МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

ВИПУСКНА РОБОТА

на тему:

«Система виявлення кібератак. Інформаційна технологія машинного навчання системи виявлення кібератак»

Завідувач випускаючої кафедри **Довбиш А.С.**

Керівник роботи **Довбиш А.С.**

Студентка групи КБ – 71 **Зарудна К.О.**

СУМИ 2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Кафедра комп'ютерних наук

Затверджую_____

Зав. кафедрою Довбиш А.С.

"_____" _____2021 р.

ЗАВДАННЯ

до випускної роботи

Студентки четвертого курсу, групи КБ-71 спеціальності «Кібербезпека» денної форми навчання Зарудної Катерини Олександрівни.

Тема: «Система виявлення кібератак. Інформаційна технологія машинного навчання системи виявлення кібератак»

Затверджена наказом по СумДУ

№_____ від _____2021 р.

Зміст пояснювальної записки: 1) аналіз сучасного стану систем виявлення атак; 2) формалізована постановка задачі й формування завдань дослідження; 3) опис основних положень, математичних моделей і критеріїв, що використовуються інформаційно-екстремальною інтелектуальною технологією;5) розробка інформаційного й програмного забезпечення системи виявлення атак; 6) аналіз результатів моделювання.

Дата видачі завдання "_____" _____2021г.

Керівник випускної роботи _____ Довбиш А.С.

Завдання приняла до виконання _____ Зарудна К.О.

РЕФЕРАТ

Записка: 51 стор., 9 рис., 3 табл., 1 додаток, 8 джерел.

Мета роботи — підвищення функціональної ефективності системи виявлення кібератак, із застосуванням машинного навчання, на основі інформаційних мір інформаційно-екстремальної інтелектуальної технології (ІЕІТ).

Об'єкт дослідження — процес виявлення кібератак.

Предмет дослідження — модель і метод інформаційно-екстремального машинного навчання системи виявлення атак із паралельною оптимізацією системи контрольних допусків.

Результати — розроблено програмне забезпечення інформаційної технології машинного навчання системи виявлення кібератак. При цьому задача інформаційного синтезу системи виявлення кібератак розв'язана в рамках інформаційно-екстремальної інтелектуальної технології аналізу даних, яка базується на максимізації інформаційної спроможності системи в процесі машинного навчання. Програмна реалізація виконана за допомогою пакету прикладних програм для розв'язання задач технічних обчислень – MATLAB.

СИСТЕМА ВИЯВЛЕННЯ КІБЕРАТАК, ІНФОРМАЦІЙНО-ЕКСТРЕМАЛЬНЕ МАШИННЕ НАВЧАННЯ, ІНФОРМАЦІЙНО-ЕКСТРЕМАЛЬНА ІНТЕЛЕКТУАЛЬНА ТЕХНОЛОГІЯ, ІНФОРМАЦІЙНИЙ КРИТЕРІЙ, НАВЧАЛЬНА МАТРИЦЯ, ТРАФІК, КАТЕГОРІЙНА МОДЕЛЬ

ЗМІСТ

ВСТУП

Проблема забезпечення безпеки функціонування суб'єктів інформаційних відносин, захисту потоків інформації при використанні інформаційних і керуючих систем, що зберігають та обробляють інформацію все більше загострюється. Оскільки захищене середовище систем, що призначені для держави, економіки та суспільства залежить від способу функціонування систем і мереж, що є складовою частиною інформаційно-комунікаційного середовища.

Особливу небезпеку для комп'ютерних систем становлять кіберзагрози – це шкідлива, свідома спроба людини або організації проникнути або завдати шкоди інформаційній системі іншою особою чи організацією.

Прес-служба Державної служби спеціального зв’язку повідомляє, що в Україні, за період з 1 січня по 1 травня 2021 року було зафіксовано близько 800 тисяч кібератак, більшість серед яких - це мережеві атаки прикладного рівня. Також виявлено і заблоковано DDoS-атаки, що були спрямовані на вебресурси Національного антикорупційного бюро України (НАБУ), Офісу Президента України, Державної служби спеціального зв’язку, тощо [1].

Проблема попередження можливих кібератак вимагає вивчення побудови математичних моделей предметної області, відтворення алгоритмів систем виявлення атак і вторгнень та систем прийняття рішень.

Єдиним відомим підходом для розв’язання цієї проблеми є використання захисних засобів та методів, таких як антивірусні рішення, брандмауери, системи сповіщення та виявлення загроз і атак, але, на жаль, досі не існує універсального достовірного рішення виявлення кібератак. Отже, виникає необхідність створення комп’ютерно-інтегрованої в інформаційно-комунікаційну систему (ІКС) автоматизованої системи виявлення атак для своєчасного виявлення, запобігання та нейтралізації кіберзагроз [2].

1 АНАЛІЗ ПРОБЛЕМИ ДОСЛІДЖЕННЯ

1.1 Сучасний стан та тенденції розвитку системи виявлення атак

В останній час велика увага приділяється методам прогнозування роботи комп'ютерних мереж в різних умовах, а особливо дослідженням оцінки функціонування інформаційної інфраструктури при проведенні націлених на неї атак.

Ідентифікація та розпізнавання впливу на мережі зв'язку на основі аналізу трафіку, що циркулює в них, відбувається на основі використання методів виявлення аномалій.

Розпізнавання порушень безпеки проводиться зазвичай за допомогою евристичних правил і аналізу сигнатур уже відомих атак.

Серед систем виявлення вторгнень (Intrusion Detection System, IDS) найбільш поширеними є локальні IDS, що передбачають встановлення системи на кожному окремому персональному комп’ютері (ПК), та мережеві IDS, які збирають пакети, що надійшли в мережу через один конкретний пристрій та досліджують на аномальні ознаки перш ніж відправляти дані до інших вузлів мережі.

Виділяються та рекомендуються до застосування три види методів виявлення атак:

– метод виявлення аномалій або поведінковий метод;

– сигнатурний метод;

– комбіновані методи (ґрунтуються на використанні сигнатурного методу та методу виявлення аномалій) [3].

Сигнатурні методи – методи виявлення вторгнень на основі сигнатур, що зазвичай використовуються в IDS, в яких містяться сигнатури (шаблони) типових атак, створені на основі заголовків або отриманих мережевих пакетів.

Велика кількість сигнатур робить цей метод більш витратним з точки зору вартості обчислень [4].

Сенс сигнатурних методів полягає у вихідних даних, що зібрані хостовими й мережевими датчиками IDS з сигнатур атак та виконання процедури пошуку сигнатури атаки.

Найбільш часто використовуваними серед методів сигнатурного виявлення атак є метод контекстного пошуку, що полягає у виявленні заданої множини символів у вихідних даних. Атаки на основі аналізу мережевого трафіку можуть бути ефективно виявлені за допомогою контекстного пошуку, оскільки цей метод дає можливість максимально точно визначити параметри сигнатур, які необхідно виявити в потоці вихідних даних.

Також були розроблені ще два сигнатурні методи: метод, який базується на експертних системах і метод аналізу станів.

Методи, що базуються на експертних системах, дозволяють описати моделі атак природною мовою з високим рівнем абстракції. Експертна система, яка лежить в основі методів цього типу, складається з окремих баз даних: фактів і правил. Факти – це результуючі дані роботи ІС, а правила – алгоритми логічних рішень про атаку на основі набору фактів. Всі правила експертної системи записуються в форматі «якщо , то». База даних результуючих правил повинна описувати характерні ознаки атак, які зобов'язана виявляти IDS.

Метод аналізу станів або контролю частоти подій заснований на формуванні сигнатури атак у вигляді послідовності переходів ІКС з одного стану в інший. Кожна така зміна стану визначається при настанні в ІКС певної події, а комплекс цих подій визначається параметрами сигнатури атаки [3].

Перевагою даних методів є ефективне визначення атак на ІКС; зниження загальної кількості помилкових спрацьовувань; можливість достовірно та якісно оцінити використання конкретно визначеного інструментального засобу або методу атаки; можливість визначити параметри сигнатур найбільш

точно, а очевидними недоліками є необхідність оновлювати бази даних сигнатур для виявлення нових загроз; неможливо виявити атаки, сигнатури яких ще не описані в експертній системі; неможливо виявити атаки, сигнатури яких відрізняються від тих, що існують в системі [4].

Поведінкові методи – це методи, що базуються не на моделях інформаційних атак, а на моделях «нормального» функціонування ІКС. Методика будь-якого з цих методів полягає у виявленні розбіжностей між поточним станом роботи ІКС й режимом функціонування, який є зразковим для інформаційної системи. Якщо буде виявлена невідповідність між станом роботи системи та моделями «нормального» функціонування системи, то така активність розглядається як аномальна. Складністю цього принципу є розробка точної зразкової моделі «нормального» режиму функціонування ІКС.

Перевагою поведінкових методів є виявлення атаки без знання конкретних сигнатур; висока чутливість до змін станів ІКС [4]; можливість виявлення нових атак без модифікації або поновлення параметрів моделі. Але на жаль, створити точну модель «нормального» функціонування ІКС досить складно [3]. Недоліками методів даного типу є помилкові спрацювання при непередбаченому поводженні користувачів та при непередбачуваній мережевій активності; досить великі часові витрати на етапі навчання системи [4].

Серед поведінкових методів є найбільш поширеними ті, що базуються на статистичних моделях. Моделі такого типу визначають параметри, які характеризують «нормальну» поведінку інформаційної системи. Якщо було виявлено певне відхилення встановлених параметрів від зразкової моделі, то фіксується факт виявлення атаки. Прикладом таких параметрів можуть бути: рівень навантаження на лінії зв'язку, ступінь навантаження на процесор, загальний час роботи системних користувачів, кількість звернень до ресурсів мережі тощо.

На стадії вторгнення виявити атаку можна за допомогою обох методів. Оскільки будь-які аномальні дії мають певні характерні ознаками, які, можна представити як у вигляді сигнатури, так і описати як відхилення від «нормальної» поведінки ІКС. Найбільш ефективним є поєднання обох методів, для отримання найкращого можливого результату [3].

1.2 Методи аналізу трафіку

Методи аналізу трафіку, виявлення і класифікації аномалій та їх поєднання використовуються при розробці програмно-апаратних комплексів і систем виявлення вторгнень. Розрізняють системи виявлення вторгнень (Intrusion Detection System, IDS) за типами, в залежності від виду сенсора, що використовується, його розміщення і методів аналізу підсистеми.

Можна визначити три рівні розвитку технологій за «глибиною» аналізу для кожного окремого пакету, тобто збільшення рівня моделі The Open Systems Interconnection (OSI), дані якого були використані для аналізу (рис.1.1).

