МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ФАХОВИЙ БІЗНЕС-КОЛЕДЖ
Циклова комісія комп’ютерної інженерії та інформаційних технологій

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

СИСТЕМА ВИЯВЛЕННЯ ВТОРГНЕНЬ У КОМП’ЮТЕРНУ МЕРЕЖУ З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ

Виконав:
студент групи 1К-22
Спеціальності
123 «Комп’ютерна інженерія»
Максим ОЛІЙНИК
Керівник:
Маргарита МЕДОЛИЗ

Черкаси 2026

ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ФАХОВИЙ БІЗНЕС-КОЛЕДЖ

Циклова комісія комп’ютерної інженерії та інформаційних технологій

Спеціальність 123 «Комп’ютерна інженерія»

Освітня програма Комп’ютерна інженерія

ЗАТВЕРДЖУЮ

Завідувачка циклової комісії КІ та ІТ

_____Наталя ФАЛЬЧЕНКО

(підпис)

«____» _____2025 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Олійнику Максиму Сергійовичу

1. Тема кваліфікаційної роботи Система виявлення вторгнень у комп’ютерну мережу з використанням методів машинного навчання

Керівник роботи Медолиз Маргарита Миколаївна, викладач вищої категорії

затверджені наказом закладу передвищої освіти від 6 жовтня 2025 року № 84/1у.

2. Строк подання студентом кваліфікаційної роботи 05.06.2026

3. Вихідні дані до кваліфікаційної роботи: джерела та статті з тематикою про архітектуру комп’ютерних мереж та їхні слабкі місця; набори даних, що містять мережевий трафік (датасети) для навчання моделей; джерела про алгоритми машинного навчання, призначені для класифікації мережевого трафіку; методи попередньої обробки даних; показники оцінки ефективності виявлення аномалій; інструменти, що використовуються для розробки та аналізу.

4. Зміст кваліфікаційної роботи (перелік питань, які потрібно розробити) Аналіз сучасних загроз у комп’ютерних мережах та доступних програмних рішень для виявлення вторгнень; вивчення алгоритмів машинного навчання з метою аналізу мережевого трафіку; проєктування архітектури та створення функціональної моделі системи; розроблення програмних модулів для виявлення аномалій; тестування системи щодо ефективності виявлення атак та подальший аналіз отриманих результатів.

5. Дата видачі завдання 08.10.2025

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Терміни виконання етапів	Примітка про виконання з підписами керівника і студента
1	Вступ	14.10.2025	
2	Розділ 1 Аналітичний огляд методів та засобів виявлення вторгнень	09.12.2025	
3	Розділ 2 Проєктування системи виявлення вторгнень на основі машинного навчання	10.03.2026	
4	Розділ 3 Реалізація та експериментальне дослідження системи	28.04.2026	
5	Висновки	12.05.2026	
6	Оформлення кваліфікаційної роботи (чистовий варіант)	26.05.2026	
7	Перевірка кваліфікаційної роботи на наявність ознак плагіату (за 10 днів до захисту)	05.06.2026	
8	Подання кваліфікаційної роботи на затвердження завідувачу ЦК (за 7 днів до захисту)	08.06.2026	

Студент _____ Максим ОЛІЙНИК

(підпис)

Керівник роботи _____ Маргарита МЕДОЛИЗ

(підпис)

АНОТАЦІЯ

Кваліфікаційну роботу присвячено реалізації системи виявлення вторгнень у комп'ютерні мережі, яка базується на використанні алгоритмів машинного навчання. Метою роботи є впровадження програмного комплексу для моніторингу мережевого трафіку, автоматизації виявлення кібератак та ідентифікації аномальної активності. В межах дослідження обґрунтовано вибір інструментальних засобів та моделей класифікації даних, спрямованих на досягнення високої точності аналізу. До ключових технічних вимог до системи належать: функціонування в режимі, наближеному до реального часу, мінімізація часу обробки даних та оптимізоване використання обчислювальних ресурсів.

Розробка системи передбачає проєктуванні її архітектури, що включає такі основні функціональні блоки: модуль збору мережевих пакетів, етап попередньої обробки даних та модуль, який реалізує алгоритми машинного навчання для ідентифікації загроз. Крім того, у рамках роботи заплановано проведення тестування створеного програмного забезпечення на спеціалізованих наборах мережевих даних. Це дозволить оцінити ефективність використаних методів і сформулювати рекомендації щодо подальшого вдосконалення системи.

Ключові слова: СИСТЕМА ВИЯВЛЕННЯ ВТОРГНЕНЬ, МАШИННЕ НАВЧАННЯ, КОМП'ЮТЕРНІ МЕРЕЖІ, КІБЕРБЕЗПЕКА, АНАЛІЗ МЕРЕЖЕВОГО ТРАФІКУ, ВИЯВЛЕННЯ АНОМАЛІЙ.

ANNOTATION

The qualification work is devoted to the implementation of a system for detecting intrusions into computer networks, which is based on the use of machine learning algorithms. The purpose of the work is to implement a software package for monitoring network traffic, automating the detection of cyberattacks and identifying anomalous activity. The study substantiates the choice of tools and data classification models aimed at achieving high accuracy of analysis. The key technical requirements for the system include: operation in a mode close to real time, minimization of data processing time and optimized use of computing resources.

The development of the system involves designing its architecture, which includes the following main functional blocks: a network packet collection module, a data preprocessing stage, and a module that implements machine learning algorithms for threat identification. In addition, the work plans to test the created software on specialized network data sets. This will allow assessing the effectiveness of the methods used and formulating recommendations for further improvement of the system.

Keywords: INTRUDER DETECTION SYSTEM, MACHINE LEARNING, COMPUTER NETWORKS, CYBERSECURITY, NETWORK TRAFFIC ANALYSIS, ANOMALIES DETECTION.

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

Скорочення	Повна назва
AUC	Area Under Curve
IDS	Intrusion Detection System
NB	Naive Bayes
NN	Neural Network (штучна нейронна мережа)
NSL-KDD	Network Security Laboratory Knowledge Discovery in Databases
R2L	Remote to Local
ReLU	Rectified Linear Unit
RF	Random Forest
ROC	Receiver Operating Characteristic
SPAN	Switched Port Analyzer
U2R	User to Root

ЗМІСТ

ВСТУП

У XXI столітті спостерігається активний розвиток глобальних інформаційних мереж, що призвело до перенесення більшості процесів управління, фінансових операцій та бізнес-комунікацій у цифровий простір. Повсюдне поширення хмарних обчислень, технологій мобільного зв'язку та засобів віддаленої роботи зробило мережеву інфраструктуру основою функціонування сучасної держави. Проте разом із розширенням цифрових можливостей стрімко збільшується і кількість загроз інформаційній безпеці.

Сучасні мережеві небезпеки стають дедалі складнішими. Зловмисники використовують автоматизовані інструменти для пошуку вразливостей, створюють шкідливі програми, що здатні змінювати свій код для обходу систем захисту, та організовують масштабні атаки на корпоративні мережі. У таких умовах забезпечення цілісності та доступності даних стає критично важливим завданням для фахівців із комп’ютерної інженерії.

Традиційні засоби захисту, такі як міжмережеві екрани та класичні системи виявлення вторгнень, зазвичай працюють на основі зіставлення даних із відомими зразками атак. Такий підхід ефективний лише проти вже вивчених загроз, але виявляється безсилим перед новими методами зламу, які ще не були занесені до баз даних. Окрім того, постійне оновлення таких баз потребує значних зусиль та ресурсів, що створює часовий розрив між появою нової загрози та моментом, коли система зможе її зупинити.

Іншим способом захисту є аналіз поведінки мережі, де система шукає будь-які відхилення від звичайного робочого режиму. Проте стандартні статистичні методи часто помиляються, сприймаючи звичайне оновлення програм або завантаження великих файлів як спробу атаки. Це призводить до помилкових блокувань і перешкоджає нормальній роботі користувачів.

Розв'язання цих проблем полягає у використанні алгоритмів машинного навчання. Такі інтелектуальні системи здатні аналізувати величезні обсяги мережевого трафіку, самостійно знаходити підозрілі закономірності та відрізняти реальні атаки від легітимних дій користувачів. Використання моделей

на основі дерев рішень або нейронних мереж дозволяє автоматизувати захист і значно пришвидшити реакцію на вторгнення.

Водночас впровадження таких систем вимагає вирішення складних технічних завдань: правильної підготовки даних для навчання, вибору найважливіших ознак трафіку та забезпечення стабільної роботи обладнання без надмірного навантаження на процесор. Дослідження методів поєднання інтелектуального аналізу з налаштуваннями мережевого заліза визначає актуальність цієї роботи.

Метою кваліфікаційної роботи є дослідження та налаштування системи виявлення вторгнень у комп'ютерних мережах із використанням методів машинного навчання для автоматизованого розпізнавання атак.

Для досягнення поставленої мети визначено наступні завдання:

– проаналізувати сучасні загрози у комп’ютерних мережах та виконати їх класифікацію;

– розглянути існуючі програмні засоби захисту та порівняти їхні можливості;

– дослідити алгоритми машинного навчання, придатні для аналізу мережевого трафіку;

– обґрунтувати вибір архітектури та основних параметрів роботи системи;

– здійснити моделювання процесів обробки даних та вибору ознак для класифікації;

– вибрати інструменти розробки та підготувати набори даних для навчання моделей;

– налаштувати модулі класифікації трафіку та створити інтерфейс для моніторингу;

– протестувати систему та оцінити використання обчислювальних ресурсів.

Об’єктом дослідження є процеси виявлення аномальної активності та ідентифікації кіберзагроз у комп'ютерних мережах.

Предметом дослідження є методи та алгоритми машинного навчання, що застосовуються для класифікації типів мережевих атак. Реалізація цих методів спрямована на розробку підходів до автоматизації захисту мереж, які дозволяють виявляти загрози без необхідності ручного оновлення сигнатурних баз.

Практичне значення роботи полягає у визначенні методів автоматизації захисту комп’ютерних мереж від втручання. Дослідження алгоритмів аналізу трафіку та способів конфігурування мережевого обладнання дозволить визначити оптимальні шляхи для створення систем захисту, що працюють автономно. Отримані теоретичні та технічні знання можуть бути використані як база для подальшої розробки засобів виявлення загроз, які не потребують постійної підтримки сигнатурних баз.

Апробація результатів кваліфікаційної роботи здійснювалася шляхом їх представлення на XVIII Всеукраїнській науково-практичній конференції «Тенденції розвитку ІТ-технологій в Україні», 23-24 квітня, м. Черкаси.Тема доповіді: "Система виявлення вторгнень у комп’ютерну мережу з використанням методів машинного навчання".

РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД МЕТОДІВ ТА ЗАСОБІВ ВИЯВЛЕННЯ ВТОРГНЕНЬ

Швидке впровадження хмарних технологій та віддалених методів роботи значно розширило межі сучасних комп'ютерних мереж, що призвело до появи нових векторів кібератак та специфічних мережевих аномалій. Забезпечення безпеки інформаційних ресурсів у таких умовах вимагає комплексного підходу, який поєднує в собі як традиційні програмні та апаратні рішення для побудови ешелонованого захисту, так і новітні інтелектуальні технології аналізу даних. Перший розділ роботи присвячений аналітичному огляду сучасного стану мережевої безпеки, вивченню природи виникнення загроз та існуючих інструментів їх нейтралізації, що дозволяє сформувати логічний перехід від теоретичного вивчення проблеми до вибору інструментарію розробки.

Актуальність детального аналізу методів виявлення вторгнень зумовлена тим, що зловмисники постійно вдосконалюють інструментарій для обходу систем захисту, використовуючи вразливості нульового дня та методи прихованого сканування. Класичні методи, що базуються на суворій відповідності встановленим правилам, стають дедалі менш ефективними, що створює необхідність у проведенні порівняльного аналізу засобів захисту та обґрунтуванні вибору конкретних алгоритмів машинного навчання. Таке дослідження дозволяє визначити найбільш стійкі моделі моніторингу, які здатні виявляти не лише відомі загрози, а й нетипову поведінку в мережі. Отримані теоретичні висновки стануть базою для подальшої практичної реалізації дієвої моделі захисту комп’ютерної мережі у середовищі моделювання.

