того, аби наглядно показати, як працюють карти, обираємо сегмент [Protocol = icmp] та [Service = eco_i] та [Flag = SF] та імпортуємо дані, які відносяться до цього сегменту.

Результати роботи наведені на рисунку 2.16.

Рис. 2.16 – Імпорт необхідних даних

Проаналізувавши карти в проекції на різні компоненти, візуально можемо виділити, що для даного сегменту інформативними (варіабельними) є лише такі ознаки:

```
hot
num_failed_logins
logged_in
count
srv_count
serror_rate
rerror_rate
same_srv_rate
diff_srv_rate
dst_host_count
dst_host_srv_count
dst_host_same_srv_rate
dst_host_diff_srv_rate
dst_host_same_src_port_rate
dst_host_srv_diff_host_rate
dst_host_serror_rate
dst_host_srv_serror_rate
dst_host_rerror_rate
dst_host_srv_rerror_rate.
```

Проекції карт ілюстровані на рис. 2.17 – 2.21.

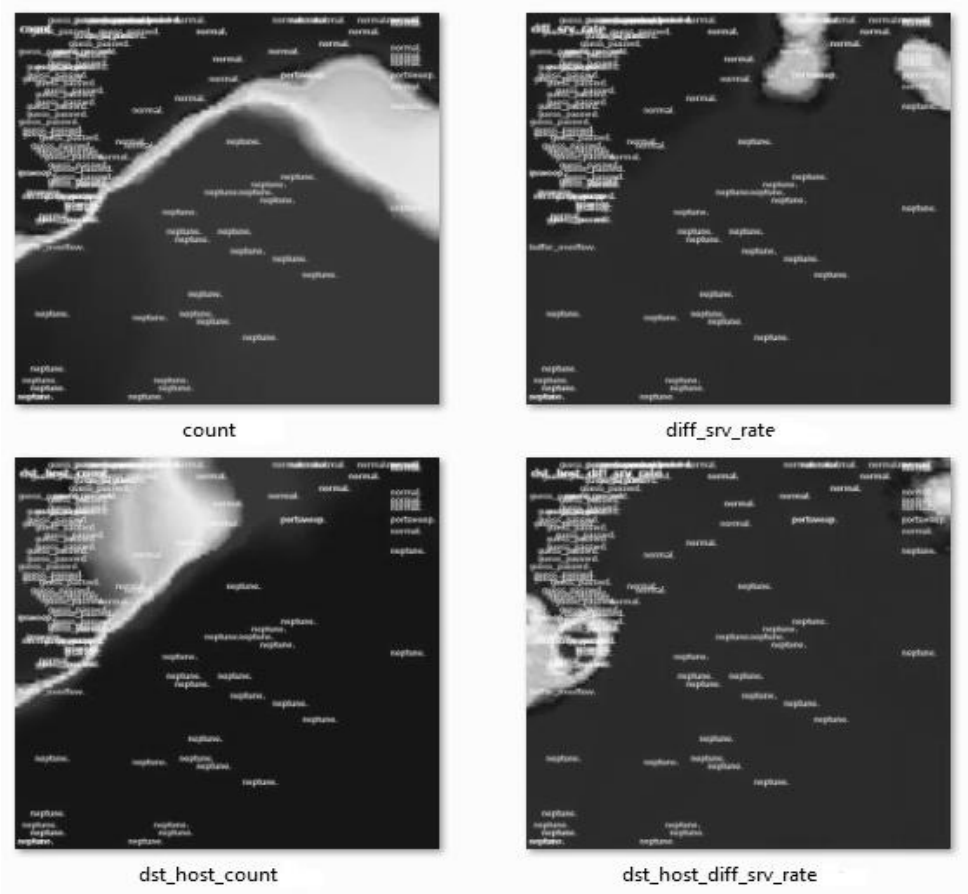


Рис. 2.17 – Проекції карт для ознак count, diff_srv_rate, dst_host_count, dst_host_diff_srv_rate

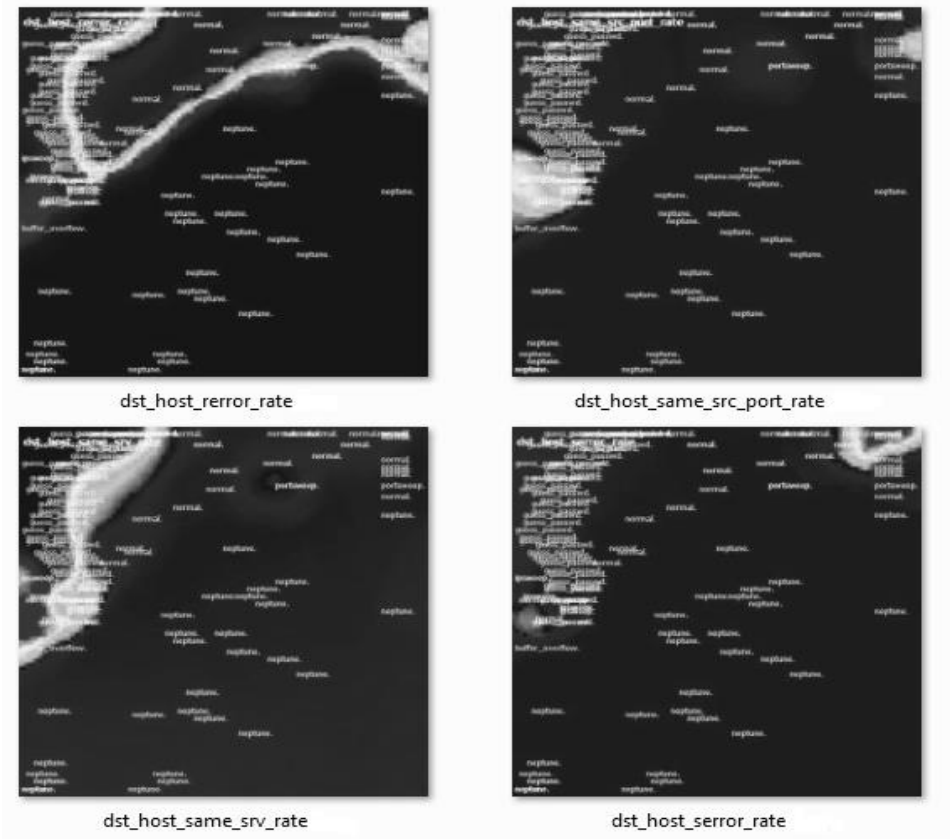


Рис. 2.18 – Проекції карт для ознак dst_host_rerror_rate, dst_host_same_src_port_rate, dst_host_same_srv_rate, dst_host_serror_rate