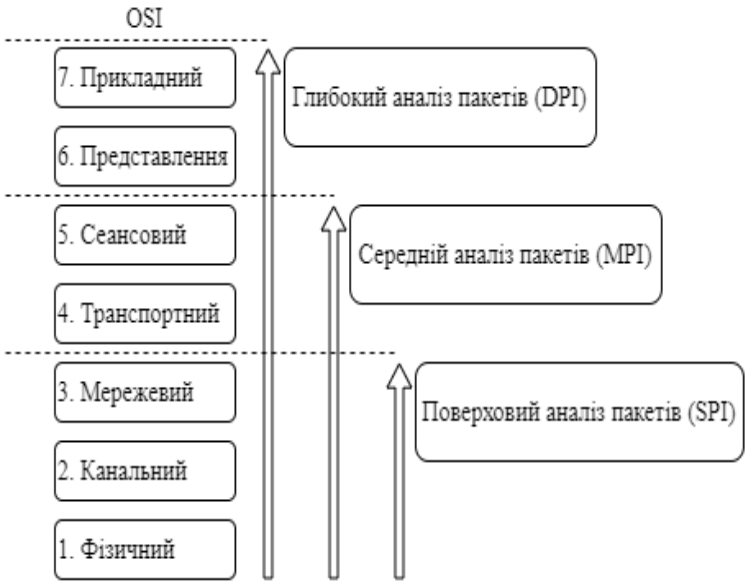


Рисунок 1.1 – Рівні розвитку технології аналізу мережевого трафіку за «глибиною».

Поверхневий аналіз пакетів (Shallow Packet Inspection, SPI) – це технологія аналізу трафіку, що базується виключно на заголовках пакетів рівнів 1-3 моделі OSI. Ця технологія вимагає низький рівень обчислюваних можливостей ресурсу, що дозволяє ефективно аналізувати досить великі обсяги трафіку [5].

Це один з методів аналізу пакетів, який застосовує поверхневу перевірку пакетів і ефективно розпізнає трафік, за умови, що мережеві атаки мають поводитись інакше, ніж звичайні потоки даних. Поверхневий аналіз пакетів в основному використовується для виявлення IP-адрес одержувача та джерела відправлення пакетів, номерів портів джерела та одержувача, імені протоколу вхідного пакета даних.

За допомогою встановлених даних, SPI ідентифікує ім’я програми з трафіку в режимі реального часу, та порівнює його з присвоєним номером (Internet Assigned Numbers Authority, IANA) цього конкретного порту. Якщо обидві назви збігаються, то трафік вважається безпечним, в іншому випадку, SPI вважає його аномальним [6].

SPI широко розповсюджена, на основі цієї технології працює більшість міжмережевих екранів операційних систем, маршрутизаторів та інших мережевих пристроїв. У такий спосіб, на основі SPI розроблені мережеві списки контроля доступу на рівні портів та IP адрес. Отже, система ефективно працює, виконуючи функції розмежування та отримання доступу для окремого комп'ютера (IP) та сервісам (портам) внутрішніх мереж [5].

Середній аналіз пакетів (Medium Packet Inspection, MPI) – це технологія аналізу трафіку, що базується на перевірці сеансів і сесій зв'язку, що були встановлені шлюзом-посередником, але ініційовані додатком. За алгоритмом даної технології, вміст пакетів аналізується вибірково, за встановленими правилами [5].

Технологія заснована на вузлах посередників, що називаються середніми. Вони розміщуються через всю мережу та здійснюють перевірку

пакетів за допомогою спеціальних програм, що працюють на цих пристроях, і здатні розпізнавати інформацію заголовка, а також деякі частини корисного навантаження даних. За допомогою проміжних блоків або вузлів у мережі MPI використовується для моніторингу та аналізу вхідних та вихідних пакетів даних із цих середніх блоків або вузлів. Технологія MPI працює як шлюз між комп'ютерами кінцевих користувачів та постачальником Інтернет послуг. Ця технологія може виконувати лише перевірку пакетів у транспортній, мережевій, лінійній передачі даних та фізичному рівні моделі OSI. Використовуючи технологію середнього аналізу пакетів, адміністратори мережі обмежували завантаження чи отримання шкідливих відео, зображень, аудіо тощо через мережу Інтернет [6].

MPI є більш гнучкою у порівнянні з SPI й, крім розмежування доступу, підходить для вирішення більшої кількості завдань – кешування вмісту, аналіз стисненого або зашифрованого трафіку, функціональні обмеження можливостей окремих протоколів методом заборони певних команд.

Головним недоліком MPI є погана масштабованість: кожна з команд та протоколів вимагає окремий «шлюз» (вхідні-вихідні порти). Крім того, робота в режимі проксі значно знижує потужність і швидкість обробки пакетів [5].

Глибокий аналіз пакетів (Deep Packet Inspection, DPI) – це найновіша технологія перевірки пакетів у режимі реального часу, яка використовується для моніторингу, а також одночасного аналізу заголовків та корисного навантаження пакета даних [6].

Іноді використовують вужчий термін - Deep Packet Processing (DPP), це технологія, яка може виконувати такі дії, як модифікація, фільтрація або перенаправлення.

Сьогодні терміни DPI та DPP часто можуть використовуватись як взаємозамінні. Ці технології є логічним наслідком розвитку MPI [5].

Технологія глибокого аналізу пакетів (DPI) призначена для того, щоб оператори мережі могли точно визначити вміст кожного пакету даних, що проходить через мережеві концентратори.

Пакети перевіряються на наявність конкретних підписів протоколів для ідентифікації мережевих програм. DPI технологія здатна виявляти протоколи та програми за допомогою трьох методів, а саме виявлення портів, виявлення підписів та евристичного методу.

Глибокий аналіз пакетів може виконувати завдання моніторингу всіх рівнів моделі OSI, що є значною перевагою цієї технології над MPI та SPI [6].

Технологія DPI є фактично поточним стандартом для функціонування засобів аналізу та моніторингу мережевого трафіку й належить до області критично важливих технологій, які є необхідними для забезпечення мережевої безпеки також вимог законодавства[5].

1.3 Формалізована постановка задачі інформаційного синтезу системи виявлення атак

Недоліками сучасних IDS є:

Значне навантаження ресурсів системи при функціонуванні IDS, що намагається виявити атаку в режимі реального часу;

Труднощі портативності, оскільки більшість IDS створюються для використання в певному програмному середовищі. Це призводить до ускладнення використання IDS в іншому середовищі.

Зміна продуктивності системи виявлення вторгнень у кожному середовищі, через відсутність загальних правил тестування.

Проаналізувавши методи роботи сучасних IDS можна визначити наступні недоліки:

– значний рівень помилкових спрацювань системи виявлення вторгнень й пропусків атак;

– неможливо визначити більшість вторгнень на ранніх етапах;

– не ефективні до виявлення нових атак;

– майже неможливо, визначити атакуючого, та встановити цілі атаки;

– відсутність оцінок точності отриманих результатів роботи;

– значне навантаження на ресурси системи, при функціонуванні IDS в режимі реального часу [3].

Метою роботи є створення комп'ютерно-інтегрованої в інформаційно-комунікаційну систему (ІКС) автоматизованої системи виявлення атак для своєчасного розпізнавання, запобігання та нейтралізації кіберзагроз, що здатна:

– негайно реагувати на кібератаки та несанкціоновані дії;

– зберігати та накопичувати дані про способи протидії, виявлення і реагування на атаки і несанкціоновані дії та використовувати їх для забезпечення надійного захисту ІКС різного призначення [2].

Визначимо формалізовану постановку задачі інформаційного синтезу системи виявлення атак стосовно визначень інформаційно-екстремальної інтелектуальної технології.

Нехай є алфавіт класів розпізнавання $\{X_m^o | m = \overline{1, M}\}$, які відображають можливі властивості вхідних трафіків, і навчальну матрицю типу «об'єкт – властивість» $||y_{m,i}^{(j)}||,\ i = \overline{1, N},\ j = \overline{1, n}$, де N – кількість ознак розпізнавання, а n – кількість структурованих векторів реалізацій відповідних класів розпізнавання. Поняття ІЕІ-технології полягає в трансформуванні вхідної навчальної матриці Y у робочу бінарну матрицю X, яка шляхом певних перетворень у процесі машинного навчання пристосовується до максимально повної ймовірності прийняття правильних класифікаційних рішень. Тому для бінарного простору Хеммінга задамо множину $\{g_m\}$ структурованих параметрів функціонування векторів, що впливають на функціональну ефективність машинного навчання системи виявлення атак (СВА). Вектор

параметрів машинного навчання СВА, що розпізнає реалізації класу X_m^o представимо у вигляді такої структури

$$g_m =< x_m, d_m, \delta >, \quad (1.1)$$

де x_m – усереднений вектор реалізацій, вершина якого визначає центр гіперсферичного контейнера класу розпізнавання X_m^o;

d_m – радіус гіперсферичного контейнера класу розпізнавання X_m^o;

δ – параметр поля контрольних допусків на ознаку розпізнавання [7].

На параметри машинного навчання, накладаються відповідні обмеження:

– закон розподілу реалізацій, для класу розпізнавання X_m^o, за якими визначається усереднена реалізація x_m, повинен бути наближеним до нормального;

– область значень радіуса контейнера класу розпізнавання X_m^o задається нерівністю $d_m < d(x_m \oplus x_c)$, де $d(x_m \oplus x_c)$ – міжцентрова відстань між реалізацією x_m і усередненою реалізацією x_c сусіднього класу X_c^o;

– область значень параметра δ для двобічних симетричних допусків задається нерівністю $\delta < 2a$, де a – число градацій контрольного поля допусків, яка є однаковою для всіх діагностичних ознак [7].

Необхідно:

– на етапі машинного навчання СВА оптимізувати параметри вектора (1.1), які забезпечують максимальне значення інформаційного критерію оптимізації в робочій (допустимій) області визначення його функції:

$$\overline{E}^* = \frac{1}{M} \sum_{m=1}^{M} \underset{G_E \cap \{k\}}{max} E_m^{(k)}$$

де $E_m^{(k)}$– обчислене на k-му кроці машинного навчання значення інформаційного критерію оптимізації параметрів навчання системи розпізнавати реалізації класу X_m^o;

G_E– робоча область обчислення інформаційного критерію; $\{k\}$ – множина кроків машинного навчання;

– за визначеними на етапі машинного навчання оптимальними геометричними параметрами контейнерів класів розпізнавання побудувати безпомилкові за навчальною матрицею вирішальні правила [7].

2 ОПИС МЕТОДУ ДОСЛІДЖЕННЯ

2.1 Основні принципи і визначення інформаційно-екстремальної технології аналізу даних

Машинне навчання у рамках інформаційно-екстремальної інтелектуальної технології (ІЕІ-технології) аналізу даних полягає у перетворенні апріорно нечіткого розбиття простору ознак у чітке розбиття класів розпізнавання методом оптимізації параметрів функціонування ІС. Одночасно здійснюється цілеспрямований пошук глобального максимуму багатоекстремальної функції статистичного інформаційного критерію у допустимій (робочій) області [2].