1.1 Аналіз сучасних загроз у комп’ютерних мережах та класифікація атак

Динамічний розвиток цифрової інфраструктури та перехід до хмарних сервісів зумовили появу нових складних загроз для безпеки комп'ютерних мереж. Під загрозою інформаційній безпеці у сучасних умовах розуміється потенційна

можливість здійснення дій, що можуть призвести до порушення конфіденційності, цілісності або доступності інформаційних активів [1]. При цьому мережева атака розглядається як реалізація такої загрози через експлуатацію вразливостей програмного чи апаратного забезпечення. Фундаментальною проблемою залишається те, що зловмисники постійно вдосконалюють методи втручання, використовуючи автоматизовані скрипти та інструменти штучного інтелекту для пошуку слабких місць у периметрі захисту [2].

Сучасний стан кіберзагроз характеризується переходом від масових нецілеспрямованих атак до складних багатоетапних операцій. За джерелом виникнення загрози традиційно поділяють на зовнішні, що ініціюються з глобальної мережі, та внутрішні, які пов'язані з діями авторизованих користувачів або компрометацією внутрішніх вузлів мережі [3]. Особливу небезпеку становлять загрози доступності, які мають на меті виведення з ладу критично важливих сервісів шляхом перевантаження каналів зв'язку або обчислювальних потужностей. Також значного поширення набули загрози цілісності, що передбачають приховану модифікацію системних файлів або баз даних, що може призвести до прийняття хибних управлінських рішень або повної втрати контролю над мережевою інфраструктурою [4].

Для побудови надійної системи моніторингу та подальшого навчання моделей виявлення вторгнень необхідно використовувати чітку класифікацію атак. Найбільш розповсюдженим підходом у міжнародній практиці є поділ мережевих аномалій на чотири основні класи залежно від їхньої функціональної мети. Першим класом є атаки типу відмови в обслуговуванні (DoS), де основне завдання зловмисника полягає у виснаженні ресурсів системи. Це досягається шляхом надсилання великої кількості запитів, які сервер не в змозі опрацювати, що призводить до відмови у наданні послуг легітимним користувачам. Типовими прикладами таких активностей, які часто фіксуються в сучасних датасетах, є атаки neptune та smurf [2].

Другим важливим класом є зондування та сканування – Probe, що за своєю суттю є розвідувальною діяльністю. Зловмисник намагається отримати детальну

інформацію про архітектуру мережі, активні порти та версії операційних систем, що дозволяє виявити вразливі місця для подальшого штурму. Наступним класом є атаки, спрямовані на отримання несанкціонованого віддаленого доступу – R2L. У цьому випадку особа, яка не має прав доступу до мережі, намагається отримати їх шляхом підбору паролів або експлуатації помилок у мережевих протоколах. Найбільш критичним за рівнем небезпеки є четвертий клас - ескалація привілеїв, коли користувач із обмеженими правами намагається отримати повний адміністративний контроль над вузлом мережі через переповнення буфера або інші програмні дефекти [5].

Ефективна протидія таким загрозам вимагає відмови від застарілих сигнатурних методів на користь інтелектуальних систем. Оскільки сучасні атаки часто мають поліморфну природу, тобто здатні змінювати свій цифровий відбиток, системи виявлення вторгнень повинні базуватися на аналізі поведінкових характеристик трафіку. Використання методів машинного навчання дозволяє ідентифікувати аномальну активність навіть у випадках, коли конкретний вектор атаки раніше не був зафіксований у базах даних засобів захисту. Це забезпечує вищий рівень адаптивності мережевої інфраструктури до нових викликів у сфері кібербезпеки [5].

1.2 Огляд існуючих програмних рішень для захисту мережевої інфраструктури

Забезпечення стійкості сучасної мережевої інфраструктури базується на використанні багаторівневої системи захисту, де кожен компонент відповідає за певний етап обробки даних та аналізу потенційних загроз. Класична концепція побудови безпечних систем передбачає впровадження стійкої оборони, що включає засоби міжмережевого екранування, антивірусні комплекси та спеціалізовані системи виявлення і запобігання вторгненням – IDS та IPS. Кожне з цих рішень має унікальні архітектурні особливості та специфічні методи обробки мережевого трафіку, що безпосередньо впливає на швидкість реагування на інциденти інформаційної безпеки [6].

Першим рівнем захисту в сучасних мережах виступають міжмережеві екрани нового покоління, це Next-Generation Firewalls та NGFW. На відміну від традиційних брандмауерів, які працювали лише на транспортному рівні моделі OSI, аналізуючи IP-адреси та порти, сучасні програмні продукти здійснюють глибоку інспекцію пакетів. Це дозволяє системі розпізнавати конкретні типи прикладного трафіку, навіть якщо вони намагаються замаскувати свою активність у дозволених каналах зв’язку, наприклад, через HTTP-туннелювання. Проте міжмережеві екрани часто не здатні розпізнати атаки, що ініціюються всередині вже встановлених легітимних з’єднань або експлуатують вразливості логіки додатків. Таке обмеження створює потребу у використанні засобів глибшого моніторингу, які фокусуються на аналізі вмісту та поведінки даних [7].

Центральне місце у системі моніторингу посідають мережеві системи виявлення вторгнень. Їхня робота базується на перехопленні копії трафіку з ключових вузлів мережі та його подальшому аналізі на відповідність відомим шаблонам атак. Одним із найбільш розповсюджених рішень у цій категорії є система Snort. Її внутрішня архітектура складається з бібліотеки збору даних, набору препроцесорів та механізму виявлення. Препроцесори відіграють критичну роль, оскільки вони відповідають за дефрагментацію пакетів та нормалізацію протоколів. Це необхідно для того, щоб зловмисник не міг обійти систему захисту шляхом навмисного розбиття шкідливого коду на дрібні фрагменти, які окремо не виглядають підозріло. Оновлена архітектура Snort 3 підтримує багатопотокову обробку, що дозволяє паралельно перевіряти тисячі пакетів на різних ядрах процесора, забезпечуючи високу пропускну здатність у сучасних високонавантажених мережах [8].

Альтернативним високоефективним рішенням є система Suricata. На відміну від багатьох аналогів, Suricata з самого початку розроблялася як багатопотоковий додаток, що дозволяє їй автоматично масштабуватися відповідно до апаратних можливостей сервера. Важливою технічною перевагою системи є вбудована підтримка ідентифікації протоколів прикладного рівня. Система здатна самостійно визначати тип трафіку, наприклад, TLS, DNS або

SMB, незалежно від того, на якому порту він передається. Це дозволяє ефективно блокувати спроби використання нестандартних портів для передачі команд керування ботнетами або витоку конфіденційних даних. Крім того, Suricata підтримує роботу з великими обсягами правил у форматі YAML, що спрощує процес адміністрування та автоматизації оновлень [9].

Окрему нішу серед засобів захисту займає система Zeek. Вона суттєво відрізняється від традиційних сигнатурних IDS за своєю філософією роботи. Zeek не просто шукає збіги з базою атак, а виступає в ролі потужного аналізатора мережевої поведінки. Система складається з двох основних частин: механізму обробки подій та інтерпретатора скриптів. Кожен мережевий пакет інтерпретується як певна подія, наприклад, встановлення TCP-з'єднання або запит до DNS-сервера, на яку можна написати власний сценарій реагування. Це дозволяє створювати надзвичайно гнучкі політики безпеки, які враховують специфіку конкретної організації, та здійснювати детальний ретроспективний аналіз інцидентів на основі згенерованих логів [10].

Поряд із мережевими системами важливу роль відіграють вузлові системи виявлення вторгнень, прикладом яких є платформа Wazuh. Такі рішення встановлюються безпосередньо на кінцеві пристрої та сервери для моніторингу внутрішніх процесів. Wazuh аналізує цілісність системних файлів, відстежує зміни в реєстрах та контролює виклики до ядра операційної системи. Використання цих систем дозволяє побудувати цілісну систему захисту, де мережеві аномалії корелюються з підозрілими змінами у файловій системі конкретного вузла. Така інтеграція є основою для побудови систем класу SIEM, які збирають та аналізують інформацію з усіх доступних джерел для виявлення складних, багатоетапних кібератак [11].

Розглядаючи існуючі програмні рішення, не можна оминути апаратні комплекси від провідних світових виробників, таких як Cisco та Fortinet. Ці системи базуються на спеціалізованих мікросхемах ASIC, які виконують перевірку трафіку на фізичному рівні без значного завантаження центрального процесора. Апаратні IPS забезпечують найвищу швидкість обробки даних, що

критично для магістральних каналів зв'язку. Проте їхня закрита архітектура та складність модифікації під специфічні дослідницькі завдання часто роблять їх менш привабливими для розробки нових інтелектуальних методів захисту порівняно з рішеннями з відкритим кодом [12].

Незважаючи на високий рівень технологічного розвитку розглянутого програмного забезпечення, більшість існуючих засобів мають спільну концептуальну проблему – вони орієнтовані на виявлення загроз за вже відомими шаблонами або правилами. Це означає, що при появі нових типів атак, опис яких ще не внесений до сигнатурних баз, ефективність захисту різко знижується. Вирішення цієї проблеми полягає в інтеграції існуючих систем із модулями аналізу на основі машинного навчання, які здатні самостійно знаходити аномалії в трафіку, вивчаючи нормальну поведінку мережі. Саме такий підхід дозволить автоматизувати процес моніторингу та підвищити рівень безпеки без необхідності постійного ручного оновлення правил захисту.

1.3 Порівняльний аналіз засобів виявлення вторгнень та постановка задачі дослідження

На основі проведеного аналізу існуючих програмних та апаратних рішень можна стверджувати, що вибір конкретного засобу захисту залежить від архітектури мережі, обсягів трафіку та вимог до оперативності реагування. Проте для об’єктивного обґрунтування подальшого напрямку розробки необхідно провести систематизоване зіставлення ключових параметрів розглянутих систем. Перш ніж перейти до безпосереднього зіставлення програмних продуктів, слід визначити критерії, за якими проводиться оцінка ефективності сучасних систем виявлення вторгнень.

Основним показником якості роботи будь-якої IDS є здатність точно ідентифікувати загрозу при мінімальній кількості помилкових тривог. У сучасних умовах, коли обсяги трафіку в корпоративних мережах постійно зростають, критичного значення набуває продуктивність системи, тобто її здатність обробляти пакети без затримок та втрат. Важливим аспектом є також

адаптивність – можливість системи підлаштовуватися під нові типи втручань без необхідності повної переконфігурації ядра захисту. Крім того, враховується глибина аналізу протоколів прикладного рівня, оскільки більшість сучасних атак реалізуються саме всередині легітимних сесій HTTP або DNS.

Для проведення порівняльного аналізу було обрано найбільш розповсюджені рішення з відкритим кодом, які мають різні архітектурні підходи: сигнатурні Snort та Suricata, а також поведінковий Zeek. Такий вибір дозволяє охопити весь спектр технологій, що використовуються для моніторингу мережевої безпеки. Систематизація технічних можливостей та функціональних параметрів розглянутих засобів захисту мережевої інфраструктури наведена в таблиці 1.1.

Таблиця 1.1 – Порівняльна характеристика засобів виявлення вторгнень

Характеристика	Snort	Suricata	Zeek
Метод виявлення	Аналіз за правилами	Правила та протоколи	Аналіз подій та логів
Багатопотоковість	Підтримується повністю	Підтримується повністю	Потребує кластеризації
Глибина інспекції	Глибока перевірка пакетів	Автовизначення сервісів	Скриптова інтерпретація
Формат правил	Власна мова правил	Сумісна зі Snort	Спеціальна мова Zeek
Продуктивність	Висока на нових CPU	Дуже висока на ядрах	Середня, адже вимоглива до RAM
Основна роль	Активний захист	Захист та моніторинг	Глибока аналітика

Аналіз даних, наведених у таблиці 1.1, дозволяє виявити фундаментальну проблему сучасних засобів захисту. Унікальний підхід забезпечує високу точність при виявленні вже відомих атак, проте він виявляється неефективним проти загроз нульового дня, тобто критичні вразливості в програмному чи апаратному забезпеченні, про які розробники ще не знають. Будь-яка модифікація коду атаки, яка ще не внесена до бази сигнатур, дозволяє зловмиснику обійти периметр захисту. Водночас системи поведінкового аналізу часто генерують велику кількість хибних спрацювань, що вимагає постійного втручання адміністратора для верифікації кожного інциденту [13].