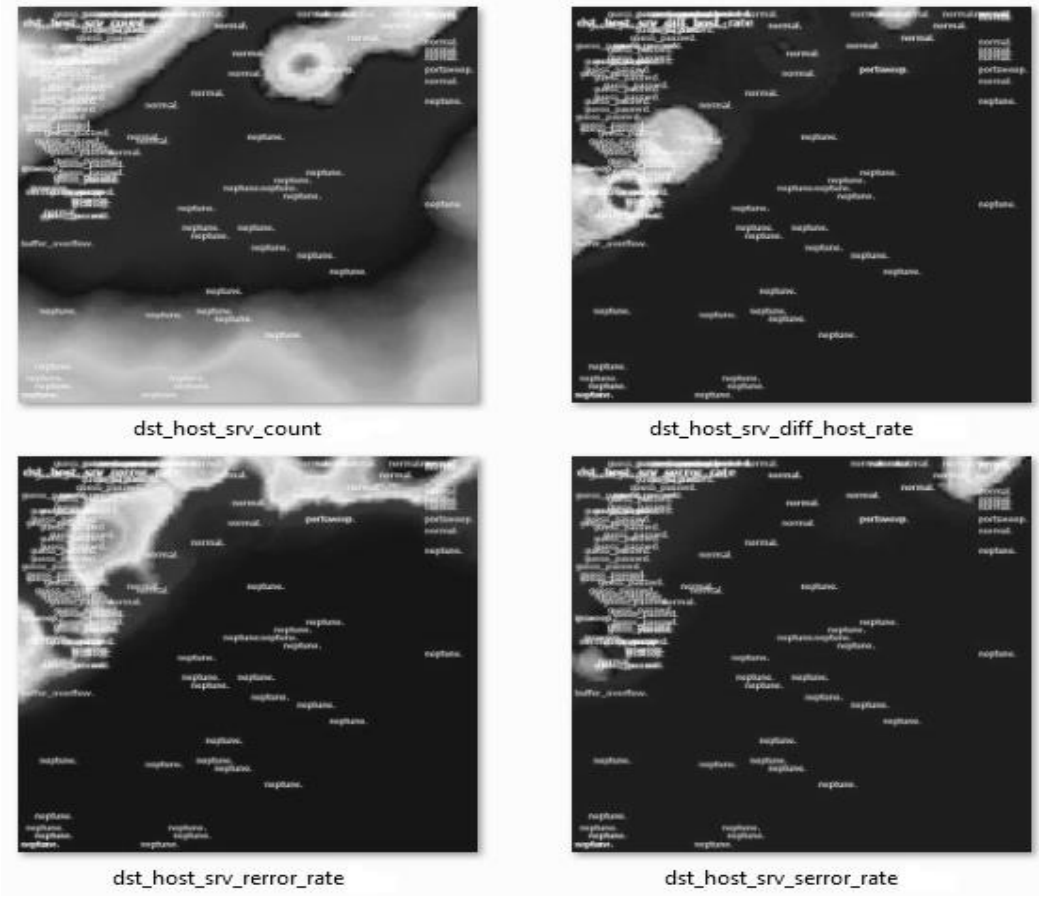


Рис. 2.19 – Проекції карт для ознак dst_host_srv_count, dst_host_srv_diff_host_rate, dst_host_srv_rerror_rate, dst_host_srv_serror_rate

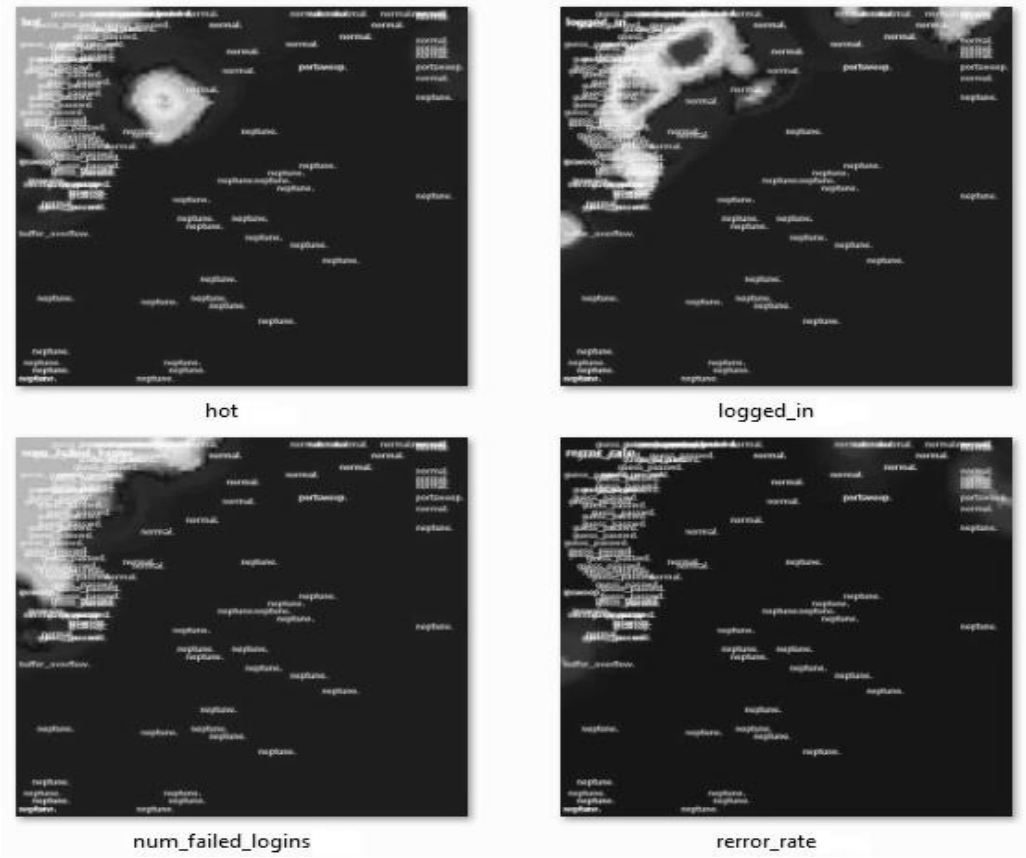


Рис. 2.20 – Проекції карт для ознак hot, logged_in, num_failed_logins, rerror_rate