Методи інформаційно-екстремального машинного навчання ґрунтуються, на таких специфічних принципах, крім принципів системного аналізу:

– максимізації інформації, обґрунтованому екстремальністю сенсорного сприйняття образу. Цей принцип реалізують методом створення додаткових інформаційних обмежень, що підвищують різноманітність класифікованих об’єктів;

– дуальності – цей принцип полягає в реалізації на етапі апріорного моделювання простих алгоритмів у випадку їх цілеспрямованого уточнення способом поглиблення машинного навчання для наближення вирішальних правил до безпомилкових за навчальною матрицею;

– апріорної недостатності обґрунтування гіпотез (принцип Бернуллі – Лапласа), відповідно до умов апріорної невизначеності даних доречно розглядати апріорні гіпотези рівно ймовірними, інакше кажучи, рішення приймаються системою за найгірших умов у статистичному розумінні;

– рандомізації вхідної інформації, яка дає змогу досліджувати детерміновано-статистичні властивості процесу;

– редукції даних, що пояснює необхідність вдосконалення словника ознак розпізнавання способом виключення з нього неінформативних ознак та тих, що заважають, в інформаційному розумінні;

– зовнішнього доповнення, що обумовлює необхідність застосування навчальної або екзаменаційної (контрольної) вибірки для оцінки функціональної ефективності машинного навчання [2].

В рамках IEI-технології, вирішальні правила в процесі оптимізації параметрів машинного навчання створюються за методом відкладених рішень О. Г. Івахненка з багатоциклічною ітераційною структурою пошуку максимального граничного значення усередненого за алфавітом класів розпізнавання інформаційного критерію оптимізації:

$$g_{\xi}^{*} = arg \max_{G_{\xi}} \{\max_{G_{\xi-1}} \{\dots \{\max_{G_1 \cap G_E} \frac{1}{M} \sum_{m=1}^{M} E_m\} \dots\}\}, \quad (2.1)$$

де E_m – інформаційний критерій оптимізації параметрів навчання системи розпізнавати реалізації класу X_m^o;

G_E– допустима область визначення функції інформаційного критерію оптимізації параметрів машинного навчання;

G_{ξ} – допустима область значень ξ -ї ознаки розпізнавання;

Але на алгоритм інформаційно-екстремального машинного навчання (2.1) накладаються певні обмеження:

$$\left(\forall X_m^o \in \widetilde{\Re}^{|M|}\right)[X_m^o \neq \emptyset], \quad (2.2)$$

де $\widetilde{\Re}^{|M|}$ – розбиття простору ознак на класи розпізнавання з потужністю $Card\widetilde{\Re} = M$;

$$\left(\exists X_k^o \in \widetilde{\Re}^{|M|}\right)\left(\exists X_l^o \in \widetilde{\Re}^{|M|}\right)[X_k^o \neq X_l^o \rightarrow X_k^o \cap X_l^o \neq \emptyset]; \quad (2.3)$$

$$\left(\forall X_k^o \in \widetilde{\Re}^{|M|}\right)\left(\forall X_l^o \in \widetilde{\Re}^{|M|}\right)[X_k^o \neq X_l^o \to KerX_k^o \cap KerX_l^o = \emptyset], \quad (2.4)$$

де $KerX_k^o$ – ядро класу розпізнавання X_k^o;

$KerX_l^o$ – ядро класу розпізнавання X_l^o, найближчого сусіда для класу розпізнавання X_k^o;

$$\left(\forall X_k^o \in \widetilde{\Re}^{|M|}\right)\left(\forall X_l^o \in \widetilde{\Re}^{|M|}\right)[X_k^o \neq X_l^o \to (d_k^* < d(x_k \oplus x_l))\& \\ \&(d_l^* < d(x_k \oplus x_l))], \quad (2.5)$$

де d_k^*– оптимальний радіус контейнера класу розпізнавання X_k^o;

$d(x_k \oplus x_l)$ – кодова відстань між усередненим вектором x_k класу розпізнавання X_k^o і відповідним вектором x_l класу розпізнавання X_l^o;

d_l^* – оптимальний радіус контейнера класу розпізнавання X_l^o;

$$\bigcup_{X_m^o \in \widetilde{\Re}} X_m^o \subseteq \Omega_{\text{Б}}; k \neq l; k, l, m = \overline{1, M}, \quad (2.6)$$

де $\Omega_Б$ – бінарний простір Хеммінга.

Глибина інформаційно-екстремального машинного навчання визначається числом параметрів функціонування системи, що були оптимізовані за інформаційним критерієм. У той же час, внутрішній цикл процедури (2.1) реалізує базовий алгоритм інформаційно-екстремального машинного навчання, що покращує геометричні параметри контейнерів класів розпізнавання [2].

Головною метою машинного навчання методом IEI-технології є приведення вхідного математичного опису системи розпізнавання до максимально повної ймовірності класифікаційних рішень. Основною відмінністю методів інформаційно-екстремального машинного навчання від

нейроподібних структур є розробка у рамках функціонального підходу до моделювання когнітивних процесів, що притаманні людині під час формування та прийняття класифікаційних рішень, тож вони безпосередньо моделюють принцип роботи природного інтелекту. Процес машинного навчання розглядають як оптимізацію параметрів системи розпізнавання, що мають вплив на її функціональну ефективність. Такі параметри називають параметрами машинного навчання. В методах ІЕІ-технології можна використовувати будь-яку статистичну інформаційну міру різноманітності аналізованих об'єктів як критерій оптимізації. У методах ІЕІ-технології інформаційну міру визначають за кількістю параметрів машинного навчання, що оптимізуються. У той же час, достатню глибину інформаційно екстремального машинного навчання визначають відповідно до принципів відкладених рішень О. Г. Івахненка за умови досягнення граничного максимального значення усередненого за алфавітом класів розпізнавання інформаційного критерію оптимізації. Вирішальні правила будують за одержаними в процесі машинного навчання оптимальними геометричними параметрами контейнерів класів розпізнавання, що відновлюються в радіальному базисі бінарного простору ознак Хеммінга. Побудова вирішальних правил у рамках геометричного підходу змінює їх на майже інваріантні до багатовимірного простору ознак розпізнавання, тому що сучасні комп'ютерні системи можуть обробляти двійкові вектори, що містять 2^{85} ознак розпізнавання. Крім того, такі вирішальні правила характеризуються високою швидкістю прийняття класифікаційних рішень у разі функціонування СВА в режимі моніторингу, що є важливим фактором для виявленні атак [2].

2.2 Оцінка функціональної ефективності машинного навчання системи виявлення атак

Оцінка функціональної ефективності етапу машинного навчання є одним з важливих питань системи розпізнавання у процесі навчання, головними параметром якої є оперативність та достовірність класифікаційних рішень. Критерії оптимізації параметрів машинного навчання у методах IEI-технології можуть використовувати декілька ознак, які задовольняють такі особливості інформаційних критеріїв:

– інформаційна міра є дійсна і знакододатна функція від імовірності;

– кількість інформації для детермінованих подій ($p_i = 1$ або $p_i = 0$) рівна нулю;

– при значенні ймовірності $p_i = \frac{1}{m}$, де m – кількість якісних ознак розпізнавання, інформаційна міра має екстремум.

– сумісна інформаційна міра двох незалежних повідомлень дорівнює сумі їх відповідних інформаційних критеріїв [2].

Серед інформаційних критеріїв для оцінки функціональної ефективності СВА, що навчається, варто надати перевагу статистичним логарифмічним критеріям, які дозволяють працювати із меншими навчальними вибірками. Такими критеріями є ентропійні міри Шеннона та інформаційна міра Кульбака.

Представимо нормовану ентропійну міру оптимізації параметрів СВА, що навчається, розпізнавати вектори ознак класу X_m^o у вигляді:

$$E_m^{(k)} = \frac{I_m^{(k)}}{I_{\max}^{(k)}} = \frac{H_m^{(k)} - H_m^{(k)}(\gamma)}{H_m^{(k)}} \tag{2.7}$$

де $I_m^{(k)}$ – кількість умовної оброблюваної інформації на k -му кроці навчання СВА для розпізнавання реалізацій класу X_m^o;

$I_{max}^{(k)}$– максимальна можлива кількість умовної інформації, одержаної на k -му кроці навчання;

$$H_m^{(k)} = -\sum_{l=1}^{M} p(\gamma_{l,k}) \log_2 p(\gamma_{l,k}) \tag{2.8}$$

безумовна (апріорна) ентропія, яка існує на k -му кроці системи, що навчається, для розпізнавання реалізацій класу X_m^o;

$$(\gamma) = -\sum_{l=1}^{M} \sum_{m=1}^{M} p(\gamma_{l,k}) p(\mu_{m,k}/\gamma_{l,k}) \, log_2 \, p(\mu_{m,k}/\gamma_{l,k}) \tag{2.9}$$

умовна (апостеріорна) ентропія, яка характеризує залишкову невизначеність після k -го кроку навчання системи розпізнавати реалізації класу X_m^o; $p(\gamma_{l,k})$ – безумовна ймовірність прийняття на k -му кроці навчання гіпотези $\gamma_{l,k}$; $p(\mu_{m,k}/\gamma_{l,k})$– апостеріорна ймовірність прийняття на k -му кроці навчання рішення $\mu_{m,k}$ за умови, що обрана гіпотеза $\gamma_{l,k}$ [2].

Для системи оцінок з двома альтернативами (M = 2) та рівноймовірних гіпотез, які характеризують найважливішу в статистичному значенні ситуацію прийняття рішень, тоді після підстановки ентропій (2.8) і (2.9) у вираз (2.7) у відповідному порядку й заміни відповідних апостеріорних ймовірностей на апріорні за виразом Байєса ентропійний критерій приймає вигляд:

$$E_m^{(k)} = 1 + \frac{1}{2}\left(\frac{\alpha_m^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \log_2 \frac{\alpha_m^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} + \right.$$

$$+ \frac{\beta_m^{(k)}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} \log_2 \frac{\beta_m^{(k)}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} +$$

$$+ \frac{D_{1,m}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} \log_2 \frac{D_{1,m}(d)}{D_{1,m}^{(k)}(d) + \beta_m^{(k)}(d)} +$$

$$\left. + \frac{D_{2,m}^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \log_2 \frac{D_{2,m}^{(k)}(d)}{\alpha_m^{(k)}(d) + D_{2,m}^{(k)}(d)} \right), \quad (2.10)$$

де $\alpha_m^{(k)}(d)$ – помилка першого роду системи прийняття рішенн на k-му кроці машинного навчання;

$\beta_m^{(k)}(d)$ – помилка другого роду системи прийняття рішенн;

$D_{1,m}^{(k)}(d)$ – перша достовірність;

$D_{2,m}^{(k)}(d)$ – друга достовірність;

d– дистанційна міра, що визначає радіуси гіперсферичних контейнерів, побудованих у радіальному базисі простору Хеммінга [8].