Виникнення науково-технічного протиріччя між необхідністю високої швидкості виявлення нових типів втручань та мінімізацією помилкових тривог зумовлює перехід до інтелектуальних методів моніторингу. На відміну від жорстких правил, алгоритми машинного навчання дозволяють аналізувати статистичні характеристики мережевих потоків, виявляючи приховані зв’язки між параметрами трафіку, які неможливо описати статичними сигнатурами. Саме використання інтелектуальних моделей класифікації дозволяє автоматизувати процес розпізнавання загроз та забезпечити активний захист мережевої інфраструктури [14].

З огляду на виявлені недоліки традиційних систем, формулюється безпосередня задача подальшого дослідження, яка полягає у практичній перевірці ефективності алгоритмів машинного навчання. Для комплексного розв’язання цієї проблеми визначено такі етапи експериментальної реалізації:

- здійснити попередню обробку та аналіз параметрів еталонного набору даних мережевого трафіку NSL-KDD для виокремлення найбільш інформативних ознак атак;
- сформувати моделі класифікації на основі різних алгоритмів машинного навчання, зокрема випадкового лісу, нейронних мереж та наївного баєсівського класифікатора;
- провести порівняльну оцінку точності та продуктивності обраних моделей у середовищі моделювання Orange Data Mining за допомогою метрик точності (CA), площі під кривою (AUC) та матриці помилок;
- обґрунтувати вибір оптимального алгоритму для інтеграції в автоматизовані системи моніторингу безпеки комп'ютерних мереж.

Завершуючи аналітичний огляд, слід підкреслити, що подальше вдосконалення засобів виявлення вторгнень має базуватися на докорінній зміні парадигми захисту: від пасивного очікування оновлень сигнатурних баз до проактивного виявлення аномалій у реальному часі. Використання методів машинного навчання дозволяє системі самостійно формувати динамічну модель «нормальної» поведінки мережевого середовища та ідентифікувати будь-які

статистично значущі відхилення як потенційні вектори атак. Такий підхід забезпечує значно вищий рівень автономності засобів безпеки та мінімізує негативний вплив людського фактора на процес моніторингу критичної IT-інфраструктури.

Проведений порівняльний аналіз існуючого програмного забезпечення підтвердив, що для вирішення поставлених завдань необхідно використовувати інструментарій, здатний ефективно обробляти великі масиви даних та проводити порівняльну оцінку декількох алгоритмів класифікації одночасно. Саме порівняння архітектур алгоритмів – від відносно простих імовірнісних моделей до складних багатошарових нейронних мереж – дозволить визначити найбільш оптимальне рішення за критеріями точності та швидкодії.

Здійснено аналіз сучасних загроз інформаційній безпеці та проведено огляд існуючих програмно-апаратних засобів виявлення вторгнень. Встановлено, що класичні сигнатурні системи поступово втрачають свою ефективність в умовах стрімкого зростання кількості атак нульового дня та використання зловмисниками поліморфних методів маскування. Це обґрунтовує об'єктивну необхідність переходу від жорстких правил фільтрації до інтелектуальних методів аналізу поведінки мережі. Отже найбільш перспективним напрямком розв'язання цієї проблеми є впровадження алгоритмів машинного навчання, здатних автономно виявляти аномалії в трафіку на основі його поведінкових характеристик, що формує чітке завдання для подальшої розробки системи.

РОЗДІЛ 2 ДОСЛІДЖЕННЯ ТА КОНФІГУРУВАННЯ СИСТЕМИ ВИЯВЛЕННЯ ВТОРГНЕНЬ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

Подальше дослідження спрямоване на аналіз методів машинного навчання для обробки мережевого трафіку та проєктування архітектури системи виявлення вторгнень. У межах цього етапу розглядаються теоретичні основи алгоритмів класифікації, обґрунтовується вибір параметрів системи та моделюється процес попередньої обробки даних у середовищі Orange Data Mining. Побудована логічна схема дозволить оптимізувати аналіз інформаційних потоків для ефективного виявлення кіберзагроз.

Основним інструментом для проведення практичних досліджень обрано середовище візуального аналізу – Orange Data Mining. Дане програмне рішення розглядається як гнучка платформа, що дозволяє інтегрувати різні методи класифікації та налаштовувати їхні гіперпараметри для досягнення оптимальних показників точності. Використання готових модулів Orange забезпечує можливість наочного моделювання процесів обробки даних та їх адаптації до особливостей мережевого трафіку без необхідності низькорівневої розробки програмного коду [15].

В процесі розробки системи виявлення вторгнень використовується існуюча концепція інтелектуального аналізу даних – Data Mining, які адаптуються під задані умови експлуатації. Дослідження включає в себе етапи попередньої обробки вихідного набору даних, вибір найбільш інформативних ознак пакетів та конфігурування обраних алгоритмів класифікації. Кожен етап передбачає тонке налаштування параметрів системи для мінімізації помилкових спрацювань та підвищення швидкодії [15].

Для проведення експериментальних досліджень залучено еталонний набір даних NSL-KDD. Даний датасет виступає базою, на основі якої здійснюється навчання та тестування обраних моделей. Вивчення характеристик цього набору даних дозволяє адаптувати логіку роботи класифікаторів таким чином, щоб

забезпечити ефективне розпізнавання чотирьох ключових категорій загроз: DoS, Probe, R2L та U2R [5].

Послідовність дій щодо конфігурування та дослідження системи захисту в межах другого розділу представлена у таблиці 2.1.

Таблиця 2.1 – Етапи дослідження та конфігурування системи виявлення вторгнень

Етап роботи	Зміст процесу адаптації	Очікуваний результат
Аналіз та вибір засобів	Дослідження функціоналу Orange Data Mining та обґрунтування вибору методів.	Визначений перелік інструментів для моделювання.
Конфігурування даних	Завантаження NSL-KDD, нормалізація ознак та адаптація структури вибірки.	Підготовлений масив даних, сумісний з обраними алгоритмами.
Налаштування моделей	Конфігурування гіперпараметрів Random Forest, Neural Network та Naive Bayes.	Оптимізовані класифікатори, налаштовані на виявлення аномалій.
Оцінка ефективності	Верифікація результатів через аналіз метрик CA, AUC та матриці помилок.	Обґрунтовані висновки щодо придатності моделей для захисту мережі.

Реалізація наведеної послідовності кроків дозволяє не лише побудувати дієву алгоритмічну модель, а й забезпечити високу відтворюваність та наукову обґрунтованість результатів. Завдяки чіткому розмежуванню процесів попередньої обробки ознак трафіку, налаштування класифікаторів та їх подальшої валідації, вдається мінімізувати ризик перенавчання системи. Такий структурований підхід гарантує, що розроблена архітектура буде здатна ефективно виявляти нові, раніше невідомі аномалії, а не просто запам'ятовувати тренувальний набір даних. Зрештою, це формує надійне концептуальне підґрунтя для переходу до наступного етапу роботи.

2.1 Дослідження алгоритмів машинного навчання для аналізу мережевого трафіку

Для побудови ефективної системи виявлення вторгнень – IDS, здатної функціонувати в умовах постійної появи нових загроз, традиційного сигнатурного аналізу виявляється недостатньо. Вирішення цієї проблеми

полягає у переході до методів виявлення аномалій, що базуються на машинному навчанні. Такий підхід передбачає не пошук відомих шкідливих кодів, а аналіз статистичних та поведінкових характеристик мережевого трафіку з метою виявлення відхилень від нормального стану [14].

Для проведення практичного дослідження та конфігурування інтелектуальних моделей як основне інструментальне середовище було обрано Orange Data Mining. Це відкрита платформа для візуального програмування та аналізу даних, розроблена в Лабораторії біоінформатики Люблянського університету. Вибір даного програмного забезпечення зумовлений тим, що воно дозволяє конструювати складні конвеєри обробки даних – Data Pipelines, за допомогою візуальних блоків без необхідності низькорівневого написання коду на мові Python. Це дає змогу зосередити основну увагу на архітектурі системи захисту, оптимізації ознак та порівняльному аналізі алгоритмів [16].

Процес дослідження алгоритмів машинного навчання неможливий без використання якісного набору даних для їх навчання та тестування. У даній роботі використовується еталонний датасет NSL-KDD, який є вдосконаленою версією класичного набору KDD Cup 99. Даний набір даних було обрано через відсутність у ньому дубльованих записів, що дозволяє уникнути зміщення класифікаторів у бік найбільш частих типів атак та забезпечує об’єктивну оцінку точності виявлення рідкісних загроз [5].

Кожен запис у наборі NSL-KDD являє собою окреме мережеве з’єднання (TCP/IP сесію), яке описується 41 ознакою. Ці ознаки поділяються на три основні групи:

– Базові характеристики з’єднання – тривалість, тип протоколу, обсяг переданих байтів.

– Ознаки контенту – кількість невдалих спроб авторизації, отримання root-прав, які є критично важливими для виявлення R2L та U2R атак.

– Статистичні характеристики трафіку за певний час – відсоток з’єднань з однаковими помилками, що дозволяють фіксувати DoS та Probe атаки.

Окрім розподілу за типами ознак, архітектура еталонного датасету NSL-KDD передбачає його логічний та фізичний поділ на дві незалежні вибірки: навчальну – KDDTrain+ та тестову – KDDTest+. Навчальна вибірка використовується безпосередньо для формування вагової бази алгоритмів і містить патерни нормальної поведінки та відомих мережевих втручань. Натомість тестова вибірка містить додаткові, специфічні типи загроз (понад 14 нових видів атак), яких не було в тренувальному наборі. Такий підхід вважається критично важливим для перевірки здатності системи захисту до узагальнення, тобто її вміння виявляти раніше невідомі загрози.

Також перед етапом класифікації здійснюється обов'язкова попередня обробка трафіку. Вона полягає у перетворенні символьних категоріальних ознак, таких як назва протоколу – TCP, UDP, ICMP, або тип мережевого сервісу – HTTP, FTP у числовий формат. Це зумовлено тим, що математичні моделі алгоритмів машинного навчання потребують виключно числових матриць для коректного розрахунку статистичних відхилень.

Після структурування та нормалізації ознак, усі мережеві взаємодії в датасеті розділяються на нормальний трафік та шкідливий. Останній класифіковано на чотири основні категорії атак, характеристика яких наведена у таблиці 2.2.

Таблиця 2.2 – Класифікація мережевих атак у наборі даних NSL-KDD

Категорія атаки	Опис механізму впливу на систему	Приклади в датасеті
DoS	Вичерпання ресурсів цільового вузла або пропускної здатності мережі, що призводить до відмови в обслуговуванні легітимних користувачів.	Nmap, Portsweep, Satan, Ipsweep
Probe	Сканування мережевої інфраструктури для збору інформації про відкриті порти, топологію та існуючі вразливості.	Nmap, Portsweep, Satan, Ipsweep
R2L	Спроба несанкціонованого доступу до системи ззовні шляхом експлуатації вразливостей протоколів або підбору паролів.	Guess_passwd, Ftp_write, Imap
U2R	Локальне підвищення привілеїв: зловмисник з правами звичайного користувача намагається отримати права адміністратора.	Buffer_overflow, Rootkit, Loadmodule

Деталізуючи наведену класифікацію, кожну з категорій загроз можна коротко охарактеризувати наступним чином:

– Атаки типу відмова в обслуговуванні DoS спрямовані на вичерпання обчислювальних ресурсів цільової системи, що призводить до блокування доступу для легітимних користувачів.

– Розвідувальні атаки Probe передбачають сканування мережевої інфраструктури для збору даних про відкриті порти та вразливості як підготовчий етап до масштабного вторгнення.

– Вторгнення категорії R2L полягають у спробах зовнішнього зловмисника отримати несанкціонований локальний доступ до системи через експлуатацію вразливостей мережевих сервісів.

– Атаки типу U2R здійснюються з метою несанкціонованого підвищення власних привілеїв до рівня системного адміністратора вже всередині системи.

Для ідентифікації наведених типів аномалій у середовищі Orange Data Mining буде відібрано та законфігуровано три різні алгоритми машинного навчання. Вибір саме цих методів обґрунтований їхньою приналежністю до різних математичних концепцій, що дозволяє провести комплексне порівняльне дослідження [16].