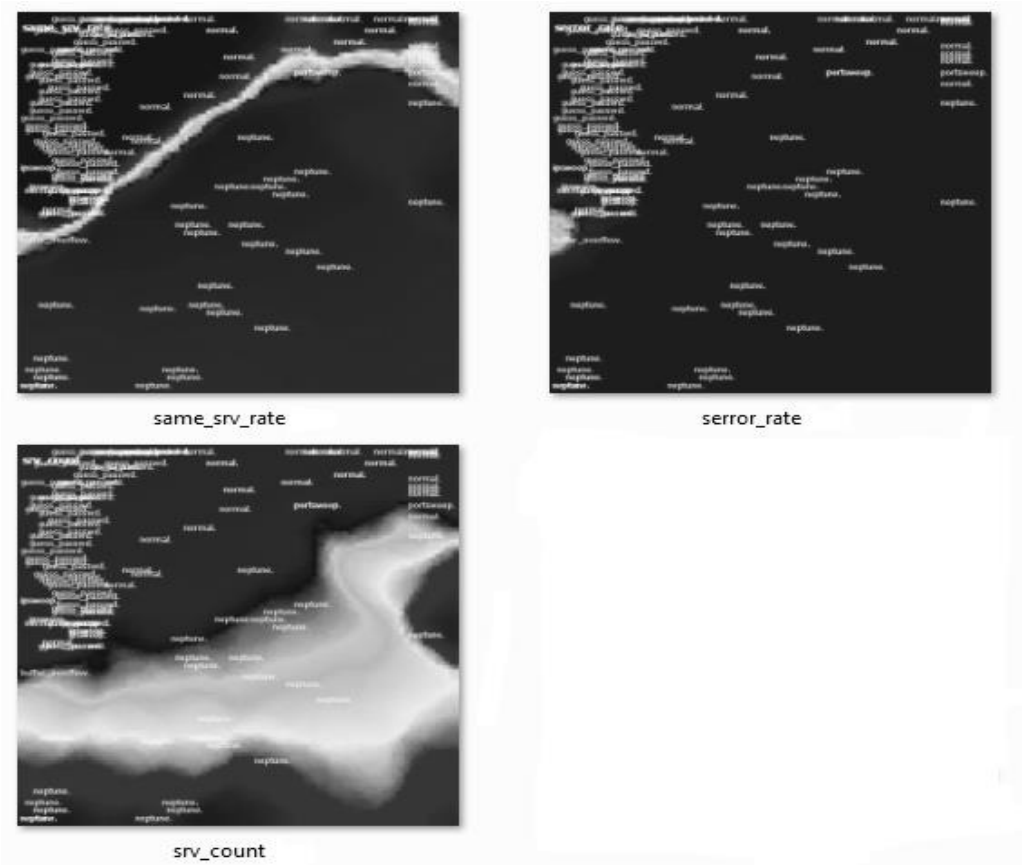


Рис. 2.21 – Проекції карт для ознак same_srv_rate, serror rate, srv_count

Карта, що описує уніфіковану матрицю відстаней між кожним нейроном і його найближчими сусідами проілюстрована на рис. 2.22.

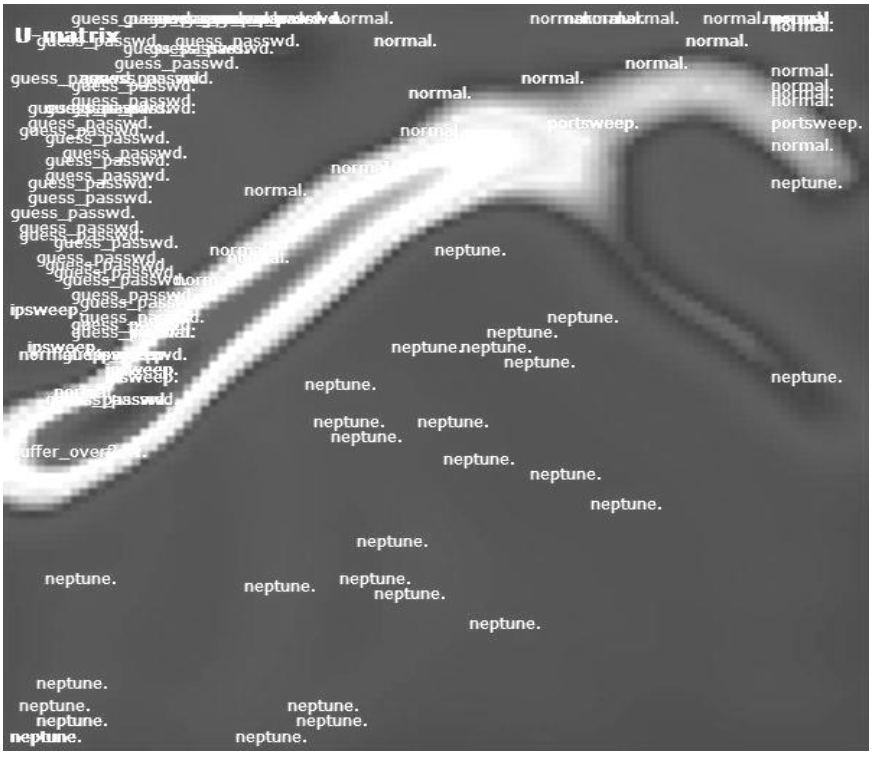


Рис. 2.22 – Карта, що описує U-матрицю.

Вузлам, які різко контрастують зі своєю околицею, відповідає чорний колір, а ділянкам, що носять характер "згладженого плато", - білий. Групу клітинок, відстань між якими всередині цієї групи менше, ніж відстань до сусідніх груп, визначимо як кластер.

Таким чином, застосування інтелектуальних карт Кохонена, що самонавчаються, дозволяє суттєво зменшити простір пошуку рішення, зокрема, при побудові рівняння багатоваріантної логістичної регресії.

В подальшому, удосконалений алгоритм ідентифікації небажаних вторгнень у локальну корпоративну мережу має включати проміжний етап побудови карт Кохонена, за допомогою яких відбувалася б сепарація ознак з'єднань на ті параметри, що істотно впливають і мають враховуватись в моделі й ті параметри, які є не суттєвими й можуть бути відкинутими.

2.6 Експериментальні дослідження запропонованої методики

Аби впевнитися, що запропонований в роботі удосконалений алгоритм ідентифікації небажаних вторгнень у локальну комп'ютерну мережу ефективний і незалежний від факторів навчальної вибірки, необхідно виконати його перевірку на даних, що не використовувались у навчанні.