Через те, що точнісні характеристики - це функції відстані роздільної гіперповерхні від геометричних центрів контейнерів відповідних класів розпізнавання, то критерій (2.10) у IEI-технології доцільно вважати нелінійним та взаємно-неоднозначним функціоналом від точнісних характеристик, який потребує обчислення в процесі навчання допустимої (робочої) області для його визначення.

Дослідимо процедуру знаходження модифікації ентропійного КФЕ за критерієм Шеннона для рішення з двома альтернативами у випадку рівноймовірних гіпотез. У зв’язку з тим що, інформаційний критерій є функціоналом від точнісних характеристик, тож при вибірковому обсязі, що

відтворює основні характеристики головної сукупності навчальної вибірки необхідно використовувати їх оцінки:

$$D_{1,m}^{(k)}(d)=\frac{K_{1,m}^{(k)}}{n_{\min}};\ \alpha_m^{(k)}(d)=\frac{K_{2,m}^{(k)}}{n_{\min}};\ \beta_m^{(k)}(d)=\frac{K_{3,m}^{(k)}}{n_{\min}};\ D_{2,m}^{(k)}(d)=\frac{K_{4,m}^{(k)}}{n_{\min}}, \quad (2.11)$$

де $K_{1,m}^{(k)}$ – кількість подій, що відображають приналежність реалізацій образу контейнеру $K_{1,mk}^{o}$, у випадку, коли дійсно $\{x_1^{(j)}\}\in X_1^o$;

$K_{2,m}^{(k)}$– кількість подій, що вказують на неналежність реалізацій контейнеру $K_{1,m}^{o}$, якщо дійсно $\{x_1^{(j)}\}\in X_1^o$;

$K_{3,m}^{(k)}$– кількість подій, що вказують на приналежність реалізацій контейнеру $K_{1,m}^{o}$, якщо вони насправді належать класу X_2^o;

$K_{4,m}^{(k)}$ – кількість подій, які вказують на неналежність реалізацій контейнеру $K_{1,m}^{o}$, у випадку, коли вони насправді відповідають класу X_2^o;

$n_{\min}$ – мінімальний обсяг репрезентативної навчальної вибірки [8].

Після виконання підстановки відповідних позначень (2.11) у вираз (2.10) одержуємо робочий вираз для обчислень у рамках IEI-технології ентропійного критерію машинного навчання СВА розпізнавання реалізацій класу X_1^o:

$$E_{1,m}^{(k)}=1+\frac{K_{3,m}^{(k)}}{K_{1,m}^{(k)}+K_{3,m}^{(k)}}\log_2\frac{K_{3,m}^{(k)}}{K_{1,m}^{(k)}+K_{3,m}^{(k)}}+\frac{K_{4,m}^{(k)}}{K_{2,m}^{(k)}+K_{4,m}^{(k)}}\log_2\frac{K_{4,m}^{(k)}}{K_{2,m}^{(k)}+K_{4,m}^{(k)}}\Bigg). \quad (2.12)$$

Дослідимо зміну диференціального інформаційного критерію Кульбака, яка має вигляд добутоку відношення реалістичності Λ на міру відхилень відповідних розподілів імовірностей [8].

Розглянемо відношення логарифму повної ймовірності $P_{t,m}^{(k)}$ достовірного прийняття рішень про приналежність векторів ознак класів X_m^o і

X_c^o контейнеру $K_{m,k}^o \in X_m^o$ до повної ймовірності хибного прийняття рішень $P_{f,m}^{(k)}$, що для системи з двома альтернативними оцінками рішень має вигляд

$$\Lambda = \log_2 \frac{P_{t,m}^{(k)}}{P_{f,m}^{(k)}} = \log_2 \frac{p(\mu_m)p(\gamma_{1,k}/\mu_m) + p(\mu_c)p(\gamma_{2,k}/\mu_c)}{p(\mu_m)p(\gamma_{2,k}/\mu_m) + p(\mu_c)p(\gamma_{1,k}/\mu_c)}, \qquad (2.13)$$

де $p(\mu_m)$–безумовна ймовірність появи реалізації класу X_m^o;

$p(\mu_c)$–безумовна ймовірність появи векторів ознак сусіднього класу X_c^o;

$\gamma_{1,k}$ – гіпотеза про належність контейнеру $K_{m,k}^o \in X_m^o$ векторів ознак класу X_m^o;

$\gamma_{2,k}$ – альтернативна гіпотеза.

Враховуючи (2.13) при допусках відповідно до принципів Лапласа-Бернуллі, тобто якщо $p(\mu_m) = p(\mu_c) = 0{,}5$, та після перевизначення апріорних умовних ймовірностей відповідними точнісними характеристиками загальний критерій Кульбака має остаточний вигляд:

$$E_{Km}^{(k)} = \log_2 \frac{P_{t,m}^{(k)}}{P_{f,m}^{(k)}} * [P_{t,m}^{(k)} - P_{f,m}^{(k)}] =$$

$$= \begin{vmatrix} P_{t,m}^{(k)} = 0{,}5D_{1,m}^{(k)}(d) + 0{,}5D_{2,m}^{(k)}(d) \\ P_{f,m}^{(k)} = 0{,}5\alpha_m^{(k)}(d) + 0{,}5\beta_m^{(k)}(d) \end{vmatrix} = 0{,}5\log_2\left(\frac{D_{1,m}^{(k)}(d) + D_{2,m}^{(k)}(d)}{\alpha_m^{(k)}(d) + \beta_m^{(k)}(d)}\right) *$$

$$* \left[(D_{1,m}^{(k)}(d) + D_{2,m}^{(k)}(d)) - (\alpha_m^{(k)}(d) + \beta_m^{(k)}(d))\right] =$$

$$= \log_2\left(\frac{2 - (\alpha_m^{(k)}(d) + \beta_m^{(k)}(d))}{\alpha_m^{(k)}(d) + \beta_m^{(k)}(d)}\right) * \left[1 - (\alpha_m^{(k)}(d) + \beta_m^{(k)}(d))\right]. \qquad (2.14)$$

Нормовану зміну критерію (2.14) можна представити у вигляді

$$E_{K,m}^{(k)} = \frac{E_{K\,m}^{(k)}}{E_{K\max}^{(k)}} \tag{2.15}$$

де $E_{Kmax}^{(k)}$ – значення інформаційного критерію при $D_{1,m}^{(k)}(d) = D_{2,m}^{(k)}(d) = 1$ і $\alpha_m^{(k)}(d) = \beta_m^{(k)}(d) = 0$ для формули (2.14).

У випадку виконання оптимізації значень функціонування СВА на етапі навчання системи за IEI-технологією не є обов'язковим нормування критеріїв оптимізації, у зв'язку з тим що, розв'язується завдання пошуку екстремальних значень параметрів навчання, що мають відповідність до глобального максимуму інформаційної міри у визначеній робочій області. Але доцільним є нормування критеріїв оптимізації при виконанні порівняльного аналізу результатів досліджень й при оцінці ступеня наближеності існуючої СВА до умовної [8].

Робочою модифікацією міри Кульбака після виконання відповідних підстановок оцінок у вираз (2.14) матиме вигляд

$$E = \frac{1}{n}\log_2\left\{\frac{2n+10^{-r}-\left[K_2^{(k)}+K_3^{(k)}\right]}{\left[K_2^{(k)}+K_3^{(k)}\right]+10^{-r}}\right\}\left\lfloor n-(K_2^{(k)}+K_3^{(k)})\right\rfloor \tag{2.16}$$

де r – кількість цифр у мантисі значення міри $E_m^{(k)}$.

Розглянемо алгоритм обчислення коефіцієнтів $K_2^{(k)}$ та $K_3^{(k)}$ у виразах (2.12) і (2.16). На рис. 2.1 відображено структуру навчальної матриці у випадку побудови оптимального контейнера для класу X_1^o. Навчальна матриця складається з послідовних векторів реалізацій $\{x_1^{(j)}\} \in X_1^o$ і $\{x_2^{(j)}\} \in X_2^o$ відповідно.

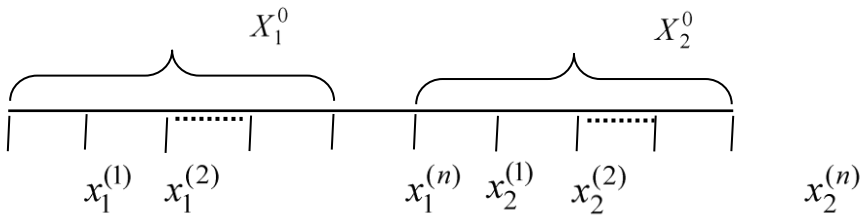


Рисунок 2.1 – Структура навчальної матриці

Схема обчислення коефіцієнтів $K_2^{(k)}$ і $K_3^{(k)}$ має такий предикатний вигляд:

$$\begin{gathered}(\forall X_1^o \in \Re^{|\Lambda|})(\forall X_2^o \in \Re^{|\Lambda|})[if\ \ x_1^{(j)} \in X_1\ \ then \\ K_1(j) := K_1(j-1)+1\ \ else\ K_2(j-1)+1]; \\ (\forall X_1^o \in \Re^{|\Lambda|})\,(\forall X_2^o \in \Re^{|\Lambda|})[if\ \ x_2^{(j)} \in X_1\ \ then \\ K_3(j) := K_3(j-1)+1\ else\ \ K_4(j) := K_4(j-1)+1]\end{gathered}. \tag{2.17}$$

Отже, інформаційні міри(2.10) та (2.14) є функціоналами від точнісних характеристик рішень, які приймаються, та від дистанційних критеріїв, інакше кажучи, їх можна сприймати як узагальнення відомих статистичних та детермінованих (дистанційних) мір оптимізації параметрів функціонування СВА [8].

2.3 Формування вхідної матриці системи розпізнавання

Головною метою формування вхідної матриці системи розпізнавання є створення багатовимірної навчальної матриці

$$||y_{m,i}^{(j)}|m = \overline{1,M}; i = \overline{1,N}, j = \overline{1,n}|| \tag{2.18}$$

При розв’язанні цієї задачі необхідно виконати:

– створення словника ознак й алфавіту класів розпізнавання;

– визначення мінімального об’єму навчальної вибірки;

– визначення нормованих допусків ознак розпізнавання.

Отже, формування вхідної матриці системи вимагає детального розгляду та аналізу властивостей функціонування джерела даних [8].