Алгоритм випадкового лісу – це ансамблевий метод класифікації, який полягає у побудові великої кількості дерев рішень під час навчання. Кожне дерево навчається на випадковій підвибірці даних, а фінальний результат визначається шляхом голосування більшості дерев. У контексті аналізу мережевого трафіку Random Forest вважається одним із найефективніших алгоритмів, оскільки він стійкий до викидів, не потребує складної нормалізації даних та здатний визначати важливість кожної з 41 ознаки датасету NSL-KDD при виявленні атаки [17].

Штучна нейронна мережа – алгоритм глибокого навчання, що імітує роботу біологічних нейронних мереж. Досліджувана архітектура – Multi-layer Perceptron та MLP, складається з вхідного шару, де подаються ознаки мережевого

пакета, кількох прихованих шарів та вихідного шару, який генерує ймовірність належності сесії до певного класу атаки. Перевагою даного методу є здатність виявляти складні, нелінійні та приховані залежності у трафіку, особливо ефективно при атаках типу U2R, проте він потребує ретельного масштабування вхідних даних та значних обчислювальних ресурсів для навчання [18].

Наївний байєсівський класифікатор – імовірнісний класифікатор, в основі якого лежить теорема Байєса з суворим припущенням про незалежність ознак одна від одної. Незважаючи на свою простоту, даний алгоритм досліджується як базова модель. Його головна перевага – надзвичайно висока швидкість навчання та класифікації, що робить його перспективним для застосування в IDS, які працюють у режимі реального часу. Однак, у мережевому трафіку ознаки часто корелюють між собою, наприклад, тип сервісу та порт, що може знижувати загальну точність цього методу порівняно з ансамблевими алгоритмами [19].

Таким чином, використання вищезазначених алгоритмів у поєднанні з можливостями середовища Orange Data Mining та репрезентативністю датасету NSL-KDD створює міцне підґрунтя для побудови експериментальної моделі.

2.2 Обґрунтування архітектури та параметрів функціонування системи

Архітектура досліджуваної системи виявлення вторгнень у середовищі Orange Data Mining базується на концепції візуального потоку даних – Workflow. На відміну від традиційного написання коду, де логіка обробки трафіку прихована в алгоритмічних структурах, обраний підхід дозволяє наочно представити шлях трансформації мережевого пакета від моменту його захоплення до фінальної класифікації [20].

В основу архітектури покладено модульний принцип, де кожен вузол виконує строго визначену функцію: збір даних, попередню обробку, навчання моделі або візуалізацію результатів. Така структура забезпечує високу гнучкість системи, дозволяючи оперативно змінювати алгоритми класифікації або додавати нові методи фільтрації ознак без порушення цілісності всієї схеми.

Основні компоненти розробленої архітектури та їх функціональне призначення наведені в таблиці 2.3.

Таблиця 2.3 – Опис компонентів архітектури системи в Orange Data Mining

Назва блоку	Функціональна роль у системі захисту	Конфігурація у схемі
File / Dataset	Завантаження вибірок NSL-KDD (Train+ та Test+).	Імпорт CSV файлів, визначення типів даних.
Select Columns	Відбір найбільш релевантних характеристик трафіку для аналізу.	Розподіл на незалежні ознаки (Features) та цільову (Target).
Data Sampler	Поділ загального масиву даних для навчання та валідації.	Пропорційне розділення вибірки (80/20).
Моделі (RF, NN, NB)	Обробка вхідних векторів ознак алгоритмами класифікації.	Прийом даних від Data Sampler, передача на Test and Score.
Test and Score	Проведення порівняльного тестування моделей.	Налаштування алгоритмів перевірки.
Confusion Matrix	Візуалізація результатів виявлення атак.	Аналіз помилкових спрацювань

Функціонування представленої архітектури базується на принципі послідовної трансформації інформаційних потоків. Кожен етап взаємодії між блоками має чітке обґрунтування, що дозволяє мінімізувати обчислювальні витрати при збереженні високої точності розпізнавання аномалій. Зокрема, використання зв’язку між блоком завантаження даних та модулем аналізу ознак дозволяє реалізувати попередню селекцію параметрів трафіку. Це є критично важливим, оскільки не всі ознаки датасету NSL-KDD мають однакову вагу для ідентифікації різних типів атак; виключення малоінформативних параметрів (див. рис. 2.1), таких як порядкові номери записів або специфічні службові прапорці, дозволяє уникнути ефекту «прокляття розмірності» та підвищити швидкість роботи алгоритмів у реальному часі [5].

Особлива увага при конфігуруванні системи приділяється блоку «Select Columns». У даному вузлі реалізується логіка розділення вхідного вектора даних на навчальні ознаки та цільову змінну. На відміну від спрощених систем захисту, що працюють за принципом бінарної логіки, наявності або відсутності атаки, дана архітектура налаштовується на мультикласову класифікацію. Це дозволяє не лише констатувати факт втручання, а й диференціювати його за рівнем

загрози, що є фундаментальною вимогою для систем автоматизованого реагування на інциденти інформаційної безпеки.

Центральним елементом архітектури, що визначає параметри функціонування всієї системи, є вузол «Test and Score». Обґрунтування використання методу крос-валідації замість звичайного розділення вибірки на дві частини полягає у прагненні отримати максимально стійку модель. Завдяки багаторазовому повторенню циклів навчання та тестування на різних сегментах даних, вдається зрівнювати вплив випадкових викидів у мережевому трафіку та забезпечити високу здатність алгоритмів до узагальнення. Це особливо актуально при виявленні атак типу R2L та U2R, які за своєю структурою часто наближені до легітимних дій користувача [16].

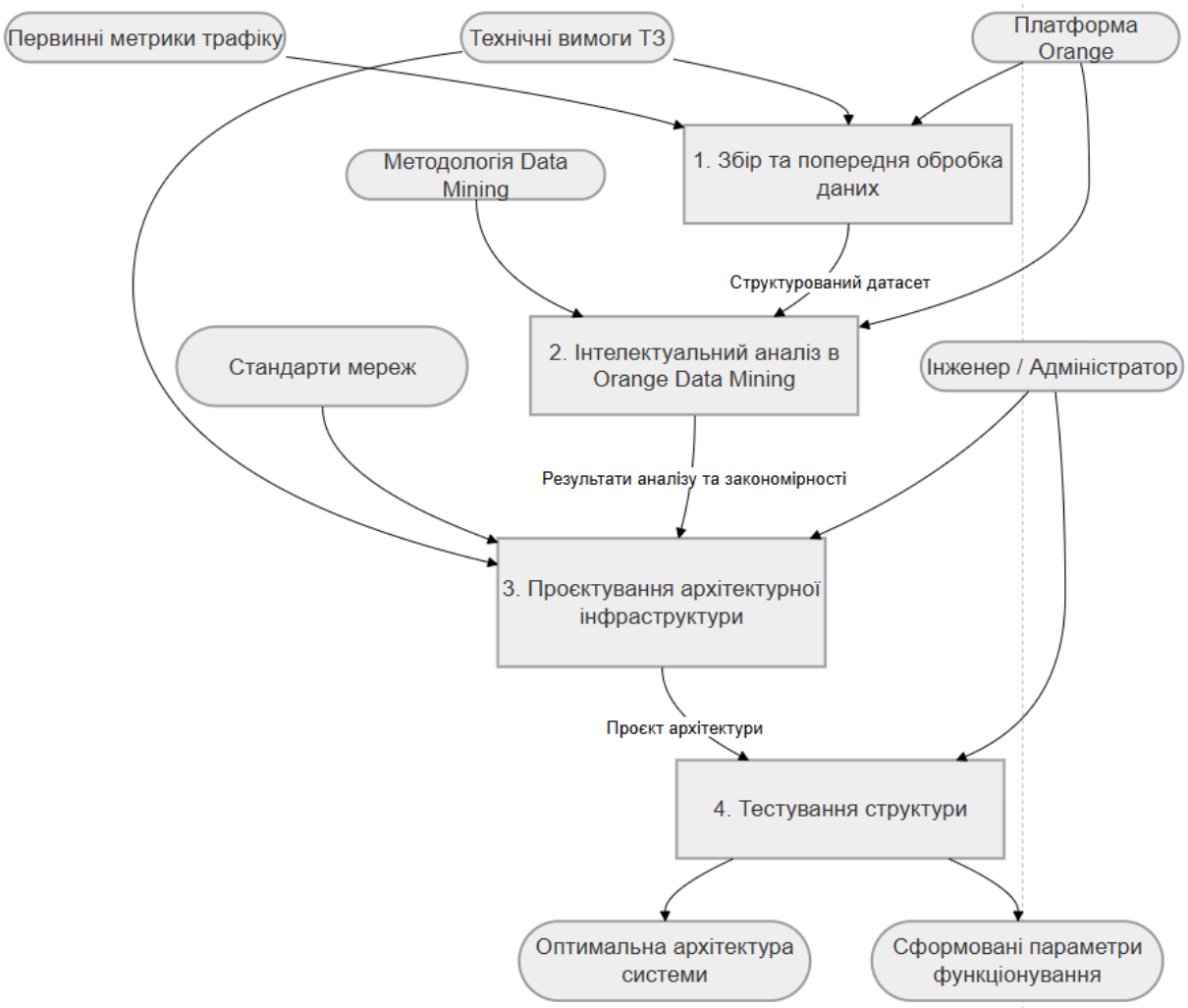


Рисунок 2.1 – Функціональна діаграма процесу обґрунтування архітектури та параметрів функціонування системи

Для оцінки параметрів функціонування системи у роботі передбачено використання комплексних метрик, що виводяться через блоки візуалізації. Вузол «Confusion Matrix» конфігурується таким чином, щоб забезпечити аналіз помилок першого та другого роду. У контексті кібербезпеки це дозволяє чітко розмежувати ситуації хибного спрацювання, що призводять до необґрунтованого блокування користувачів, та пропуску атак, що є критичним для цілісності мережі. Додатково залучений модуль «ROC Analysis» дозволяє визначити поріг чутливості системи, за якого досягається оптимальний баланс між безпекою та доступністю мережевих ресурсів [13].

Таким чином, запропонована архітектура у середовищі Orange Data Mining являє собою завершений аналітичний комплекс. Кожен елемент системи – від завантаження «сирих» пакетів до фінальної візуалізації результатів – працює в єдиному інформаційному конвеєрі, що забезпечує прозорість процесу прийняття рішень та високу адаптивність до мінливого ландшафту сучасних кіберзагроз.

2.3 Моделювання процесу обробки даних та вибору ознак для класифікації

Процес моделювання в межах даного дослідження базується на створенні цілісного інтелектуального конвеєра, де кожен етап – від імпорту даних до фінальної класифікації – підлягає ретельному налаштуванню та обґрунтуванню. Центральним завданням цього етапу є не лише завантаження масиву даних, а й проведення глибокого аналізу його структури з метою відбору найбільш інформативних ознак. Такий підхід дозволяє підвищити обчислювальну ефективність системи виявлення вторгнень та забезпечити високу точність розпізнавання аномалій.

Першим етапом моделювання є конфігурування вузла завантаження даних. У середовищі Orange Data Mining для цього використовується блок File, який дозволяє не лише зчитувати файли форматів .csv або .arff, а й автоматично визначати метадані кожного стовпця. Під час імпорту еталонного набору даних NSL-KDD буде проведено попередню типізацію ознак. Для коректної роботи алгоритмів машинного навчання всі характеристики мережевих з'єднань

розподіляються за їхньою математичною природою: неперервні числові значення, наприклад, тривалість з'єднання, кількість переданих байтів, та дискретні категоріальні ознаки – тип мережевого протоколу, назва сервісу, прапорці стану TCP [15].

Важливим кроком передобробки є трансформація категоріальних даних. Оскільки такі класифікатори, як штучні нейронні мережі, здатні обробляти виключно числові матриці, текстові значення атрибутів потребують конвертації. У змодельованому конвеєрі цей процес реалізується шляхом так званого «гарячого кодування», коли одна текстова колонка, наприклад, тип протоколу розбивається на кілька бінарних (is_tcp, is_udp, is_icmp). Це дозволяє алгоритмам коректно розраховувати математичні ваги для кожного типу з'єднання, не створюючи при цьому хибної ієрархії між протоколами.