До тестової вибірки включено 2 106 200 записів, знятих керуючою компанією MIT Lincoln Labs за звітами системних адміністраторів підприємства протягом двох тижнів 2009 року. Час зняття даних тестової вибірки не збігається з часом зняття даних навчальної.

Обраний метод тестування є традиційним для даного класу задач [23]. Класифікатори зазвичай тренуються на наявному масиві вхідних даних, після чого застосовуються для аналізу нових вхідних даних. В результаті аналізу приймається рішення про приналежність нових даних до одного з відомих класів (кластерів).

Основні показники точності класифікації - ймовірність правильного визначення приналежності до заданого класу (Detection Rate - DR) та ймовірність помилкового визначення приналежності до заданого класу (False Alarm Rate - FAR). Також в разі появи у даних невідомих раніше класів об’єктів, слід очікувати відмови системи, тобто неможливості ідентифікувати приклад (No Response Rate – NRR). Для більш детальної оцінки якості класифікації застосовується матриця помилок (confusion matrix).

Важливо зазначити, що тестові дані розподілені не з тією ж імовірністю, що й тренувальні дані, і включають в себе певні спеціальні типи атак, які не містяться в тренувальних даних. Це робить задачу більш реалістичною. Деякі експерти з вторгнень вважають, що більшість нових атак є різновидами вже відомих атак, і знання "почерку" відомих атак може бути достатньо, щоб вловити їх нові різновиди.

Після завантаження тестові дані були поділені на кластери, за тим самим принципом, що й навчальні, а саме за сполученнями категорійних змінних protocol, flag, service та типу атак result. На відміну від навчальної вибірки, у тестовій знайдено не 346, а 378 непорожніх кластерів. Вочевидь, вони відповідають новим, раніше невідомим типам атак та містять їх перетини з раніше відомими класами.

Всі приклади, що входять у нові 378 – 346 = 32 кластери, згідно із запропонованою методикою, будуть нерозпізнані. Сукупна кількість прикладів, що припадає на нові типи кластери – 17 894. Таким чином, розраховуємо першу характеристику системи, а саме кількість «відмов» - відсоток даних, що з урахуванням гіпотези що тестова вибірка відображає генеральну сукупність, не будуть розпізнані системою взагалі:

*NRR = 17 894 / 2 106 200 * 100% = 0,85%.*

Для аналізу якості ідентифікації невідомих прикладів звернімося до тих самих кластерів, що розглядалися нами раніше.

Для кластеру *[Protocol = icmp] & [Service = erc_i] & [Flag = SF]*, навчання якого відображено в таблицях 2.5 – 2.7, маємо наступні результати:

Оцінка на тестовій виборці 1:

Правильно класифіковані об’єкти: 1 643 468 99,9968 %

Неправильно класифіковані об’єкти: 52 0,0032 %

Надійність моделі: 0,9969

Середня абсолютна похибка: 6,1*10-7.

Корінь із середньоквадратичної похибки: 0,00078

Загальна кількість об’єктів (спостережень): 1 643 250.

Таблиця 2.11

Результати ROC–аналізу по класах

	TRUE	FALSE	Точність	Чутливість	F-статистика	Площа ROC-кривої	Клас
	0,999	0	0,980	0,999	0,989	1	normal.
	1	0	1	1	1	1	smurf.
	0,956	0	0,999	0,956	0,977	1	pod.
	0	1	0	0	0	1	nmap.
	0	1	0	0	0	1	satan.
	0,952	0	0,983	0,952	0,967	1	ipsweep.
	0	1	0	0	0	1	saint
Зважене середнє	1	0	1	1	1	1	

Таблиця 2.12

Матриця зв’язаності

normal	smurf	pod	nmap	satan	ipsweep	saint	← розпізнано	↓ реальний тип
1726	0	1	0	0	1	0	|	normal
4	1640905	0	0	0	0	0	|	smurf
32	0	778	2	2	0	0	|	pod
0	0	0	0	0	0	0	|	nmap
0	0	0	0	0	0	0	|	satan
0	0	0	1	2	59	0	|	ipsweep
0	0	0	1	6	0	0	|	saint

Слід звернути увагу, що у тестовій виборці для цього кластеру не було жодного представника об’єктів з класів *nmap* і *satan*, натомість з’явилася незначна кількість нових небажаних вторгнень, що ідентифікуються як *saint*. Загальна структура кластера залишилася незмінною – більше 99% з’єднань такого типу припадають на *smurf*, решта розділяються між нормальними з’єднаннями та атаками *pod* й *ipsweep*.

Як видно з таблиць 2.11 та 2.12, запропонований алгоритм цілком вдало впорався із завданням ідентифікації, помилившись у 52 випадках з 1 643 250. Надійність системи близька до 1, хоча більшість помилок – ідентифікація потенційно небезпечних з'єднань як нормальних.

Щодо нового типу атак *saint*, всі вони були ідентифіковані як небезпеки й були розкидані по відомим класам. Не дивно. Що більшість (6 з 7) випадків атак *saint* була ідентифікована класом *satan*, адже згідно з [34] *saint* – вірус-нащадок, програми *satan* і виконує аналогічні операції.

Тепер аналогічно отримаємо модель для іншого р з розглянутих сегментів, а саме *[Protocol = udp] & [Service = private] & [Flag = SF]*. У порівнянні з навчальною вибіркою, ньому з'являються одразу два типи небажаних вторгнень: вищезгаданий *saint* та один з новітніх методів захоплення ресурсів сервера *updstorm*.

Результати роботи програми WEKA відображають таблиці 2.13 та 2.14.

Оцінка на тестовій виборці 2:

Правильно класифіковані об'єкти: 130 133 56,1533 %

Неправильно класифіковані об'єкти: 101 613 43,8467 %

Надійність моделі: 0,4367.

Середня абсолютна похибка: 4,3*10-6.

Корінь із середньоквадратичної похибки: 0,00208

Загальна кількість об'єктів (спостережень): 231 746.