Вхідний математичний опис представимо у вигляді теоретико-множинної структури

$$\Delta_{\text{В}} = < G, T, \Omega, Z, Y;\ \text{П}, \Phi >, \tag{2.19}$$

де G – простір вхідних факторів (сигналів), що впливають на систему;

T – множина промыжків часу зчитування інформації;

Ω – простір ознак розпізнавання;

Z – простір можливих станів системи;

Y – множина сигналів, що знімаються з виходу блоку первинної обробки інформації;

$П:\ G \times T \times \Omega \rightarrow Z$ – оператор переходів,який відображає механізм зміни станів системи під впливом зовнішніх і внутрішніх факторів;

$Ф:\ G \times T \times \Omega \times Z \rightarrow Y$ – оператор формування вибіркової множини Y на вході у систему.

Отже, як множина випробувань W розглядається декартовий добуток наведених у (2.19) множин $W = G \times T \times \Omega \times Z$.

Словник ознак розпізнавання $\Sigma^{|N|}$, у якому $N = Card\Sigma^{|N|}$, складовими якого є первині ознаки, що є безпосередньо властивостями процесу, який досліджується з другорядних ознак, які є похідними від першочергових. Обов’язковою умовою є структурованість словника ознак. У практичному використанні значення параметрів, що зчитуються з датчиків інформації можуть бути первинними ознаками, або експериментальні дані, що були отримані безпосередньо під час дослідження процесу, враховуючи умови його реалізації. Найбільш розповсюдженими вторинними ознаками є різного виду

статистичні властивості векторів ознак класів $\{x_{m,i}^{(j)}|i=\overline{1,N}\}$, навчальних вибірок $\{x_{m,i}^{(j)}|j=\overline{1,n}\}$ чи загальної навчальної матриці.

Розробник інформаційного забезпечення або інформаційна системи, що здатна функціонувати в режимі кластер-аналізу можуть здійснювати формування алфавіту класів розпізнавання $\{X_m^o\}$. У цьому випадку варто враховувати, що збільшення потужності алфавіту при незмінному словнику ознак розпізнавання значно впливає на асимптотичні точнісні характеристики, які описують функціональну ефективність системного навчання, бо відбулося збільшення ступеня перетину класів розпізнавання. Одним із простих імовірнісних критеріїв перетину класів для визначеного алфавіту може бути представлено як відношення помилки другого роду до першої достовірності, що обчислюється на k-му кроці ітераційного навчання: $\eta=\frac{\beta^{(k)}}{D_1^{(k)}}$.

Ефективним шляхом покращення точнісних характеристик при збільшенні потужності алфавіту класів є формування ієрархічних алгоритмів навчання системи, які дозволяють класи розбивати на групи меншої потужності й здійснювати навчання для кожної з цих груп, та створення штучної надлишковості словника ознак [8].

2.4 Визначення мінімального обсягу навчальної вибірки

Під час практичного використання навчальна матриця має скінченний обсяг n, це пояснює існування статистичної похибки ε між імовірністю p_i й емпіричною частотою $\widetilde{p_i}=\frac{k_i}{n}$ знаходження значення i-ї ознаки розпізнавання у контрольному полі допусків $\delta_{K,i}$. Верхня границя оцінки похибки $\varepsilon=|p_i-\widetilde{p_i}|$ залежить від номера випробувань n та визначається за теоремою Муавра-Лапласа:

$$\mathrm{P}\left\{\left|\frac{k_i}{n}-p_i\right|\geq\varepsilon\right\}=\mathrm{P}\left\{\left|\frac{k_i-np}{\sqrt{np_iq_i}}\right|-\frac{\varepsilon\sqrt{n}}{\sqrt{p_iq_i}}\geq 0\right\}=2\Phi\left(-\frac{\varepsilon\sqrt{n}}{\sqrt{p_iq_i}}\right)\geq 2\Phi\left(-2\varepsilon\sqrt{n}\right),\quad(2.20)$$

де k_i – кількість подій, де значення *i*-ї ознаки розміщується у полі допусків $\delta_{K,i}$;

$q_i=1-p_i$ – ймовірність, що значення *i*-ї ознаки не належить до поля допусків $\delta_{K,i}$;

$\Phi(\ldots)$ – функція Лапласа.

Визначення мінімального об'єму n_{min} репрезентативної навчальної матриці виконаємо за умови одержання прийнятних із практичної точки зору статистичної похибки й швидкості алгоритму обчислень. Ці міркування є суперечливими, це пояснює компромісну поведінку розв'язання задачі. Скористаємося методом динамічного довірчого інтервального оцінювання, ідея якого полягає в побудові довірчого інтервалу після кожного з випробувань. Метод оцінює ймовірність p_i знаходження *i*-ї ознаки у полі контрольних допусків із ймовірністю довіри $1-Q$:

$$P\left\{\frac{k_i}{n}-\varepsilon_Q\leq p_i\leq\frac{k_i}{n}+\varepsilon_Q\right\}=1-Q,\qquad(2.21)$$

де Q – межа значущості (будь-яке додатне число наближене до нуля).

Розрахунок максимальної похибки ε_Q із заданим рівнем значущості Q виконується із співвідношення:

$$2\Phi(-2\varepsilon_Q\sqrt{n})=Q.\qquad(2.22)$$

Із врахуванням властивостей функції Лапласа $\Phi(\mathrm{x})=1-\Phi(-\mathrm{x})$ вираз (2.22) набуватиме вигляду

$$\Phi(2\varepsilon_Q\sqrt{n})=1-\frac{Q}{2}.\qquad(2.23)$$

Для визначення мінімального обсягу випробувань n_{min}, що гарантує прийнятні з практичного значення величини похибки й швидкість реалізації алгоритму обчислення, то необхідно визначити критерій зупинки випробувань.

Таким критерієм можна вважати випробування, за якого поточний довірчий інтервал покривається заданим інтервалом [0,5±Δ], де $|\Delta| < 0{,}5$. Для більшості практичних завдань значення Δ визначається з інтервалу $[0{,}3; 0{,}4]$. Правий перетин цього інтервалу з однією із меж довірчого інтервалу задає випробування n_{min}, що гарантує із ймовірністю $1-Q$, що максимальна похибка ε_Q не перевищує значення функції $\varepsilon_Q = f(n)$ при $n = n_{min}$.

Загалом треба створити довірчі інтервали для всіх N реалізацій і обрати n_{min} за умови

$$n_{\min} = \max_{\{i\}}(n_{\min 1}, \dots, n_{\min i}, \dots, n_{\min N})$$

У практичному значенні для незалежних ознак розпізнавання можна обирати n_{min} за довірчим інтервалом, що був побудований для будь-якої з ознак, що значно знижує обчислювальну навантаженість алгоритму [8].

3 ІНФОРМАЦІЙНЕ, АЛГОРИТМІЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ВИЯВЛЕННЯ АТАК

3.1 Категорійна модель машинного навчання системи виявлення атак з оптимізацією контрольних допусків

За IEI-технологією оптимізація системного вхідного математичного опису, у процесі навчання до її максимальної інформаційної спроможності виконується методом оптимізації параметрів машинного навчання за інформаційною мірою. У загальному випадку, реалізація базового алгоритму машинного навчання не гарантує високу якість розпізнавання трафіку під час функціонуванні СВА в режимі моніторингу, бо початкові значення контрольних допусків на ознаки розпізнавання не є оптимальними. Отже, виникає необхідність покращення глибини машинного навчання одним методів оптимізації системи контрольних допусків, які мають суттєвий вплив на геометричні параметри контейнерів класів розпізнавання й на точнісні характеристики класифікаційних рішень. Додатковим параметром оптимізації будемо вважати параметр δ поля контрольних допусків.

Категорійна модель навчання СВА з оптимізацією системи контрольних допусків на ознаки розпізнавання із урахуванням моделі базового алгоритму навчання представлена на рис. 3.1.

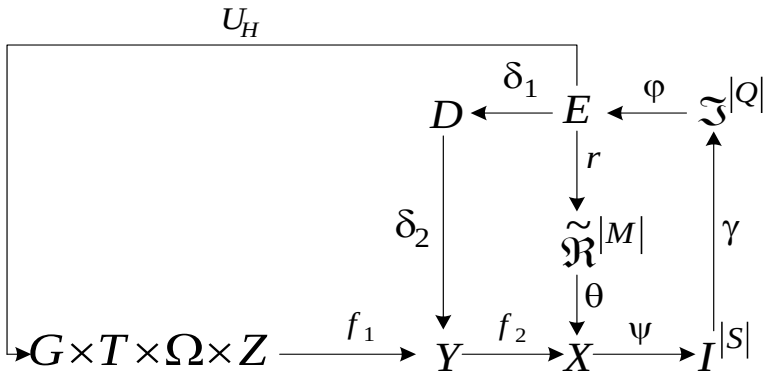


Рисунок 3.1– Категорійна модель машинного навчання з оптимізацією системи контрольних допусків

Категорійна модель, що зображена на рис.3, містить ще один контур операторів оптимізації контрольнх допусків на ознаки розпізнавання, що замикається через терм-множину D, яка складається із допустимих значень системи контрольних допусків.

Існує три основні схеми оптимізації системи контрольних допусків на ознаки розпізнавання:

– алгоритм паралельної оптимізації, який оптимізує контрольні допуски для всіх ознак одночасно;

– алгоритм послідовної оптимізації, який оптимізує контрольні допуски для всіх ознак одночасно послідовно для кожної ознаки розпізнавання при фіксованих (стартових) значеннях інших ознак;

– алгоритм оптимізації за зведеним полем допусків, який використовується як послідовно-паралельний алгоритм за наявності різних шкал вимірювання для окремих груп ознак розпізнавання.

Перевага паралельного алгоритму оптимізації СКД полягає у високій оперативності реалізації алгоритму, а недолік – алгоритм не дає можливості отримати точне значення глобального максимуму інформаційного критерію в робочій області визначення функції.

Перевагою алгоритму послідовної оптимізації СКД є обчислення точних значень глобального максимуму КФЕ у робочій області, а недолік – низька оперативність.

На практиці доцільно застосувати оптимізацію СКД на ознаки розпізнавання за паралельно-послідовним алгоритмом, з метою поєднання переваг цих алгоритмів. У цьому випадку реалізація паралельного алгоритму дозволить визначити початкові контрольні допуски, що є вхідними для алгоритму послідовної оптимізації.

У роботі детально розглянемо паралельний алгоритм оптимізації СКД.

Для алгоритму паралельної оптимізації контрольних допусків на ознаки розпізнавання вхідними даними є масив реалізацій образу

$\{y_m^{(j)} \mid m = \overline{1,M};\ j = \overline{1,n}\}$; областю значень параметра δ є інтервал $[1; \delta_H/2]$, де δ_H– ширина нормованого поля допусків.