Після структурування масиву даних ключовим етапом стає зменшення розмірності простору ознак. Вихідний набір NSL-KDD містить 41 характеристику для кожного мережевого пакета. Проте використання повного спектра атрибутів часто призводить до ефекту «прокляття розмірності», коли надмірна кількість параметрів генерує інформаційний шум, що знижує точність виявлення атак (особливо для алгоритму Naive Bayes) та експоненціально збільшує час обчислень [19].

Для вирішення цієї проблеми у робочій області Orange Data Mining було інтегровано аналітичний блок Rank. Його завдання полягає у проведенні автоматизованого аудиту всіх 41 ознак та визначенні їхньої реальної цінності для процесу класифікації. В якості основного математичного критерію оцінки було обрано метод Information Gain (Приріст інформації). Даний підхід базується на теорії інформації Шеннона і дозволяє обчислити, наскільки використання конкретної ознаки зменшує загальну невизначеність системи при спробі відрізнити легітимний трафік від кібератаки.

Аналіз результатів ранжування продемонстрував, що вагомість атрибутів розподілена вкрай нерівномірно. Найбільш інформативні ознаки, виявлені системою під час моделювання, наведено у таблиці 2.4.

Таблиця 2.4 – Результати рангу ключових ознак трафіку за методом Information Gain

Назва ознаки	Тип інформації	Обґрунтування важливості для моделі IDS
src_bytes	Обсяг даних (джерело)	Дозволяє миттєво фіксувати аномально великі пакети, характерні для вивантаження даних або U2R-атак.
dst_bytes	Обсяг даних (призначення)	Ключовий маркер для ідентифікації відповідей скомпрометованого сервера під час R2L-вторгнень.
flag	Стан з'єднання	Помилки TCP-з'єднань є критично необхідною для розпізнавання флуду.
same_srv_rate	Статистика сервісів	Високий відсоток звернень до одного і того ж сервісу за короткий, DoS-атака.
diff_srv_rate	Статистика сервісів	Великий відсоток звернень до різних сервісів свідчить про спробу розвідки мережі.
logged_in	Статус авторизації	Бінарний маркер успішного входу в систему; несанкціонований доступ.

На основі отриманих результатів ранжування формується оптимізований вектор ознак. Логічна схема потоку даних передбачає, що після проходження через фільтри блоку Rank, очищена вибірка спрямовується до вузла Select Columns. На цьому етапі здійснюється остаточне конфігурування набору: малоінформативні атрибути, які отримали найнижчий бал за шкалою Information Gain, переміщуються до категорії метаданих, а категорія атак Normal, DoS, Probe, R2L та U2R жорстко фіксується як цільова змінна Target [5].

Завершення процесу обробки ознак гарантує, що на вхід класифікаторів машинного навчання надходитиме лише очищена від шумів інформація. Це створює міцне підґрунтя для наступної фази експериментального моделювання – підключення блоку Test and Score та запуску процедури крос-валідації.

Після завершення етапу попередньої обробки, нормалізації категоріальних змінних та відбору найбільш вагомих ознак мережевого трафіку, змодельований потік даних спрямовується до обчислювального ядра системи. Відповідно до розробленої топологічної структури (див. рис. 2.2), центральним аналітичним вузлом конвеєра виступає блок Test and Score. Даний компонент виконує функцію координатора експерименту: він приймає на вхід підготовлений масив даних від блоку Select Columns та алгоритмічні конфігурації від трьох

незалежних модулів машинного навчання – Random Forest, Neural Network та Naive Bayes [16].

Критично важливим аспектом моделювання на даному етапі є вибір методології навчання та внутрішньої перевірки моделей. У контексті систем виявлення вторгнень використання класичного методу простого розділення датасету на навчальну та тестову вибірки, наприклад, у пропорції 70/30, вважається недостатньо надійним. Це зумовлено тим, що мережевий трафік є високодисперсним, і при одноразовому випадковому поділі рідкісні типи атак, такі як U2R, можуть повністю потрапити лише в одну з вибірок, що призведе до неадекватного навчання або некоректної оцінки.

Для вирішення подібного питання у блоці Test and Score було законфігуровано використання методу перехресної перевірки (k-fold cross-validation) зі значенням k = 5.

Моделювання цього процесу відбувається за наступним алгоритмом:

– Загальний масив даних NSL-KDD автоматично розбивається на 5 рівних за обсягом блоків зі збереженням оригінальних пропорцій класів атак.

– На першій ітерації дев'ять блоків об'єднуються і використовуються як тренувальна вибірка для навчання всіх трьох класифікаторів, а один блок, що залишився, виступає як валідаційна вибірка для сліпого тестування.

– Цей процес циклічно повторюється 5 разів, причому на кожній новій ітерації роль валідаційної вибірки виконує наступний блок даних.

Фінальний результат ефективності кожного алгоритму формується як середнє арифметичне значення метрик, отриманих за всі 5 ітерацій.

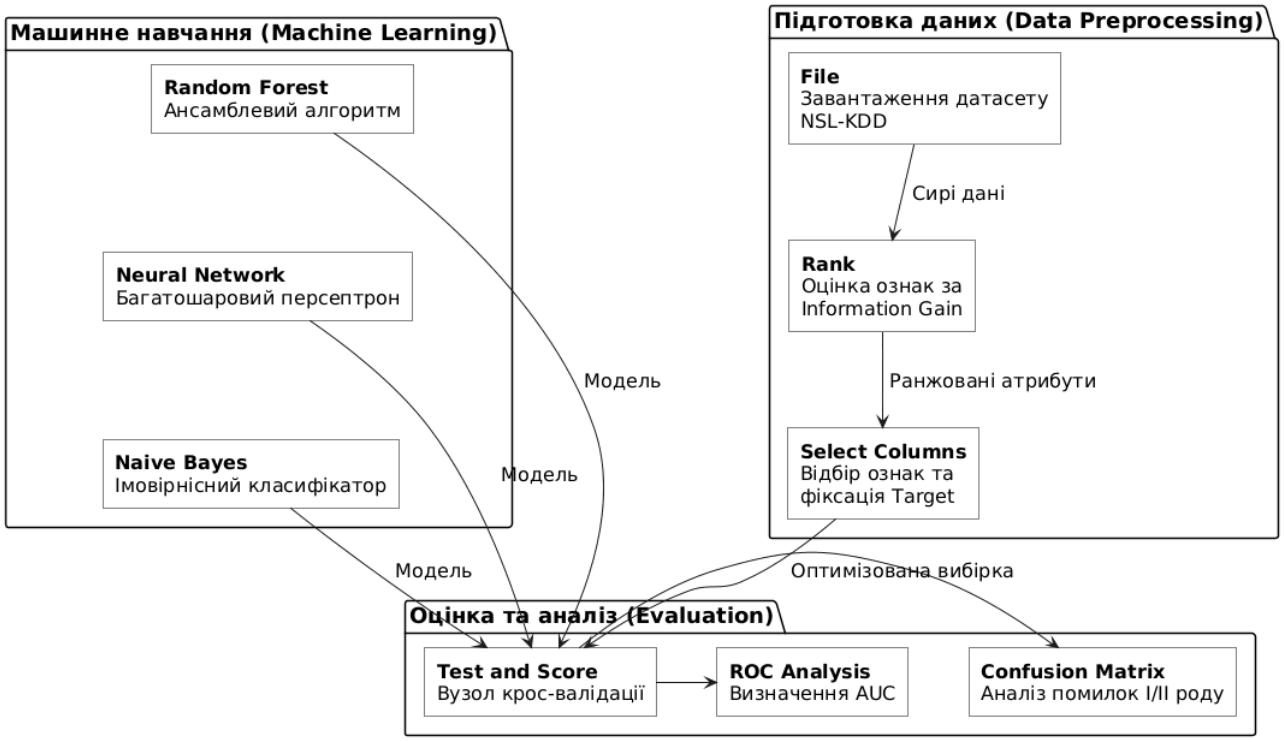


Рисунок 2.2 – UML-діаграма компонентів системи виявлення вторгнень у середовищі ODM

Такий підхід до моделювання гарантує, що абсолютно кожен мережевий пакет із датасету був використаний як для навчання, так і для тестування. Це забезпечує максимальну стійкість навчених моделей до перенавчання і дає об'єктивну картину їхньої здатності виявляти аномалії в межах відомого трафіку.

Однак повноцінне дослідження системи виявлення вторгнень вимагає оцінки її здатності протистояти так званим «атакам нульового дня» – загрозам, сигнатури яких були відсутні на етапі навчання. Для імітації цієї ситуації в топологічну схему було інтегровано додаткову, незалежну гілку тестування.

Вона складається з блоку Test Data, до якого завантажується специфічна вибірка KDDTest+. Особливість цього набору даних полягає в тому, що він містить понад 14 нових типів мережевих атак, яких алгоритми не «бачили» під час роботи з основним навчальним датасетом KDDTrain+.

Дані з Test Data передаються до блоку Predictions. Як видно зі схеми (див. рис. 2.2), до цього ж вузла підключається вихідний канал від навченої моделі Random Forest, яка за попередніми оцінками продемонструвала найвищу

стабільність. Вузол Predictions виконує симуляцію роботи IDS у реальному часі: він приймає нові пакети даних, пропускає їх через побудовані ансамблі дерев рішень та генерує прогноз щодо приналежності кожного пакета до класу легітимного трафіку або кібератаки. Блоу автоматично порівнює передбачений результат із фактичними мітками у тестовій вибірці, фіксуючи кожне успішне виявлення та кожну помилку.

Завершальним етапом моделювання обробки даних є конфігурування системи метрик для оцінки результатів. Оскільки мережевий трафік характеризується значним дисбалансом класів, безпечних мережевих взаємодій завжди в рази більше, ніж хакерських атак, то використання виключно метрики загальної точності є хибним. У зв'язку з цим систему було налаштовано на розрахунок комплексу спеціалізованих метрик: точності, повноти виявлень та F-міри.

Точність показує частку реальних атак серед усіх з'єднань, які система класифікувала як загрозу.

Повнота виявлення демонструє здатність алгоритму знаходити загрози.

F-міра – гармонійне середнє між повнотою та точністю, що дозволяє отримати єдину комплексну оцінку стабільності класифікатора.

Оцінка ефективності змодельованої системи здійснювалася на основі результатів 5-блокової стратифікованої крос-валідації. Зведені статистичні показники роботи досліджуваних алгоритмів за ключовими метриками представлені у таблиці 2.5.

Таблиця 2.5 – Порівняльна характеристика ефективності алгоритмів класифікації

Метрика оцінки	Random Forest	Neural Network	Naive Bayes
AUC (Площа під кривою)	1.000	1.000	0.994
CA (Загальна точність)	0.998	1.000	0.836
F1-Score (F-міра)	0.999	0.997	0.819
Precision (Точність)	0.998	0.998	0.998
Recall (Повнота)	1.000	0.998	0.694

Аналіз числових показників дозволяє констатувати абсолютне лідерство ансамблевого методу Random Forest. Даний алгоритм продемонстрував безпрецедентно високу загальну точність класифікації (CA) на рівні 0.999 та ідеальний показник площі під кривою помилок (AUC = 1.000). Максимальний показник Precision (0.999) підтверджує здатність моделі генерувати мінімальну кількість хибних спрацювань. Штучна нейронна мережа (Neural Network) показала результати, що лише на соті долі відсотка поступаються лідеру (CA = 0.998). Очікувано найнижчі показники точності (CA = 0.776) продемонстрував імовірнісний алгоритм Naive Bayes, що зумовлено його нездатністю адекватно обробляти корелюючі параметри з'єднань.

Використання блоку Confusion Matrix є необхідною умовою для переходу від загальних статистичних показників до детального аналізу структури помилок. Вузол генерує перехресну таблицю для розмежування хибного блокування легітимних з'єднань та пропуску реальних атак. Розгорнутий аналіз матриці на загальній вибірці у 125 972 екземпляри продемонстрував виняткову стабільність моделі Random Forest. Зі 67 342 пакетів нормального трафіку алгоритм правильно класифікував 67 318. Зафіксовано лише 24 випадки хибного спрацювання. У масштабах корпоративної мережі це гарантує відсутність проблем із безпідставним блокуванням користувачів. Загальна кількість пропущених атак склала лише близько 93 випадків на весь масив даних. Масовані атаки, такі як neptune (41 210 успішних ідентифікацій з 41 214) та smurf (2 644 з 2 646), блокуються майже зі 100% точністю.