Таблиця 2.13

Результати ROC–аналізу по класах

	TRUE	FALSE	Точність	Чутливість	F-статистика	Площа ROC-кривої	Клас
	0,992	0	1	0,992	0,996	1	teardrop.
	1	0	0,904	1	1	1	satan.
	0	1	0	0	0	1	nmap.
	1	0	0,998	1	0,999	1	normal.
	0	1	0	0	0	1	saint.
	0	1	0	0	0	1	smurf.
	0	1	0	0	0	1	udpstorm
Зважене середнє	0,564	0,436	0,562	0,564	0,563	1	

Таблиця 2.14

Матриця зв’язаності

teardrop.	satan.	nmap.	normal.	saint.	smurf.	udpstorm	← розпізнаний клас	↓ реальний клас
123	0	1	0	0	0	0	|	teardrop.
0	2077	0	0	0	0	0	|	satan.
0	0	0	0	0	0	0	|	nmap.
0	0	21	127933	0	7	0	|	normal.
0	221	3	0	0	0	0	|	saint.
0	0	0	0	0	0	0	|	smurf.
0	0	42	296	0	100022	0	|	udpstorm

На перший погляд, отримані для другого кластеру результати викликають розчарування, адже надійність моделі менша за 0,5, а на головній діагоналі більшість клітин займають нулі. Однак, звернімо увагу, що один з нових класів, а саме *udpstorm* є атакою, спрямованою на вичерпання ресурсів сервера локальної мережі й має на меті відмову в обслуговуванні[34]. Інакше кажучи, його без сумніву можна об’єднати зі знайомим вже програмі класом *smurf*. На загальних рисах атак класів *satan* та *saint* ми вже зупинялися вище. Для об’єктивності, переглянемо результати розрахунків, об’єднавши класи-аналоги.

Оцінка на тестовій виборці 2_2:

Правильно класифіковані об’єкти: 230 376 99,4088 %

Неправильно класифіковані об’єкти: 1 370 0,5912 %

Надійність моделі: 0,9984.

Середня абсолютна похибка: $4{,}3*10^{-6}$.

Корінь із середньоквадратичної похибки: 0,00208

Загальна кількість об’єктів (спостережень): 231 746.

Таблиця 2.15

Результати ROC–аналізу по класах

	TRUE	FALSE	Точність	Чутливість	F-статистика	Площа ROC-кривої	Клас
	0,992	0	1	0,992	0,996	1	teardrop.
	0,999	0	1	0,999	1	1	satan+saint
	0	1	0	0	0	1	nmap.
	1	0	0,998	1,000	0,999	1	normal.
	0,997	0	1	0,997	0,998	1	smurf+ udpstorm
Зважене середнє	0,9984	0	0,9987	0,9984	0,9985	1	

Таблиця 2.16

Матриця зв’язаності

teardrop	satan+saint	nmap	normal	smurf+udpstorm	← розпізнаний клас	↓ реальний клас
123	0	1	0	0	|	teardrop.
0	2298	3	0	0	|	satan+saint
0	0	0	0	0	|	nmap.
0	0	21	127933	7	|	normal.
0	0	42	296	100022	|	smurf+udpstorm

Отриманий результат свідчить, що навіть при виникненні нових типів загроз, що мають свій аналог серед заздалегідь відомих класів, навчена на прикладах система вдало ідентифікує небезпеки, співставляючи їх з відомим типом. Відсоток атак, що ідентифікуються як нормальні з’єднання, залишається доволі високим.

Наведені приклади є окремими випадками перевірки роботи запропонованого удосконаленого алгоритму та системи підтримки прийняття рішень, що його використовує, на тестових даних. Загалом були отримані наступні результати:

Обробка тестової вибірки:

Правильно класифіковані об’єкти: 2 045 622; DR = 97,1238 %

Неправильно класифіковані об’єкти: 42 684; FAR = 2,0266 %

Некласифіковані об’єкти: 17 894; NRR = 0,8496%

Надійність моделі (ідентифікація класу): 0,9679.

Надійність моделі (ідентифікація нормальних з’єднань): 0,9908.

Загальна кількість об’єктів (спостережень): 2 106 200.

Показана в тестах висока точність та надійність запропонованої системи ідентифікації небажаних вторгнень свідчить про правильність обраного підходу до вирішення задачі, та розробленого алгоритму, який вдало вирішує задачу ідентифікації як відомих загроз, так і таких, що ще не зустрічалися.

Особливо слід відзначити надійність класифікації з’єднань на нормальні та небажані – якщо відкинути додаткову задачу класифікації типу вторгнення, удосконалений алгоритм ідентифікації має надійність вищу за 0,99.

2.7 Висновки до розділу

В ході аналізу таблиці початкових даних було проаналізовано кожен з параметрів, що описують класи з'єднань у комп'ютерній мережі, визначено їх типи та діапазони зміни. Вхідним даним були призначені виміри багатовимірного сховища агрегованих даних (гіперкуба) та створений гіперкуб.

Було визначено, що крім числових дійсних ключову роль у класифікації типів з'єднань комп'ютерної мережі грають три категорійні (символьні) параметри: протокол, за яким встановлюється з'єднання, служба цього протоколу, яка використовується та прапорець ознак. Згадані параметри, а також тип з'єднання є опорними вимірами кубу даних.

Простір даних навчальної вибірки було кластеризовано за перетином категорійних параметрів. Виявилося, що в навчальній виборці є 364 не порожніх кластерів, кожен з яких містить елементи одного або більше класів. Серед інших були визначені 210 кластерів, які відповідають одному єдиному типу з'єднання. Кожному такому кластеру для ідентифікації достатньо записати правило вигляду

$$(Protocol = id1) \,\&\, (Service = id2) \,\&\, (Flag = id3) \rightarrow (Type = id\_klass)$$

де *id1, id2* та *id3* – ідентифікатори відповідно протоколу з'єднання, служби та прапорця, а *id_klass* – ідентифікатор класу з'єднань.

Втім, левова частка прикладів з навчальної виборки не попадає у «чисті» класи, тому для подальшої ідентифікації було застосовано механізм багатоваріантної логістичної регресії. Для кожного з 136 решти кластерів було знайдено рівняння, яке дозволяє класифікувати приклади з'єднань на один з припустимих класів.

Для відбору значимих параметрів, що включалися до моделі багатоваріантної логістичної регресії, застосовувалися карти Кохонена – нейронні мережі з самонавчанням, що дозволяють візуально визначити й виключити з подальшого

розгляду виміри, які не вносять в результат суттєвої інформації (не є класифікуючими).

Як показала експериментальна перевірка на навчальній виборці, система підтримки прийняття рішень з удосконаленим алгоритмом ідентифікації вторгнень демонструє надійність, близьку до 0,999, припускаючись в залежності від кластера від 0,0003% до 0,03% помилок.