Розглянемо кроки алгоритму реалізації паралельного алгоритму оптимізації СКД:

a) Обнуління лічильника кроків зміни параметра δ: l:=0.

b) Запуск лічильника: l:=l+1, обчислення нижніх та верхніх контрольних допусків для всіх ознак: $\{A_{HK,i}[l] := y_{1,i} - \delta[l]\}$ і $\{\mathrm{A}_{BK,i}[l] = y_{1,i} + \delta[l]\}$, $i = \overline{1,N}$, де $y_{1,i}$– вибіркове середнє значення i -ї ознаки для векторів-реалізацій базового класу X_1^o, що є важливим, для особи, що приймає рішення.

c) Реалізація базового алгоритму навчання.

d) Якщо $E_1^*[l] \geq E_1^*[l-1]$, то виконання пункту e, інакше – пункту f.

e) Якщо $\delta \leq \delta_H/2$, то виконання пункту b, інакше– пункту f.

f) $\{A_{HK,i}^* := A_{HK,i}[l-1]\}; \{A_{BK,i}^* := A_{BK,i}[l-1]\}, i = \overline{1,N};\quad E_1^* := E_1^*[l-1]$ і «зупин» [8].

3.2 Опис алгоритму машинного навчання системи виявлення атак

Базовий алгоритм інформаційно-екстремального машинного навчання СВА створюється у внутрішньому циклі структурованої ітераційної процедури машинного навчання.

Вхідні дані для алгоритму машинного навчання – це дійсний тривимірний масив реалізацій класів розпізнавання $\{y_{m,i}^{(j)} | m = \overline{1,M}; i = \overline{1,N}; j = \overline{1,n}\}$; значення параметра поля контрольних допусків δ на ознаки розпізнавання й рівні квантування координат усереднених двійкових векторів-реалізацій, $\{\rho_m\} = 0{,}5$ для всіх класів розпізнавання.

Базовим класом вважається клас X_1^o, що відповідає за нормальний стан функціонування ІС та відповідно до якого визначається система контрольних допусків

Кроки алгоритму машинного навчання системи виявлення атак :

a) Обчислення усередненого вектору-ознак $\{y_{1,i} \mid i = \overline{1,N}\}$ для навчальної матриці класу розпізнавання X_1^o;

b) Формування масиву $\{x_{1,i}^{(j)}\}$ двійкових векторів-ознак класу X_1^o за умовою

$$x_1^{(j)} = \begin{cases} 1, & \text{if} \quad y_{1,i} - \delta \le y_{1,i}^{(j)} \le y_{1,i} + \delta, \\ 0, & \text{if else} \end{cases} \tag{3.1}$$

c) Формування масиву усереднених двійкових векторів-реалізацій $\{x_{m,i} | m = \overline{1,M}, i = \overline{1,N}\}$, складові яких визначаються за умовою:

$$x_{m,i} = \begin{cases} 1, & if \quad \frac{1}{n} \sum_{j=1}^{n} x_{m,i}^{(j)} > \rho_m; \\ 0, & if else, \end{cases}$$

де ρ_m – рівень селекції координат двійкового вектора $x_m \in X_m^o$.

d) Розподіл множини еталонних векторів ознак на пари найближчих «сусідів»: $\Re_m^{|2|} =< x_m , x_l >$, де x_l– еталонний вектор ознак класу-сусіда X_l^o, за такими кроками:

1) структурування множини усереднених векторів, починаючи з вектора базового класу x_1;

2) побудова матриці розмірності $M \times M$ кодових відстаней між усередненими векторами ознак усіх класів розпізнавання;

3) обчислення мінімального елемента для кожного рядка матриці кодових відстаней, що відповідає стовпчику вектора. У випадку наявності декількох однакових мінімальних елементів вибирається один будь-який із них, оскільки вони є рівноправними;

4) формування структурованої множини елементів попарного розбиття $\{\Re_m^{|2|} | m = \overline{1, M}\}$, що задає план навчання.

e) Здійснення оптимізації кодової відстані d_m при $E_m(0) = 0$.

f) Алгоритм зупиняється при знаходженні максимуму інформаційного критерію оптимізації параметрів машинного навчання в робочій області визначення його функції.

Отже, головною функцією базового алгоритму машинного є визначення на кожному кроці навчання інформаційного критерію і організація пошуку його глобального максимуму в робочій області, визначення функції критерію з метою обчислення оптимальних параметрів (кодові відстані $\{d_m^*\}$, усередненні вектори-реалізації $\{x_m^*\}$ для заданого алфавіту $\{X_m^o\}$) розбиття простору ознак на класи розпізнавання [2].

3.3 Короткий опис програмного забезпечення

Під час виконання роботи було реалізовано алгоритм системи виявлення атак у режимі навчання. Програмна реалізація виконана за допомогою пакету прикладних програм для розв’язання задач технічних обчислень – MATLAB.

Пакет MATLAB представляє собою інструмент для вирішення наукових та прикладних задач, в таких областях як: моделювання об'єктів, розробка систем управління, проектування комунікаційних систем, обробка сигналів та зображень, вимірювання сигналів і тестування та ін.

MATLAB дає можливість користувачу використовувати високорівневу мову програмування, яка дає можливість працювати із широким спектром функцій, заснованими на матрицях структур даних, інтегрованим середовищем розробки, інтерфейсами до програм та об'єктно-орієнтованими можливостями.

Для побудови графіків за результатами розрахунків у MATLAB було використано табличний процесор Microsoft Office Excel.

Основні змінні, що були використані під час проектування системи виявлення атак у режимі навчання наведені у таблиці 3.1.

Таблиця 3.1 – Змінні, що були використані під час проектування СВА

Змінна	Опис
Y	Вхідний математичний опис
m	Кількість реалізацій класів розпізнавання
n	Кількість ознак класів розпізнавання
k	Кількість класів
deltaMax	Максимальне значення параметра поля контрольних допусків на ознаку розпізнавання
opti_delta	Оптимальне значення параметра поля контрольних допусків на ознаку розпізнавання
binMatrix	Робоча бінарна матриця
upperTolerance	Значення верхнього контрольного допуску
lowerTolerance	Значення нижнього контрольного допуску
x	усереднений вектор реалізацій
binMatrixNearest	Найближча сусідня бінарна матриця для класу розпізнавання
dc	Міжцентрові відстані від еталонного вектора до найближчого сусіднього класу розпізнавання
sk	Масив кодових відстаней від геометричних центрів контейнерів класів до їх реалізацій
E	Значення критерію функціональної ефективності за мірою Шеннона.
J	Значення критерію функціональної ефективності за мірою Кульбака.
opti_rE	Значення оптимальних радіусів контейнерів за мірою Шеннона

opti_rJ	Значення оптимальних радіусів контейнерів за мірою Кульбака

Для реалізації системи виявлення атак, що навчається було розроблено наступні функції, що наведені у таблиці 3.2.

Таблиця 3.2 – Функції системи у режимі навчання

Назва	Опис
binaryMatrix.m	функція трансформування вхідної навчальної матриці у робочу бінарну матрицю.
Main.m	головна функція програми.
nearest.m	функція пошуку найближчого сусіда для класу розпізнавання.
normalization_data.m	функція нормалізації вхідних даних.
optR_Kulbaka.m	функція розрахунку оптимальних радіусів контейнерів за мірою Кульбака.
optR_Shennona.m	функція розрахунку оптимальних радіусів контейнерів за мірою Шеннона.
parOpti.m	функція алгоритму паралельної оптимізації контрольних допусків на ознаки розпізнавання.
referenceVector.m	функція розрахунку усереднених векторів реалізацій.
sk.m	функція розрахунку масиву кодових відстаней.

Програмний код системи наведено у додатку.

3.4 Результати фізичного моделювання

Під час проектування системи виявлення атак було побудовано вхідний математичний опис Y, для процесу розпізнавання трьох класів, які є зразками

нормального та аномального трафіків: клас X_1^0 характеризує нормальний трафік, класи X_2^0 та X_3^0 – аномальний трафік.

Базовим класом будемо вважати X_1^0. Для отримання оптимальних результатів роботи системи виявлення атак, було реалізовано алгоритм навчання системи з паралельною оптимізацією системи контрольних допусків.

Значення рівня селекції становить 0,5 для всіх класів ознак. Ознаки розпізнавання були нормалізовані у проміжку $[1; 255]$, тому для отримання системи контрольних допусків будемо змінювати параметр δ з кроком 1 на цьому проміжку. Результати паралельної оптимізації представлено на рис.3.2

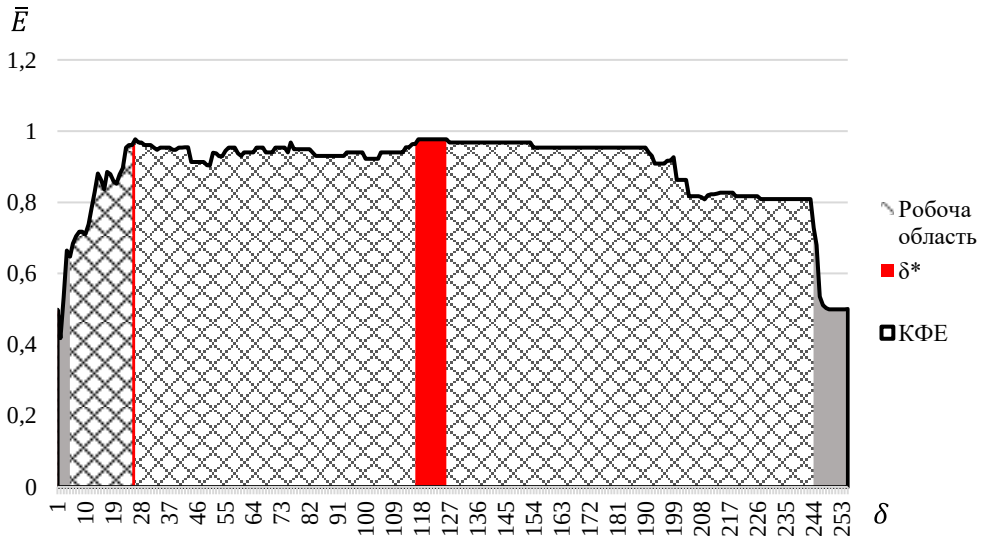


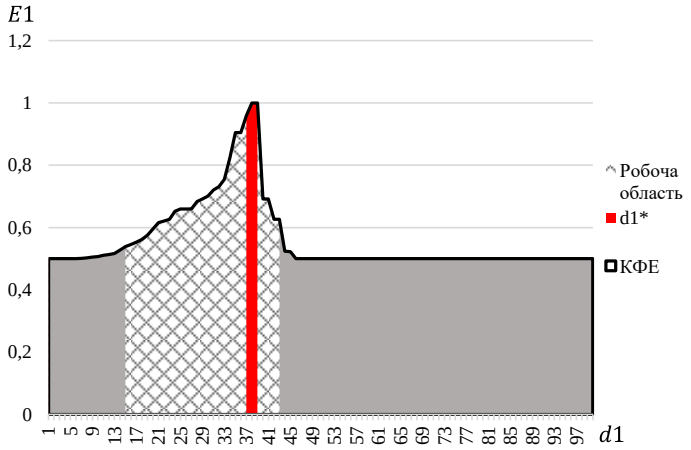
Рисунок 3.2 – Графік залежності усередненого критерію функціональної ефективності від ширини поля допусків

За рис. 3.2 можна встановити, що максимальне значення усередненого критерію функціональної ефективності від ширини поля допусків становить 0,9768 та було досягнуто на кроках 26, 117, 118, 119, 120, 121, 122, 123, 124, 125 та 126. Для подальшого навчання системи, оптимальне значення параметру δ будемо вважати 26.