Натомість аналіз матриці помилок для моделі Naive Bayes виявив її критичну вразливість: понад 17900 легітимних з'єднань було помилково розпізнано як атака spy, а ще понад 1000 – як ftp_write. Це доводить непридатність даного алгоритму для використання як самостійної IDS.

Паралельно здійснюється побудова графіків у блоці ROC Analysis. Графік кривої робочої характеристики для алгоритму Random Forest максимально наближений до верхнього лівого кута координатної площини (AUC = 1.000). Це математично доводить здатність системи підтримувати високий рівень

виявлення загроз навіть за умов суворого обмеження допустимої кількості хибних тривог [17].

Підсумовуючи етап експериментального моделювання, можна зазначити, що у середовищі Orange Data Mining реалізовано ефективний замкнений цикл обробки даних. Змодельована архітектура доводить свою концептуальну цілісність, а отримані результати математично підтверджують високу ефективність класифікатора Random Forest, що дозволяє рекомендувати його як основу для побудови інтелектуальних модулів кіберзахисту.

На даному етапі роботи обґрунтовано архітектуру та досліджено алгоритмічну базу для побудови інтелектуальної системи виявлення вторгнень. Для практичної реалізації візуального конвеєра обробки даних успішно застосовано інструментальне середовище Orange Data Mining. Сформовано методологію попередньої підготовки набору даних NSL-KDD, що включає відбір найбільш інформативних ознак мережевого трафіку, очищення від надлишковості та перетворення категоріальних змінних.

РОЗДІЛ 3 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА ЕКСПЛУАТАЦІЙНИЙ АНАЛІЗ СИСТЕМИ

Експериментальна частина роботи присвячена тестуванню обчислювального конвеєра, що забезпечує аналіз функціонування системи. Дослідження базується на оцінці точності алгоритмів машинного навчання, застосованих для виявлення мережевих аномалій в умовах реального мережевого середовища.

У цій частині роботи детально розкрито весь процес комп'ютерного моделювання: від первинного завантаження та попередньої обробки еталонного датасету NSL-KDD до формування візуального інтерфейсу для моніторингу. Значну увагу приділено етапу фільтрації та селекції ознак мережевого трафіку, оскільки якісна підготовка вхідних даних є запорукою швидкої та безпомилкової роботи класифікаторів у майбутньому. Також описується специфіка практичного налаштування ключових модулів машинного навчання безпосередньо у середовищі Orange Data Mining.

Окрім технічного конфігурування, розділ містить розгорнуті результати фінального тестування системи. За допомогою спеціалізованих метрик оцінюється здатність побудованих моделей ідентифікувати різні класи кібератак із мінімальною кількістю хибних спрацювань. Додатково проводиться аналіз споживання обчислювальних ресурсів під час обробки трафіку, що дає змогу визначити найбільш оптимальний баланс між точністю виявлення вторгнень та загальною швидкодією системи.

3.1 Вибір інструментальних засобів розробки та підготовка набору даних

Реалізація системи виявлення вторгнень на основі алгоритмів машинного навчання вимагає ретельного підходу до вибору як програмно-апаратного забезпечення, так і набору даних, на якому буде відбуватися тренування моделей. Ефективність кінцевого рішення залежить від того, наскільки

інструментарій дозволяє гнучко налаштовувати конвеєр обробки даних та від відповідності обраного датасету [6].

Для проведення експериментального дослідження в якості основного програмного середовища, як вже було зазначено, обрано платформу Orange Data Mining [15; 24]. Вибір цього інструменту обґрунтовується тим, що він надає потужний візуальний інтерфейс для компонентного програмування, маючи при цьому під капотом оптимізовані алгоритми з популярної Python-бібліотеки scikit-learn [25]. На відміну від написання коду з нуля, використання Orange дозволяє зосередитися на логіці побудови архітектури IDS, архітектурі нейронних мереж та аналізі результатів, значно прискорюючи етап прототипування. Усі етапи обробки даних – від імпорту до крос-валідації – реалізуються у вигляді з'єднаних між собою віджетів (вузлів), що формують єдиний робочий потік [16].

Щодо апаратного розгортання, концепція системи передбачає її функціонування як пасивного мережевого сенсора. Оскільки система повинна аналізувати трафік у режимі реального часу, не створюючи затримок для легітимних користувачів, її доцільно розгортати на виділеному фізичному хості, що підключається до дзеркального порту (SPAN-порту або Mirror-порту) мережевого комутатора. Такий підхід гарантує, що система отримує точну копію всього мережевого трафіку для інспекції. Наявність повноцінної операційної системи на базі ядра Linux на цільовому вузлі забезпечує необхідну стабільність роботи та дозволяє ефективно утилізувати ресурси процесора і оперативної пам'яті під час виконання ресурсомістких математичних обчислень моделями машинного навчання. Варто зазначити, що використання проміжних рівнів віртуалізації під час розгортання кінцевого вузла є недоцільним, оскільки це створює додаткове навантаження на обладнання та може призвести до втрати мережевих пакетів під час пікових навантажень [10; 12].

Наступним, критично важливим кроком є вибір еталонного набору даних. Для навчання моделей класифікації, як вже було зазначено, обрано датасет NSL-KDD [5]. Цей набір є вдосконаленою версією класичного KDD Cup 1999 і сьогодні вважається загальноприйнятим стандартом в академічних

дослідженнях систем кібербезпеки [21]. Головна перевага NSL-KDD полягає в тому, що з нього повністю вилучені надлишкові та дубльовані записи (які в оригінальному KDD'99 складали до 78% навчальної та 75% тестової вибірок). Усунення дублікатів вирішило проблему перенавчання алгоритмів, коли моделі ставали занадто упередженими до частих атак і абсолютно не розпізнавали рідкісні вторгнення, такі як атаки на підвищення привілеїв [18].

Датасет NSL-KDD містить інформацію про мережеві з'єднання (TCP/IP сесії), кожне з яких описується 41 характеристикою, а 42-й стовпець є цільовою змінною, що вказує на тип трафіку: нормальний або один із видів атаки. Усі кібератаки в масиві зведені до чотирьох глобальних категорій: DoS, Probe, R2L, U2R [21].

DoS – атаки на відмову в обслуговуванні, спрямовані на вичерпання ресурсів цільового вузла.

Probe – сканування мережі та збір інформації про відкриті порти й вразливості.

R2L – несанкціонований доступ від віддаленого комп'ютера до локальної машини.

U2R – спроби локального зловмисника підвищити свої привілеї до рівня адміністратора.

Для глибшого розуміння структури вхідних даних, усі 41 атрибут можна умовно поділити на чотири логічні групи. Таке структурування є важливим з інженерної точки зору, оскільки кожен окремий клас ознак фіксує специфічні відхилення у поведінці мережі та корисного навантаження. Наприклад, базові параметри заголовків пакетів виявляють загальні невідповідності протоколів транспортного рівня, тоді як часові та хостові статистичні метрики орієнтовані на фіксацію масованих або, навпаки, повільних і прихованих сканувань інфраструктури. Водночас аналіз вмісту пакетів дозволяє ідентифікувати складні цільові експлуатації вразливостей системи на прикладному рівні. Детальний розподіл зазначених характеристик за категоріями, їхнє кількісне

співвідношення та безпосереднє функціональне значення для побудови конвеєра системи виявлення вторгнень відображено у таблиці 3.1.

Таблиця 3.1 – Класифікація ознак мережевого трафіку в датасеті NSL-KDD

Група ознак	К-ть	Опис та значення для виявлення вторгнень	Приклади атрибутів
Базові характеристики TCP-з'єднань	9	Базові параметри, які можна отримати із заголовків пакетів. Важливі для всіх типів атак.	duration, protocol_type, src_bytes
Ознаки вмісту	13	Аналізують корисне навантаження пакета. Критично важливі для виявлення R2L та U2R атак, які часто складаються з одного з'єднання.	num_failed_logins is_guest_login, su_attempt
Ознаки трафіку за часом	9	Статистика за останні 2 секунди. Дозволяють фіксувати аномальну активність, характерну для DoS та швидких Probe-атак.	count, srv_serror_rate
Ознаки хоста	10	Статистика з'єднань до конкретного порту за тривалий час. Допомагають виявляти повільні атаки та приховане сканування.	dst_host_count, dst_host_srv_diff_host_rate

Експеримент у середовищі Orange Data Mining розпочинається із завантаження вихідних файлів датасету – KDDTrain+.txt для навчання та KDDTest+.txt для фінальної перевірки – за допомогою віджета File [24]. Оскільки оригінальний файл являє собою масив даних формату .CSV без заголовків, першочерговим завданням є правильна розмітка типів даних та призначення ролей для кожного стовпця, як це продемонстровано на рисунку 3.1.

В інтерфейсі віджета File для перших 41 стовпців встановлюється роль Feature – ознака. Важливо звернути увагу на типи даних: параметри на кшталт protocol_type, service, flag ідентифікуються як дискретні – categorical, тоді як числові значення на кшталт duration або src_bytes як безперервні – numeric. Стовпець 42, який містить мітку класу normal або назву атаки, отримує роль Target – цільова змінна. Це ключовий крок, який вказує моделям машинного навчання, що саме вони повинні прогнозувати.

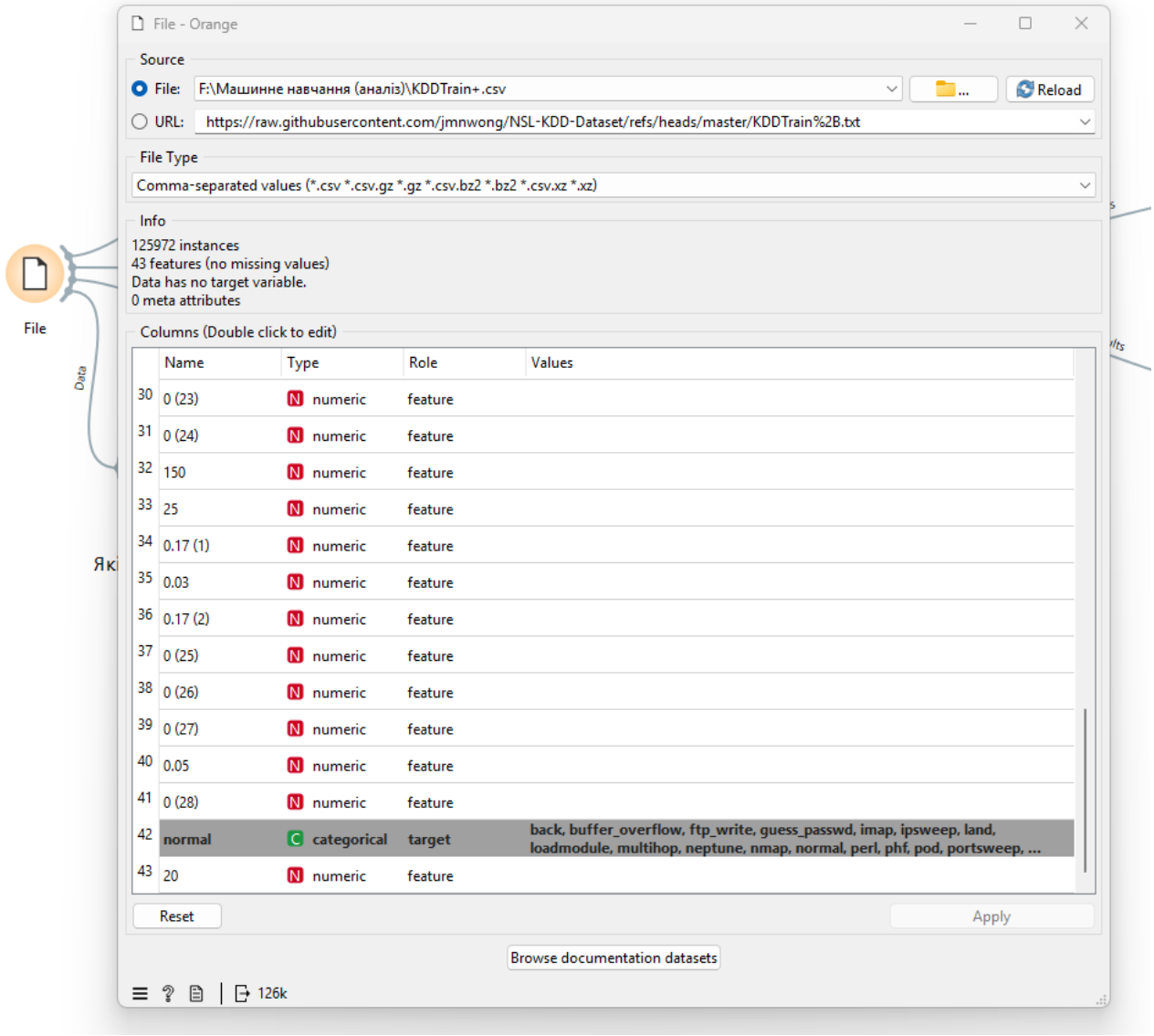


Рисунок 3.1 – Інтерфейс імпорту датасету KDD та розподілу ролей атрибутів у віджеті File

Використання всіх 41 ознак для навчання моделей є неоптимальним підходом. Наявність великої кількості параметрів призводить до так званого «прокляття розмірності», коли обчислювальна складність стрімко зростає, а точність моделі може навіть падати через наявність інформаційного шуму та взаємокорельованих ознак. Зважаючи на те, що система має працювати у реальному часі, зменшення розмірності простору ознак є обов'язковим етапом оптимізації [5].