Тестування на виборці, що містила 14 невідомих раніше типів атак та мала інше співвідношення між видами з'єднань, показала, що надійність системи при ідентифікації класу загрози, або найближчої до неї за типом становить не менше 0,968, а при розділенні з'єднань на нормальні й небезпечні – не менше ніж 0,99. Така висока точність ідентифікації свідчить про правильність обраного підходу до вирішення задачі, та розробленого алгоритму, який вдало вирішує задачу ідентифікації як відомих загроз, так і таких, що ще не зустрічалися.

Розроблена система підтримки прийняття рішень та удосконалений алгоритм ідентифікації небажаного вторгнення в локальну мережу підприємства в умовах недостатньої інформації із застосуванням технологій OLAP можуть бути запропоновані фахівцям з безпеки комп'ютерних мереж та системним адміністраторам корпоративних локальних комп'ютерних мереж.

ВИСНОВКИ

Задача ідентифікації (процесів, систем) або побудова математичної моделі за результатами спостережень посідає одне з головних місць у сучасній теорії управління і прийнятті рішень у різних сферах: техніці, економіці, біології та в ін. Розв'язання задачі ідентифікації є обов'язковим етапом для наступного прийняття рішень або формування керуючого впливу.

Останнім часом стрімко поширюються зловмисні мережі, підвищується інтелектуальність мережевих атак, зростає організована кіберзлочинність та шпигунство з використанням всесвітньої мережі Інтернет, а також з використанням зростаючої мережі мобільних систем. Що в свою чергу, призводять до нових форм додаткових загроз, більш витонченим способам витоку інформації, у тому числі інсайдерських атак, і це далеко не повний перелік різноманіття і складності реальних загроз, до яких схильні інформаційні мережі підприємств. Тому задача класифікації мережевих з'єднань на підприємстві чи в установі є актуальною і потребує застосування сучасних алгоритмів.

Кількість переданих за 1 секунду пакетів (PPS) в 1 кварталі 2012 року збільшилася в чотири рази в порівнянні з четвертим кварталом 2011 року. Такого порядку темпи росту спостерігаються останні 5-6 років. Це обумовлено підвищенням пропускної здатності каналів та збільшенням потужності зловмисних мереж. Для аналізу інформації про з'єднання, що надходить у кількості 500 000 – 1 000 000 записів на тиждень, необхідно використовувати сучасні сховища даних. Сховища даних повинні забезпечувати високу швидкість отримання даних, можливість отримання і порівняння так званих зрізів даних, а також несуперечність, повноту і достовірність даних.

OLAP є ключовим компонентом побудови та застосування сховищ даних. Ця технологія заснована на побудові багатовимірних наборів даних - OLAP-кубів, осі якого містять параметри, а клітинки - залежні від них агрегатні дані. Програми з OLAP-функціональністю повинні надавати користувачеві результа-

ти аналізу за прийнятний час, здійснювати логічний і статистичний аналіз, підтримувати багатокористувацький доступ до даних, здійснювати багатовимірне концептуальне подання даних і мати можливість звертатися до будь-якої потрібної інформації.

У якості вихідних даних для роботи були розглянуті відомості про мережеву активність локальної мережі, підготовлені керуючою компанією MIT Lincoln Labs за звітами системних адміністраторів підприємства за дев'ять тижнів 2009 року (набір поділений на близько 5 мільйонів записів навчальної вибірки та близько 2 мільйонів записів тестової вибірки). Близько 20% з'єднань – нормальні. Решта – спроби несанкціонованих вторгнень різних видів (23 види).

Кожне з'єднання описано в базі вхідних даних 42-ма параметрами символьного, логічного та дійсного типів даних. Про наявність апріорного зв'язку між тим чи іншим параметром та видом з'єднання не відомо.

Обрана для вирішення *проблема* ідентифікації типу з'єднання пов'язана з побудовою моделі класифікації типів з'єднань на основі відомих параметрів.

Об'єктом дослідження в роботі є система захисту інформації локальної комп'ютерної мережі підприємства від небажаного вторгнення.

Предметом дослідження є алгоритми класифікації небезпечних комп'ютерних з'єднань, як багатопараметричних слабо визначених об'єктів, з використанням технологій OLAP.

Мета кваліфікаційної роботи – удосконалення алгоритму ідентифікації небажаного вторгнення в локальну мережу підприємства.

Для досягнення поставленої мети в роботі використані наступні *наукові методи*: агломеративна кластеризація – для кластеризації з'єднань, описаних символьними параметрами; багатовимірна логістична регресія – для побудови безпосередньо моделі ідентифікації та нейронні мережі Кохонена – для візуалізації багатовимірного представлення кластерів з'єднань.

У якості інструментарію для розв'язання проблеми запропоновано Weka - набір засобів візуалізації і бібліотека алгоритмів машинного навчання для вирішення задач інтелектуального аналізу даних (data mining) та прогнозування.

В ході досягнення загальної мети дослідження в роботі були поставлені й вирішені наступні *наукові та практичні задачі*:

– на основі аналізу структури початкових даних, типів даних та ступеню їх інформативності запропоновано структуру сховища даних, метрики та виміри, звернення за якими до нього приймаються;

– визначені методи дослідження, програмне середовище та шляхи розв'язання задачі;

– розроблено методологічні підходи до класифікації з'єднань на нормальні та небажані вторгнення, а також класифікації останніх за видами вторгнень;

– розроблено стійкий алгоритм ідентифікації нормальних з'єднань та класифікації небажаних вторгнень;

– здійснено експериментальну перевірку розробленого алгоритму ідентифікації на тестовій виборці.

В ході аналізу початкових даних було проаналізовано кожен з параметрів, що описують класи з'єднань у комп'ютерній мережі, визначено їх типи та діапазони зміни. Вхідним даним були призначені виміри багатовимірного сховища агрегованих даних (гіперкуба) та створений гіперкуб.

Простір даних навчальної вибірки було кластеризовано за перетином категорійних параметрів (протокол, служба та прапорець з'єднання). Виявлені 364 не порожніх кластерів, кожен з яких містить елементи одного або більше класів. Визначені 210 кластерів, які відповідають лише одному типу з'єднання. Кожному з них для ідентифікації записані правила вигляду

$$(Protocol = id1)\ \&\ (Service = id2)\ \&\ (Flag = id3) \rightarrow (Type = id\_klass)$$

де *id1, id2* та *id3* – ідентифікатори відповідно протоколу з'єднання, служби та прапорця, а *id_klass* – ідентифікатор класу з'єднань.