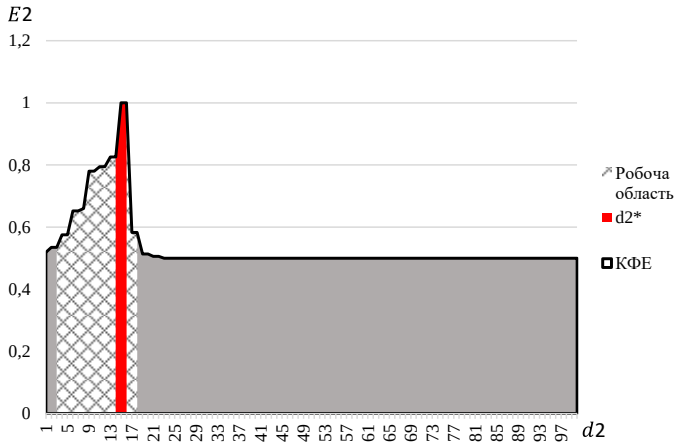
Під час навчання системи було визначено оптимальні значення радіусів: $d_1 = 38, d_2 = 15, d_3 = 7$, за ентропійною мірою Шеннона максимальні значення КФЕ для кожного класу становлять: $E_1 = E_2 = 1, E_3 = 0.93038$,

значення найбільшого усередненого КФЕ становить $\bar{E} = 0{,}97679$. Результати представлені на рис.3.3.

a)



b)



c)

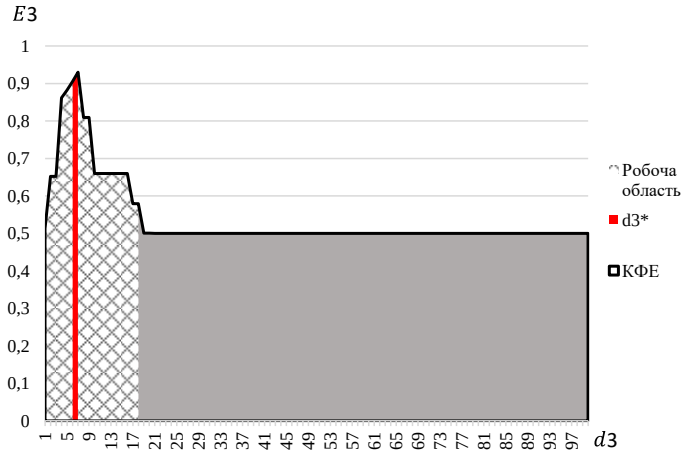
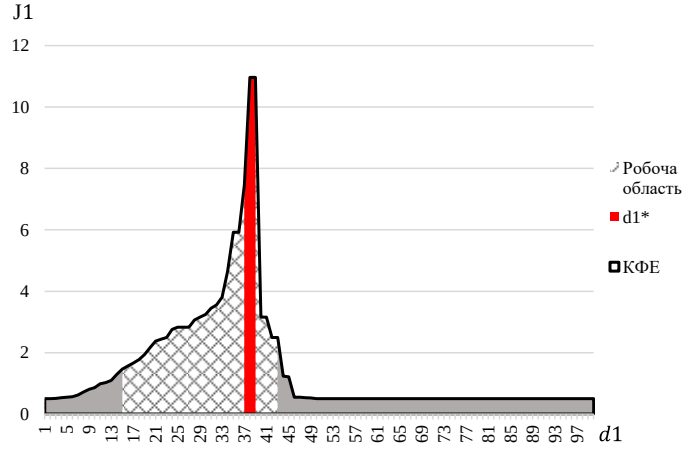


Рисунок 3.3 – Графіки залежності критерію функціональної ефективності від радіусів центрів за мірою Шеннона: a) – клас X_1^0, b) – клас X_2^0,c) – клас X_3^0

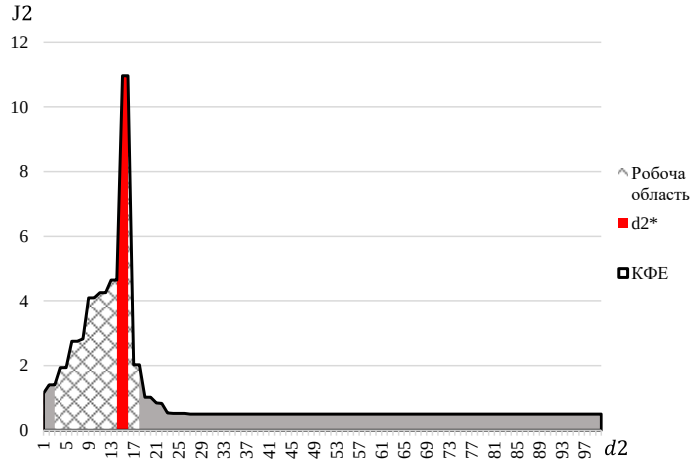
За інформаційною мірою Кульбака значення радіусів класів мають ті ж самі значення, що і за ентропійною мірою Шеннона, а максимальні значення

КФЕ за критерієм Кульбака для кожного класу становлять: $J_1 = J_2 = 10{,}96651, J_3 = 6{,}4285$, значення найбільшого усередненого КФЕ становить $\bar{J} = 9{,}45384$. Результати представлені на рис.3.4.

a)



c)



b)

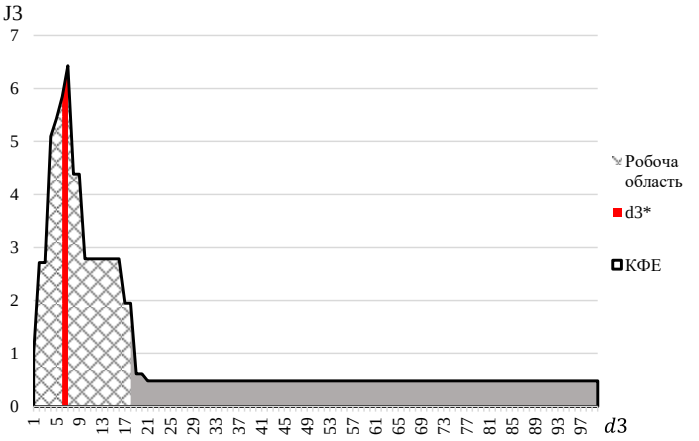


Рисунок 3.4 – Графіки залежності критерію функціональної ефективності від радіусів центрів за мірою Кульбака: a) – клас X_1^0, b) – клас X_2^0,c) – клас X_3^0

Результати навчання системи наведено у табл. 3.3.

Таблиця 3.3 – результати навчання системи виявлення атак

Клас	Опис класу	КФЕ за мірою Шеннона	КФЕ за мірою Кульбака	Радіус	D_1
X_1^o	Нормальний трафік	1	10,96651	38	1
X_2^o	Аномальний трафік	1	10,96651	15	1
X_3^o	Аномальний трафік	0,93038	6,4285	7	0,98

За результатами навчання було встановлено, що при оптимальному значенні параметру $\delta = 26$ та розрахованої на основі цього параметру системи контрольних допусків отримали наступні результати: радіуси контейнерів класів розпізнавання за мірами Шеннона та Кульбака збігаються і становлять $d_1^* = 38, d_2^* = 15, d_3^* = 7,$ максимальні значенні критеріїв становлять $E_1 = E_2 = 1, E_3 = 0.93038$ за критерієм Шеннона та $J_1 = J_2 = 10{,}96651,$ $J_3 = 6{,}4285$ за критерієм Кульбака.

Проаналізувавши табл. 3.1 можна зробити висновок, що після навчання системи виявлення атак з паралельною оптимізацією системи контрольних допусків можна сформувати вирішальні правила, точність яких у найгіршому випадку становить 98%. Отже алгоритм навчання СВА з паралельною оптимізацією системи контрольних допусків дозволить сформувати високоточні вирішальні правила.

ВИСНОВКИ

Під час виконання випускної роботи було проаналізовано сучасний стан та тенденції розвитку системи виявлення атак, розглянуто існуючі методи аналізу трафіку, виявлено недоліки існуючих систем, що полягають у високому рівні помилкових спрацювань системи СВА, неможливість визначити вторгнення на ранніх етапах, відсутності оцінок точності отриманих результатів роботи, значному навантаженні на ресурси системи, при функціонуванні СВА в режимі реального часу;

У процесі розв’язання проблеми було обрано та проаналізовано сучасну інформаційну-екстремальну інтелектуальну технологію для проектування системи виявлення атак, що здатна до навчання. Досліджено методи оцінки функціональної ефективності машинного навчання системи виявлення атак, формування вхідної матриці системи розпізнавання, визначення мінімального обсягу навчальної вибірки, формування вхідної навчальної матриці системи виявлення атак, категорійну модель машинного навчання СВА з оптимізацією контрольних допусків.

Під час розробки системи у середовищі прикладних програм для розв’язання задач технічних обчислень – MATLAB, було розроблено систему виявлення атак з паралельною оптимізацією системи контрольних допусків, що дозволяє сформувати високоточні вирішальні правила, точність яких у найгіршому випадку становить 98%.

СПИСОК ЛІТЕРАТУРИ

1. Прес-центр Державної служби спеціального зв'язку та захисту інформації України [Електронний ресурс] / Режим доступу: https://cip.gov.ua/ua.

2. Сучасні інформаційні технології в кібербезпеці : монографія / А. С. Довбиш, В. К. Ободяк, І. В. Шелехов та ін. ; за ред. В. К. Ободяка, І. В. Шелехова. – Суми : Сумський державний університет, 2021.

3. Бєрковський В. В., Безсонов О. С. Аналіз та класифікація методів виявлення вторгнень в інформаційну систему // Системи управління, навігації та зв’язку– 2017. – № 3(43) – С. 57–62.

4. Ananin E., Kozhevnikova I., Lysenko A., Nikishova A. Anomalies and intrusions detection methods [Електронний ресурс] / Режим доступу: https://cyberleninka.ru/article/n/metody-obnaruzheniya-anomaliy-i-vtorzheniy

5. Гетьман А. И., Евстропов Е. Ф., Маркин Ю. В. Анализ сетевого трафика в режиме реального времени: обзор прикладных задач, подходов и решений [Электронний ресурс] / Режим доступа: https://www.ispras.ru/preprints/docs/prep_28_2015.pdf

6. Argha Ghosh, A. Senthilrajan, Research on Packet Inspection Techniques // International Journal of Scientific & Technology Research – 2019. – № 8(11) [Електронний ресурс] / Режим доступу: https://www.researchgate.net/publication/339297592_Research_on_Packet_Inspection_Techniques.