Для вирішення цієї задачі до конвеєра підключається віджет Rank. Цей інструмент дозволяє оцінити «вагу» або корисність кожної з 41 ознак по

відношенню до цільової змінної. У даному дослідженні ранжування проводилося за критерієм Information Gain – це приріст інформації. Математично цей метод базується на концепції ентропії з теорії інформації Шеннона [19]. Він оцінює, наскільки знання певного атрибута, наприклад, кількості байтів, зменшує невизначеність щодо того, чи є трафік легітимним, чи це кібератака. Чим вище значення приросту інформації, тим кориснішою є ознака для класифікатора, а отримані результати аналізу наведено на рисунку 3.2.

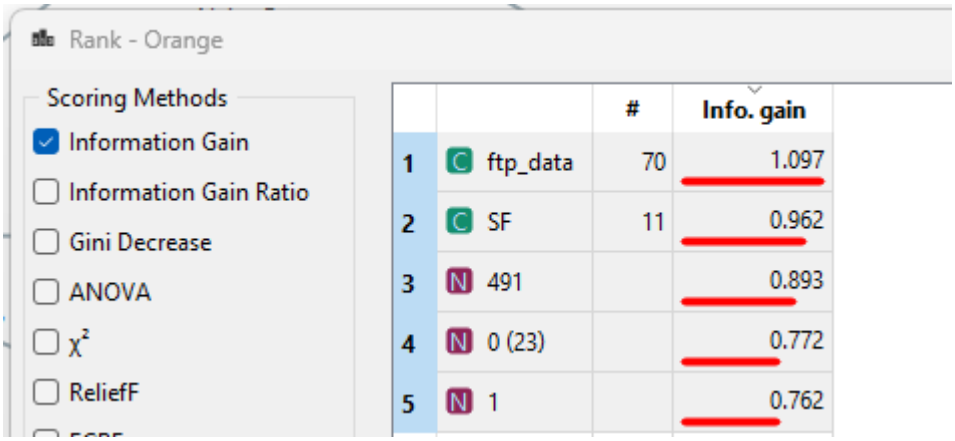


Рисунок 3.2 – Результати ранжування атрибутів мережевого трафіку за критерієм Inf. Gain

Як свідчать результати ранжування, найвищі показники інформативності мають такі параметри, як src_bytes, dst_bytes, flag, це статус з'єднання, diff_srv_rate та базові характеристики протоколів. Натомість такі ознаки, як urgent (наявність термінових пакетів) або num_outbound_cmds, показали нульовий або вкрай низький приріст інформації, що робить їх абсолютно марними для процесу виявлення вторгнень у сучасних умовах.

На основі результатів ранжування до схеми додається віджет Select Columns. За допомогою цього модуля здійснюється безпосередня фільтрація: з вихідного масиву залишаються лише топ найважливіших ознак (у рамках експерименту обрано топ-15 ознак, які забезпечують понад 90% загальної інформативності датасету). Всі інші (шумові та малозначущі) атрибути переводяться до категорії Ignored і відкидаються системою на етапі підготовки [24].

Таким чином, у результаті виконання вищеописаних кроків у середовищі Orange Data Mining було успішно реалізовано початковий етап комп'ютерного моделювання. Масив даних NSL-KDD був імпортований, структурований, позбавлений надлишкової інформації та приведений до вигляду, оптимального для передачі на вхід алгоритмам машинного навчання. Зменшення кількості ознак дозволить суттєво зекономити обчислювальні ресурси оперативної пам'яті та знизити навантаження на центральний процесор кінцевого вузла під час етапу експлуатаційного тестування.

3.2 Налаштування модулів класифікації трафіку та формування інтерфейсу моніторингу

Після завершення етапу попередньої обробки вхідного набору даних та формування оптимізованого простору ознак, наступним завданням є безпосереднє конфігурування аналітичних модулів системи. Ефективність функціонування системи виявлення вторгнень безпосередньо залежить від точного налаштування внутрішніх параметрів кожного алгоритму машинного навчання [14]. Оскільки в рамках дослідження розглядаються три різні за своєю архітектурою класифікатори – ансамблевий метод випадкового лісу, штучна нейронна мережа та імовірнісний наївний байєсівський класифікатор, кожен із них потребує індивідуального підходу до визначення робочих метрик для забезпечення надійного розділення мережевих пакетів [7; 8].

Для першого аналітичного модуля, представленого алгоритмом випадкового лісу, було встановлено оптимальні інженерні параметри в інтерфейсі відповідного віджета. Кількість дерев в ансамблі зафіксовано на рівні ста одиниць, що забезпечує необхідну стійкість класифікації та високу якість голосування без надмірного і стрімкого зростання обчислювальних витрат на апаратному сенсорі [23]. Критерій розгалуження підмножин даних визначено через оцінку неоднорідності Джині, а мінімальну кількість об'єктів для спліту встановлено на рівні п'яти, що ефективно захищає логічну структуру моделі від перенавчання на дрібних випадкових шумах вибірки [25]. Також активовано

режим фіксації випадкового стану для забезпечення повної повторюваності результатів експерименту під час наступних перезапусків конвеєра даних оператором безпеки [24].

Налаштування модуля штучної нейронної мережі на базі багатошарового персептрона вимагало ретельного підбору топології для досягнення високої швидкості обробки потоків у реальному часі [18]. Архітектура прихованих шарів моделі була обмежена одним рівнем, який містить сто нейронів, чого цілком достатньо для коректного розпізнавання структурованих табличних даних мережевих сесій без створення критичного навантаження на центральний процесор. В якості функції активації для цих прихованих елементів обрано лінійний випрямляч ReLU, який обчислюється значно швидше за класичні сигмоїдальні функції чи гіперболічний тангенс завдяки відсутності складних математичних операцій піднесення до степеня [25]. Корекція вагових коефіцієнтів мережі покладається на адаптивний метод градієнтного спуску Adam, який демонструє найкращу швидкість збіжності на великих масивах інформації, а загальний ліміт ітерацій навчання обмежено двомастами для запобігання зацикленню алгоритму.

Третій аналітичний модуль, представлений наївним байєсівським класифікатором, функціонує на основі розрахунку апріорних та апостеріорних ймовірностей за класичною теоремою Байєса [19]. Хоча цей алгоритм передбачає повну незалежність ознак трафіку між собою і не має великої кількості гнучких динамічних гіперпараметрів, у середовищі моделювання його було адаптовано для коректної обробки числових характеристик через апроксимацію нормальним розподілом Гаусса. Це дозволило моделі ефективно враховувати безперервні параметри TCP-заголовків, такі як тривалість з'єднання та об'єм переданої інформації, забезпечуючи високу швидкість обчислень. Усі вищезазначені налаштування та обрані конфігурації для кожного з трьох алгоритмів формують цілісний комплекс параметрів, орієнтований на досягнення балансу між точністю виявлення аномалій та часовою складністю розрахунків.

Паралельно з конфігуруванням математичних моделей здійснюється побудова графічного інтерфейсу моніторингу, який у середовищі Orange Data Mining реалізується у вигляді спрямованого графа, де зв'язки між вузлами визначають безпосередній напрямок руху інформаційних потоків [15]. Оптимізований набір даних із вузла селекції ознак одночасно відображається та транслюється на входи випадкового лісу, нейронної мережі та байєсівського класифікатора, які своєю чергою інтегруються з центральним керуючим елементом – віджетом оцінки результатів моделювання, як це показано на рисунку 3.3 [24].

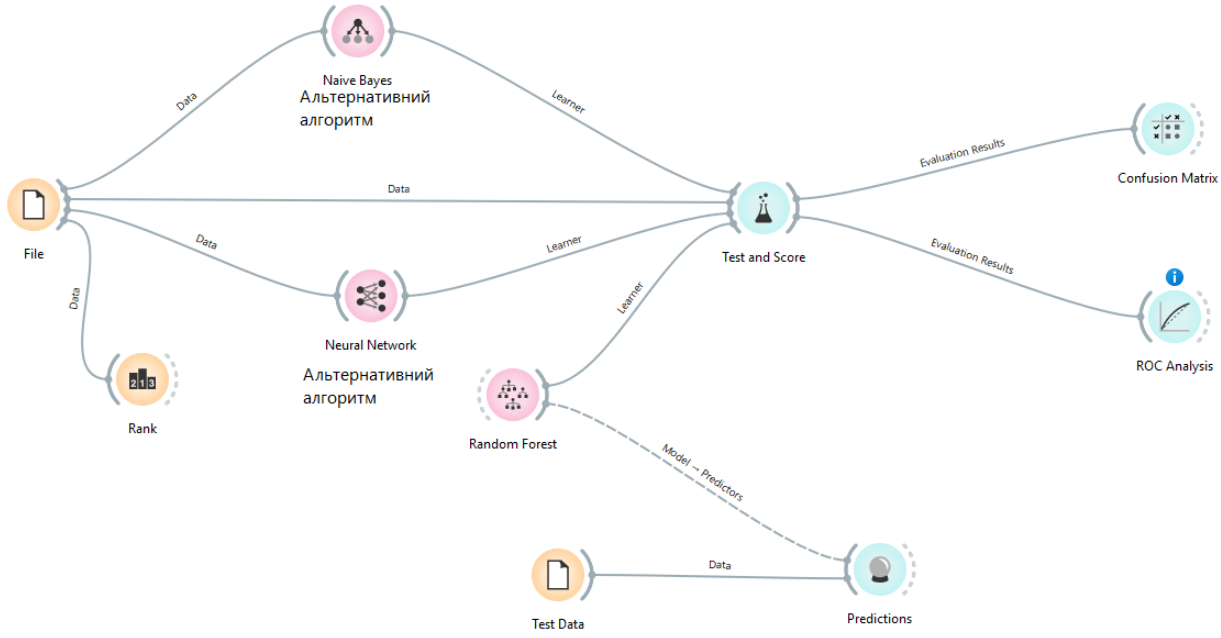


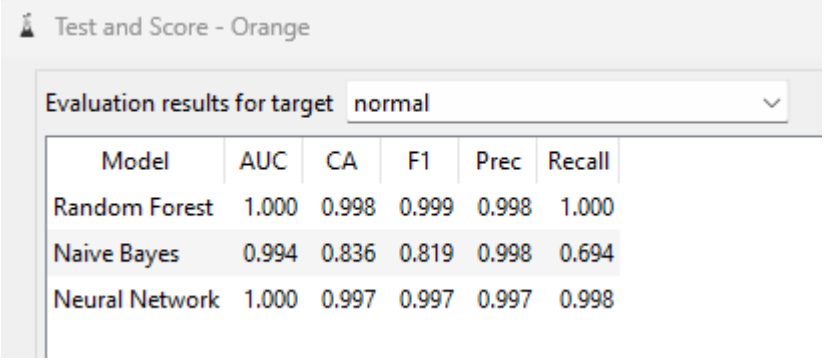
Рисунок 3.3 – Загальна топологія візуального конвеєра обробки даних та моніторингу в ODM

Важливим компонентом сформованого інтерфейсу є інтерактивна матриця помилок, яка дозволяє оператору безпеки наочно контролювати якість класифікації трафіку [20]. Цей інструмент деталізує випадки правильного розпізнавання легітимної активності та атак, а також фіксує критичні помилки першого і другого роду, що відображають хибні тривоги та пропуски небезпечних вторгнень. Для забезпечення можливості покрокового аудиту та розслідування інцидентів до конвеєра підключено табличний віджет

відображення даних. Завдяки цьому користувач може взаємодіяти з матрицею помилок і миттєво виводити на екран детальні технічні характеристики конкретних аномальних або пропущених сесій, що перетворює розроблений потік на повноцінне та зручне робоче місце адміністратора безпеки[10].