Для подальшої ідентифікації було застосовано механізм багатоваріантної логістичної регресії. Для кожного кластера були знайдені рівняння, які дозволяють класифікувати приклади з'єднань на один з припустимих класів.

Для відбору значимих параметрів, що включалися до моделі багатоваріантної логістичної регресії, застосовувалися карти Кохонена – нейронні мережі з

самонавчанням, що дозволяють візуально визначити й виключити з подальшого розгляду виміри, які не є класифікуючими.

Як показала експериментальна перевірка на навчальній виборці, система підтримки прийняття рішень з удосконаленим алгоритмом ідентифікації вторгнень демонструє надійність, близьку до 0,999, припускаючись в залежності від кластера від 0,0003% до 0,03% помилок.

Тестування на виборці, що містила 14 невідомих раніше типів атак та мала інше співвідношення між видами з'єднань, показала, що надійність системи при ідентифікації класу загрози, або найближчої до неї за типом становить не менше 0,968, а при розділенні з'єднань на нормальні й небезпечні – не менше ніж 0,99. Така висока точність ідентифікації свідчить про правильність обраного підходу до вирішення задачі, та розробленого алгоритму.

На захист виноситься наступне *наукове положення*:

Відокремлення категорійних параметрів опису складних слабко структурованих об'єктів від числових шляхом застосування на першому етапі агломераційної кластеризації, а потім на кожному з кластерів багатовимірної логістичної регресії дозволяє з надійністю не нижче 0,99 ідентифікувати ці об'єкти як елементи одного з наперед відомих класів.

Результати кваліфікаційної роботи магістра, а саме алгоритм роботи програмного забезпечення, що ідентифікує кожне нове з'єднання комп'ютерної мережі та визначає можливий тип вторгнення у реальному часі, може бути запропонований системним адміністраторам та фахівцям з безпеки корпоративних мереж.

Наукова новизна отриманих у роботі результатів полягає в наступному:

- обґрунтовано застосування методів агломеративної кластеризації та багатовимірної логістичної регресії для класифікації видів з'єднань, що встановлюються локальною комп'ютерною мережею із глобальною мережею;

- систематизовано множину ознак потенційно небезпечних комп'ютерних з'єднань для ідентифікації типу несанкціонованого вторгнення;

- покращена ефективність алгоритму класифікації типів небезпечних комп'ютерних з'єднань для запобігання несанкціонованого вторгнення з використанням технологій OLAP;

Отримані у роботі наукові результати відображено у публікації та апробовано виступами на двох студентських конференціях (всі – у 2012 році).

Практична цінність результатів полягає у розробці та реалізації стійкого алгоритму ідентифікації спроб несанкціонованого вторгнення у локальну мережу підприємства з використанням засобів інтелектуального аналізу даних, який би ефективно протидіяв зовнішнім атакам. Згаданий алгоритм не просто ідентифікує вид відомої мережевої атаки, а й вірно реагує на атаки нових типів.

Економічний ефект від реалізації результатів магістерської роботи досягається за рахунок усунення втрат від недобросовісної конкуренції, зменшення ймовірності викрадення та пошкодження цінної інформації, що є комерційною таємницею та необхідна для життєдіяльності підприємства. Витрати на створення чи вдосконалення системи захисту від небажаних електронних вторгнень не порівняні з можливими наслідками від пошкодження чи розповсюдження цієї інформації.

Соціальний ефект від впровадження результатів роботи в практику очікується позитивним завдяки підвищенню впевненості керівництва підприємства у захищеності своїх корпоративних даних, усуненню підозри в можливостях несанкціонованого доступу до даних та спрощенню роботи системного адміністратора й служби безпеки підприємства.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1 Дебра Литтлджон Шиндер. Основы компьтерных сетей. - Изд-во: Вильямс, 2002. - 615 с.

2 стандарти RFC 793, RFC 760, RFC 768, RFC 792

3 Брестский Государственный технический университет. Разработки. Активный интеллектуальный модуль защиты компьютерных систем от вредоносных программ и сетевых атак.

4 Отчет Prolexic Technologies по DDoS-атакам за 12 месяцев [Електронний ресурс] http://www.comss.info/page.php?al=ddoss_attack_12_monthes

5 NCC Group. Latest Industry News. Origins of Global Hacks. [Електронний ресурс] http://www.nccgroup.com/NewsAndEvents/Latest/12-02-01/Origins_of_Global_Hacks.aspx .

6 Stolfo S. J. "Cost-based Modeling for Fraud and Intrusion Detection: Results from the JAM Project" / Salvatore J. Stolfo, Wenke Lee, Wei Fan, Andreas Prodromidis, and Philip K. Chan // DARPA Information Survivability Conference. – 2000.

7 Чубукова И.А. Data Mining: учебное пособие / И.А. Чубукова.—М.: Интернет-университет информационных технологий: БИНОМ: Лаборатория знаний, 2006.— 382 с.

8 Steve R. Gunn Support Vector Machines for Classification and Regression // Technical Report (University of Southampton, Faculty of Engineering, Science and Mathematics, School of Electronics and Computer Science, 10 May 1998). http://www.svms.org/tutorials/Gunn1997.pdf

9 Журавлёв Ю.И. Об алгебраическом подходе к решению задач распознавания или классификации // Пробл. кибернетики. – М.: Наука, 1978. – Вып. 33. – С. 5–68

10 Тоби Сегаран. Программируем коллективный разум / Сегаран. Т. – Пер. с англ. – СПб: Символ-Плюс, 2008. – 368 с., ил.

11 BaseGroup Labs. Технологии анализа данных. [Електронний ресурс] http://www.basegroup.ru/

12 Профессиональный информационно-аналитический ресурс, посвященный машинному обучению, распознаванию образов и интеллектуальному анализу данных. [Електронний ресурс] http://www.machinelearning.ru

13 Paul D. Allison. Logistic Regression Using the SAS® System: Theory and Application, 1999. – 305 p.

14 Agresti A. Categorical Data Analysis / Alan Agresti. - Gainesville, Florida: University of Florida, 2002. – 732 p.

15 Джонс М. Т. Программирование искусственного интеллекта в приложениях / М. Тим Джонс; Пер. с англ. Осипов А. И. - М.: ДМК Пресс, 2004. - 312 с: ил.

16 Барсегян А.А. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP: [учебное пособие] / 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.