7. Довбиш А. С., Симоновський Ю. В., Коробченко О. В., Летюга М. А. Інформаційно-екстремальний алгоритм машинного навчання системи розпізнавання транспортних засобів // Вісник НТУ «ХПІ» – 2016. – № 45(1217) – С. 22–28.

8. Довбиш, А. С. Основи проектування інтелектуальних систем [Текст] : навч. посіб. / А. С. Довбиш. – Суми : СумДУ, 2009. – 170 с.

ДОДАТОК

```
%open file 1
file1=fopen('1.txt');
tr1=[];
%reserved memory for array file1
for i = 1:115
    tr1=[tr1 , '%g'];
end;
% scan data from file1
a =fscanf(file1,tr1,[115 100]);
%close file1
fclose(file1);

%open file2
file2=fopen('2.txt');
tr2=[];
%reserved memory for array file2
for i = 1:115
    tr2=[tr2 , '%g'];
end;
% scan data from file2
b =fscanf(file2,tr2,[115 100]);
%close file2
fclose(file2);

%open file3
file3=fopen('3.txt');
tr3=[];
%reserved memory for array file3
for i = 1:115
    tr3=[tr3 , '%g'];
end;
% scan data from file2
c =fscanf(file3,tr3,[115 100]);
%close file2
fclose(file3);
```

```
%normalization
[ymin,ymax,a,b,c,Y,m,n,k]= normalization_data(a,b,c);

%Class 1 is base

% delta is step from 1 to 255
deltaMax=255;
[Edelta, WORKSPACE] = parOpti(Y, m, deltaMax);

WORKSPACE=find(WORKSPACE>0);

E_MAX=max(Edelta(WORKSPACE));
opti_delta=WORKSPACE(find(Edelta(WORKSPACE)==E_MAX));
opti_delta=opti_delta(1);

[binMatrix1,binMatrix2,binMatrix3,upperTolerance,lowerT
olerance,mathExpectation]=
binaryMatrix(Y,m,opti_delta);

[x1]= referenceVector(binMatrix1);
[x2]= referenceVector(binMatrix2);
[x3]= referenceVector(binMatrix3);

[binMatrixNearest1,dc1] = nearest(x1, binMatrix2, x2,
binMatrix3, x3);
[binMatrixNearest2,dc2] = nearest(x2, binMatrix1, x1,
binMatrix3, x3);
[binMatrixNearest3,dc3] = nearest(x3, binMatrix1, x1,
binMatrix2, x2);

[sk1,sk_nearest1]=
sk(x1,binMatrix1,binMatrixNearest1,m);
[sk2,sk_nearest2]=
sk(x2,binMatrix2,binMatrixNearest2,m);
[sk3,sk_nearest3]=
sk(x3,binMatrix3,binMatrixNearest3,m);

d_radius=1:m;
```

```
[E1,E_max1,tochn_char1,workspace1,opti_rE_1,D1_1,D2_1,a
lpha1,betta1]=
optR_Shennona(m,sk1,sk_nearest1,d_radius,dc1);
[J1, opti_rJ_1]=
optR_Kulbaka(D1_1,D2_1,alpha1,betta1,workspace1);
opti_rE_1=opti_rE_1(1);
opti_rJ_1=opti_rJ_1(1);


[E2,E_max2,tochn_char2,workspace2,opti_rE_2,D1_2,D2_2,a
lpha2,betta2]=
optR_Shennona(m,sk2,sk_nearest2,d_radius,dc2);
[J2, opti_rJ_2]=
optR_Kulbaka(D1_2,D2_2,alpha2,betta2,workspace2);
opti_rE_2=opti_rE_2(1);
opti_rJ_2=opti_rJ_2(1);

[E3,E_max3,tochn_char3,workspace3,opti_rE_3,D1_3,D2_3,a
lpha3,betta3]=
optR_Shennona(m,sk3,sk_nearest3,d_radius,dc3);
[J3, opti_rJ_3]=
optR_Kulbaka(D1_3,D2_3,alpha3,betta3,workspace3);
opti_rE_3=opti_rE_3(1);
opti_rJ_3=opti_rJ_3(1);

function [ymin,ymax,a,b,c,Y,m,n,k]=
normalization_data(a,b,c)
for i=1:115
    yminTr1=min(min(a(i,:)));
    ymaxTr1=max(max(a(i,:)));

    yminTr2=min(min(b(i,:)));
    ymaxTr2=max(max(b(i,:)));

    yminTr3=min(min(c(i,:)));
    ymaxTr3=max(max(c(i,:)));

    ymin = min(min(yminTr1, yminTr2),yminTr3);
    ymax = max(max(ymaxTr1, ymaxTr2),ymaxTr3);
```

```
    a(i,:) = (a(i,:)-ymin).*(255./(ymax-ymin));
    b(i,:) = (b(i,:)-ymin).*(255./(ymax-ymin));
    c(i,:) = (c(i,:)-ymin).*(255./(ymax-ymin));
end;

a=a';
b=b';
c=c';
Y(:,:,1)=a; %реалізації, ознаки, класи
Y(:,:,2)=b;
Y(:,:,3)=c;

[m n k] = size(Y);
End
function [Edelta, WORKSPACE] = parOpti(Y, m, deltaMax)

    for delta=1:deltaMax;


[binMatrix1,binMatrix2,binMatrix3,upperTolerance,lowerT
olerance]= binaryMatrix(Y,m,delta);

     [x1]= referenceVector(binMatrix1);
     [x2]= referenceVector(binMatrix2);
     [x3]= referenceVector(binMatrix3);

     [binMatrixNearest1,dc1] = nearest(x1, binMatrix2,
x2, binMatrix3, x3);
     [binMatrixNearest2,dc2] = nearest(x2, binMatrix1,
x1, binMatrix3, x3);
     [binMatrixNearest3,dc3] = nearest(x3, binMatrix1,
x1, binMatrix2, x2);

     [sk1,sk_nearest1]=
sk(x1,binMatrix1,binMatrixNearest1,m);
     [sk2,sk_nearest2]=
sk(x2,binMatrix2,binMatrixNearest2,m);
```

```
     [sk3,sk_nearest3]=
sk(x3,binMatrix3,binMatrixNearest3,m);

     d_radius=1:m;


[E1,E_max1,tochn_char1,workspace1,opti_rE_1,D1_1,D2_1,a
lpha1,betta1]=
optR_Shennona(m,sk1,sk_nearest1,d_radius,dc1);


[E2,E_max2,tochn_char2,workspace2,opti_rE_2,D1_2,D2_2,a
lpha2,betta2]=
optR_Shennona(m,sk2,sk_nearest2,d_radius,dc2);


[E3,E_max3,tochn_char3,workspace3,opti_rE_3,D1_3,D2_3,a
lpha3,betta3]=
optR_Shennona(m,sk3,sk_nearest3,d_radius,dc3);

    Edelta(delta)=(E_max1+E_max2+E_max3)./3;


if((D1_1>0.5)&(D2_1>0.5)&(D1_2>0.5)&(D2_2>0.5)&(D1_3>0.
5)&(D2_3>0.5))
            WORKSPACE(delta)=delta;
        end;

    end;
end

function
[binMatrix1,binMatrix2,binMatrix3,upperTolerance,lowerT
olerance,mathExpectation]= binaryMatrix(Y,m,delta)

mathExpectation=mean(Y(:,:,1));
upperTolerance=mathExpectation+delta;
lowerTolerance=mathExpectation-delta;
```

```
for i=1:m
    binMatrix1(i,:)=Y(i,:,1)>=lowerTolerance &
Y(i,:,1)<=upperTolerance;
    binMatrix2(i,:)=Y(i,:,2)>=lowerTolerance &
Y(i,:,2)<=upperTolerance;
    binMatrix3(i,:)=Y(i,:,3)>=lowerTolerance &
Y(i,:,3)<=upperTolerance;
end;

end

function [refVector]= referenceVector(binMatrix)

ro=0.5;
refVector=mean(binMatrix)>ro;

end

function [binMatrixN, dc] = nearest(x1, binMatrix1, x2,
binMatrix2, x3)

dc1 = sum(xor(x1, x2));
dc2 = sum(xor(x1, x3));

if dc1 < dc2
    binMatrixN = binMatrix1;
    dc = dc1;
else
    binMatrixN = binMatrix2;
    dc = dc2;
end

end

function [sk,sk_nearest]= sk(x,binMatrix,binMatrixN,m)

for i=1:m
    sk(i)=sum(abs(x-binMatrix(i,:)));
    sk_nearest(i)=sum(abs(x-binMatrixN(i,:)));
```

```
end;

sk=sk';
sk_nearest=sk_nearest';
end

function
[E,E_max,tochn_char,workspace,opti_r,D1,D2,alpha,betta]
= optR_Shennona(m,sk1,sk2,d_radius,dc)

for i=1:m
K1(i)=sum(sk1<=d_radius(i));
K2(i)=sum(sk2<=d_radius(i));
end

D1=K1/m;
alpha = 1-D1;
betta = K2/m;
D2 = 1-betta;

part1=alpha./(alpha+D2);
part1=part1.*log2(part1);
warning off;
part1(find(isnan(part1)))=0;

part2=D1./(D1+betta);
part2=part2.*log2(part2);
warning off;
part2(find(isnan(part2)))=0;

part3=betta./(D1+betta);
part3=part3.*log2(part3);
warning off;
part3(find(isnan(part3)))=0;

part4=D2./(alpha+D2);
part4=part4.*log2(part4);
warning off;
part4(find(isnan(part4)))=0;
```

```
E=1+0.5*(part1+part2+part3+part4);

tochn_char=[d_radius;K1;K2;D1;D2;alpha;betta;E]';

workspace=find((D1>0.5)&(D2>0.5)&(d_radius<=dc));
%оптимальный радиус по критерию Шеннона

if isempty(workspace)
    E_max=max(E);
    tmp=find(E==E_max);
    opti_r=-1;
    D1=D1(tmp(1));
    D2=D2(tmp(1));
else
    E_max=max(E(workspace));
    opti_r=(find(E(workspace)==E_max));
    opti_r=workspace(opti_r(1));
    D1=D1(opti_r);
    D2=D2(opti_r);
end

end

function [J, opti_rJ]=
optR_Kulbaka(D1,D2,alpha,betta,workspace)

J=0.5*log2((D1+D2+0.001)./(alpha +
betta+0.001)).*(D1+D2-alpha-betta);
J_max=max(J(workspace));
opti_rJ=workspace(find(J(workspace)==J_max));

end
```