17 Erkki Oja. Kohonen maps / Erkki Oja, Samuel Kaski. - Elsevier Science BV, Netherlands, 1999. – 401 p.

18 Люггер Д.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем. – М.: Издательский дом «Вильямс», 2005. – 864 с.

19 Хайкин С. Нейронные сети: полный курс, 2-е издание. – М. Издательский дом «Вильямс», 2006. – 1104 с. (Haykin S.S. Neural Networks: A Comprehensive Foundation. – Prentice-Hall, second edition, 1998.)

20 Ralph Kimball, "The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses ", John Wiley & Sons, 1996

21 Ralph Kimball "The Data Webhouse Toolkit: Building the Web-Enabled Data Warehouse ", John Wiley & Sons, 2000

22 EF Codd, SB Codd, and CTSalley, Providing OLAP (on-line analytical processing) to user-analysts: An IT mandate. Technical report, 1993

23 Полубояров В.В. Использование MS SQL Server Analysis Services 2008 для построения хранилищ даннях / В.В. Полубояров. - Интуит, 2010. - 487 с.

24 Федоров А. Введение в OLAP / Алексей Федоров, Наталия Елманова // КомпьютерПресс. – 2001. - №4

25 Microsoft® SQL Server 2005 Analysis Services. OLAP и многомерный анализ данных / Бергер А.Б., Горбач И.В., Меломед Э.Л., Щербинин В.А., Степаненко В.П. / Под общ. ред. А.Б. Бергера, И.В. Горбач. – СПб.: БХВ - Петербург, 2007. – 928 с.: ил. – (В подлиннике).

26 Chris Webb. Expert Cube Development with Microsoft SQL Server 2008 Analysis Services / Chris Webb, Alberto Ferrari, Marco Russo. - Birmingham – Mumbai, 2009. – 360 p.

27 Joe Schafer. Multinomial Logistic Regression Models (Stat 544, Lecture 19), 2006. – 20 p.

28 le Cessie, S., van Houwelingen, J.C. Ridge Estimators in Logistic Regression. Applied Statistics, 1992. – 41(1):191-201.

29 Дьяконов А.Г. Анализ данных, обучение по прецедентам, логические игры, системы WEKA, RapidMiner и MatLab / А. Г. Дьяконов. – М: МГУ имени М.В. Ломоносова, 2010. – 278 с. (Практикум на ЭВМ кафедры математических методов прогнозирования, Учебное пособие)

30 Machine Learning Group at University of Waikato. [Електронний ресурс] http://www.cs.waikato.ac.nz/~ml/weka/

31 http://weka.wikispaces.com/

32 Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten (2009); The WEKA Data Mining Software: An Update; SIGKDD Explorations, Volume 11, Issue 1. // http://www.kdd.org/explorations/issues/11-1-2009-07/p2V11n1.pdf

33 Remco R. Bouckaert, Eibe Frank, Mark Hall, Richard Kirkby, Peter Reutemann, Alex Seewald, David Scuse WEKA Manual for Version 3-7-1 (файл поставляется вместе с программой Weka версии 3.7.1).

34 A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems by Kristopher Kendall Department of Electrical Engineering and Computer Science,May 21, 1999

ДОДАТКИ

Додаток А

Матеріали кваліфікаційної роботи

		Позначення	Найменування	Кількість листів	Примітка
1					
2			Документація		
3					
4		САУ.РД.23.13 ПЗ	Пояснювальна записка	94	Формат А4
5					
6		САУ.РД. 23.13 ДМ	Демонстраційні матеріали		Презентація Power Point на компакт диску
7					
8		САУ.РД. 23.13 КР	Копія роботи	1	(Диск CD-R)

					САУ. РД. 23.13. ДА. ПЗ.			
Зм	Лист	№ докум.	Підпис	Дата				
Розроб.		Педан			Матеріали магістерської роботи	Літ	Аркуш	Аркушів
К. розд.		Мінєєв						
Керівн.		Мінєєв				НТУ «ДП»; 124м-22-1		
Н. контр.		Хом’як						
Зав. каф		Желдак						

Додаток Б

ВІДГУК

на кваліфікаційну роботу магістра
студента групи 124м-22-1 (група) Педана Антона Миколайовича (ПІБ у родовому відмінку)
спеціальності 124 Системний аналіз

Тема кваліфікаційної роботи: «Удосконалення алгоритму ідентифікації небажаного вторгнення в локальну мережу підприємства в умовах недостатньої інформації із застосуванням технологій OLAP».

Обсяг кваліфікаційної роботи 94 стор.

Мета кваліфікаційної роботи: удосконалення алгоритму ідентифікації небажаного вторгнення в локальну мережу підприємства.

Тема кваліфікаційної роботи безпосередньо пов'язана з об'єктом діяльності магістра спеціальності 124 "Системний аналіз і управління", оскільки предметом дослідження в роботі є алгоритми класифікації небезпечних комп'ютерних з'єднань як багатопараметричних слабо визначених об'єктів з використанням технологій OLAP, що відповідає освітній кваліфікації магістра з системного аналізу.

Виконані в кваліфікаційній роботі завдання відповідають вимогам до професійної діяльності фахівця освітньо-кваліфікаційного рівня магістра.

Оригінальність наукових рішень полягає в застосуванні методів багатовимірної регресії, агломераційної класифікації та нейронних мереж Кохонена для побудови моделі ідентифікації небажаних вторгнень в локальну мережу компанії.

Практичне значення результатів кваліфікаційної роботи полягає у розробці та реалізації стійкого алгоритму ідентифікації спроб несанкціонованого вторгнення у локальну мережу підприємства з використанням засобів інтелектуального аналізу даних, який ефективно протидіє зовнішнім атакам. Згаданий алгоритм не просто ідентифікує вид відомої мережевої атаки, а й вірно реагує на атаки нових типів.

Висновки підтверджують можливість використання результатів роботи в роботі спеціалістів внутрішньої безпеки, а також системних адміністраторів великих підприємств і корпорацій.

Оформлення пояснювальної записки та демонстраційного матеріалу до неї виконано згідно з вимогами. Роботу виконано самостійно, відповідно до завдання та у повному обсязі.

Як керівник, до роботи зауважень не маю.

Кваліфікаційна робота в цілому заслуговує оцінки «відмінно» (90 балів).

З урахуванням висловлених зауважень автор заслуговує присвоєння кваліфікації "магістр з системного аналізу".

Керівник кваліфікаційної роботи магістра,
к.т.н., доцент кафедри СА та У _____ / Мінєєв О.С.