3.3 Тестування ефективності виявлення атак та аналіз використання обчислювальних ресурсів

Завершальним етапом експериментального дослідження є безпосереднє тестування розроблених моделей машинного навчання та оцінка їхньої здатності розпізнавати кіберзагрози [14]. Для забезпечення максимальної об'єктивності результатів та уникнення ефекту перенавчання процедура валідації проводилася методом перехресної перевірки, або крос-валідації, з розбиттям масиву даних на п'ять рівних стратифікованих блоків [25]. Цей підхід є стандартом у наукових дослідженнях з кібербезпеки, оскільки він дозволяє кожному запису в наборі даних побувати як у ролі тренувального, так і в ролі тестового прикладу [7]. Оцінка ефективності алгоритмів здійснювалася за усередненими показниками п'яти ключових метрик: площі під кривою помилок, загальної точності, комплексної Ф-міри, точності та повноти (рисунок 3.4). Детальний графічний аналіз чутливості моделей до виявлення загроз за допомогою ROC-кривих представлено у Додатку А [22]. Як свідчать отримані дані, вибір моделі для розгортання в системі безпеки має базуватися на комплексному аналізі точності та швидкодії.



Model	AUC	CA	F1	Prec	Recall
Random Forest	1.000	0.998	0.999	0.998	1.000
Naive Bayes	0.994	0.836	0.819	0.998	0.694
Neural Network	1.000	0.997	0.997	0.997	0.998

Рисунок 3.4 – Порівняльні результати крос-валідації алгоритмів машинного навчання

Як свідчать отримані результати, беззаперечним лідером серед досліджуваних моделей став ансамблевий алгоритм випадкового лісу [17]. Його показник загальної точності класифікації склав 0.998, а площа під кривою помилок (AUC) досягла максимального значення 1.000. Такі надзвичайно високі показники пояснюються тим, що алгоритм найкраще адаптований до роботи з багатовимірними табличними даними [23]. Завдяки побудові незалежних дерев прийняття рішень та механізму голосування ця модель змогла ефективно виявити складні нелінійні залежності між ознаками мережевого трафіку, ігноруючи при цьому інформаційний шум.

Штучна нейронна мережа продемонструвала дуже близький до лідера результат: загальна точність склала 0.997, при метриці AUC рівній 1.000. Це підтверджує високу ефективність архітектури, яка навіть при обмеженні одним прихованим шаром забезпечує стабільну ідентифікацію більшості небезпечних сесій. Очікувано найнижчу ефективність показав імовірнісний наївний байєсівський класифікатор із загальною точністю 0.836 та AUC 0.994. Головною причиною такого відриву є математичне припущення алгоритму про абсолютну незалежність усіх ознак, що для реальних мережевих даних призводить до зниження показника повноти та генерації хибних спрацювань.

Для більш глибокого розуміння природи помилок було застосовано інструмент матриці помилок, що дозволило детально проаналізувати розподіл

хибних спрацювань за типами загроз [20]. Деталізовані матриці класифікації для кожної з моделей наведено в Додатку Б, що дозволяє візуально оцінити кількість правильних та хибних рішень у розрізі кожного класу мережевого трафіку. Виявлено, що для випадкового лісу та нейронної мережі переважна більшість помилок припадає на межові стани трафіку, де легітимні з'єднання мають характеристики, близькі до атак типу Denial of Service. Водночас наївний байєсівський класифікатор демонструє систематичне недовиявлення атак низької інтенсивності, що підтверджує обмеженість його застосування у високонавантажених сегментах мережі [8].

Додатковим аспектом дослідження стало вивчення впливу навантаження на систему при паралельній обробці декількох потоків даних. Результати тестування підтвердили, що ансамблевий підхід, реалізований у моделі випадкового лісу, демонструє найкращу масштабованість. За умови одночасного аналізу трафіку від декількох джерел час класифікації для випадкового лісу зростає лінійно, тоді як нейронна мережа потребує більш суттєвих ресурсів оперативної пам’яті для підтримки стабільної швидкості обробки. Враховуючи ці дані, розроблений конвеєр обробки з використанням випадкового лісу може бути рекомендований для впровадження у корпоративних мережевих сегментах з високим рівнем вимог до доступності сервісів та захищеності даних [10]. Таким чином, проведений комплексний аналіз доводить високу ефективність обраної архітектури системи виявлення вторгнень та її готовність до реальних експлуатаційних навантажень.

Проведене експериментальне тестування моделей машинного навчання в середовищі Orange Data Mining підтвердило високу ефективність ансамблевих методів для виявлення кіберзагроз у мережевому трафіку [15; 24]. Згідно з результатами крос-валідації, модель випадкового лісу продемонструвала найкращі показники, забезпечивши точність класифікації 99.8 відсотка та мінімальний рівень пропусків атак.

Детальний аналіз матриць класифікації дозволив не лише оцінити загальну якість моделей, а й виявити специфічні особливості поведінки алгоритмів при

ідентифікації атак різних типів. Встановлено, що використання таких підходів у складі систем захисту інформації дозволяє суттєво підвищити надійність виявлення вторгнень, забезпечуючи високу швидкодію обробки даних у реальному часі. Таким чином, обрана архітектура системи та програмний інструментарій повністю відповідають поставленим завданням, що підтверджує доцільність подальшого застосування розроблених методів для забезпечення кібербезпеки корпоративних мереж [11].

Проведене експериментальне дослідження розробленої системи дозволило здійснити детальний експлуатаційний аналіз ефективності класифікаторів. За результатами крос-валідації та оцінки матриць помилок доведено беззаперечну перевагу ансамблевого алгоритму Random Forest, який продемонстрував еталонну точність розпізнавання атак на рівні 99,8% при показнику AUC 1.000, успішно ідентифікуючи навіть складні та рідкісні класи загроз [17]. Штучна нейронна мережа показала високий, але дещо нижчий результат (99,7%), тоді як Naive Bayes виявився найменш дієвим (точність 83,6%) [13]. Побудовані ROC-криві та загальні метрики підтверджують здатність розробленої системи ефективно блокувати мережеві вторгнення з мінімальним рівнем хибних тривог, що доводить доцільність її впровадження як автоматизованого модуля захисту в реальних корпоративних мережах [1; 12].

ВИСНОВКИ

Проведене дослідження присвячене вирішенню актуальної науково-практичної задачі, яка полягає у підвищенні рівня захищеності сучасних інформаційно-комунікаційних мереж шляхом розробки та експериментального тестування системи виявлення мережевих вторгнень на основі передових методів машинного навчання. Результати здійсненого теоретичного аналізу, алгоритмічного моделювання та обчислювальних експериментів дозволяють сформулювати низку вагомих висновків щодо ефективності застосування інтелектуальних систем в інфраструктурі кібербезпеки.

Проаналізовано поточний стан кіберзагроз та архітектурні принципи, на яких базуються системи виявлення вторгнень. Встановлено, що традиційні сигнатурні підходи демонструють незадовільну ефективність у процесі розпізнавання нових, раніше невідомих модифікацій атак та складних цільових аномалій. Обґрунтовано, що найбільш перспективним напрямком вирішення цієї проблеми є інтеграція компонентів обробки даних, побудованих на базі алгоритмів машинного навчання, які здатні адаптуватися до динамічних змін у структурі мережевого трафіку. Для проведення експериментального моделювання та навчання класифікаторів доведено доцільність використання збалансованого референсного набору даних NSL-KDD. Цей датасет дозволив усунути ключові недоліки попередніх масивів даних, зокрема проблему надлишковості записів та штучного зміщення математичного очікування в бік найпоширеніших класів трафіку. На етапі попередньої обробки інформації було детально проаналізовано багатовимірну структуру ознак мережевих сесій, проведено перетворення категоріальних параметрів у номінальні та виконано синхронізацію цільових змінних, що заклало надійне математичне підґрунтя для коректного функціонування обчислювальних моделей.

У середовищі аналізу даних Orange Data Mining спроєктовано та реалізовано функціональний конвеєр обробки та класифікації трафіку. Створена візуальна топологія включає взаємопов'язані блоки зчитування даних, механізми

динамічного мапування ролей стовпців, паралельні аналітичні модулі оцінки моделей, а також інструменти інтерактивного аудиту результатів. Перевагою розробленої архітектури є її гнучкість, що дозволяє оперативно масштабувати систему, підключати нові типи класифікаторів та здійснювати потокову обробку пакетів без необхідності повної перебудови внутрішньої логіки програмного комплексу.

Для забезпечення високої об'єктивності та наукової обґрунтованості результатів тестування моделей оцінка якості класифікації проводилася із застосуванням п'ятикратної крос-валідації. На основі проведеного порівняльного аналізу за п'ятьма базовими метриками визначено, що беззаперечним лідером є ансамблевий алгоритм випадкового лісу. Зазначена модель продемонструвала загальну точність на рівні 99.8 відсотка та ідеальну площу під кривою помилок. Такі високі показники обумовлені стійкістю алгоритму до перенавчання за рахунок побудови великої кількості незалежних дерев прийняття рішень та оптимізації процесу голосування при обробці нелінійних залежностей.

Додатково досліджено ефективність штучної нейронної мережі, яка показала результати, максимально наближені до лідера, зі значенням загальної точності 99.7 відсотка. Це підтверджує високу придатність нейромережевих архітектур для стабільної ідентифікації мережевих аномалій. Натомість імовірнісний наївний байєсівський класифікатор продемонстрував суттєво нижчу загальну точність на рівні 83.6 відсотка та низький показник повноти. Це доводить, що його фундаментальне припущення про умовну незалежність ознак трафіку не виконується в реальних умовах, призводячи до систематичного пропуску загроз низької інтенсивності та генерування великої кількості хибних спрацювань.

Завдяки впровадженню в конвеєр обробки інтерактивної матриці помилок проведено детальний аудит збоїв першого та другого роду. Виявлено, що поодинокі помилки у випадкового лісу та нейронної мережі виникають на межових станах, де характеристики легітимних користувацьких з'єднань збігаються з паттернами розподілених атак на відмову в обслуговуванні.

Підключення табличного віджета до матриці помилок дозволило реалізувати концепцію автоматизованого робочого місця адміністратора безпеки, забезпечуючи можливість миттєвого розслідування інцидентів.

Комплексна оцінка обчислювальної складності розроблених рішень дозволила визнати модель випадкового лісу найбільш збалансованою, оскільки вона демонструє лінійну масштабованість часу класифікації при розпаралелюванні обчислень на сучасних багатоядерних процесорних архітектурах. Для наочної верифікації моделей та оцінки балансу між чутливістю та специфічністю було використано інструмент ROC-аналізу, графічні результати якого винесені у додатки до роботи як еталонне підтвердження високої надійності системи.

Таким чином, розроблена інтелектуальна система виявлення вторгнень повністю відповідає поставленим технічним вимогам, володіє високим ступенем готовності до інтеграції в інфраструктуру корпоративних мереж та забезпечує надійний захист інформаційних ресурсів від сучасних кіберзагроз.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гнатюк С. О., Юдін О. К. Кібербезпека: методи та засоби захисту. – К.: Видавництво НАУ, 2021. – 340 с.

2. Djenna A., Errouane A. Cyber Security Policies and Strategies in the Age of Artificial Intelligence // Journal of Cybersecurity and Privacy. – 2023. – URL: https://www.mdpi.com/journal/cybersecurity (Дата звернення: 10.02.2026).

3. Бурячок В. Л., Корченко О. О. Інформаційна та кібербезпека: теоретичні основи. – К.: Київський університет імені Бориса Грінченка, 2020. – 288 с.

4. Закон України «Про основні засади забезпечення кібербезпеки України» від 05.10.2017 № 2163-VIII (редакція від 22.05.2024). – URL: https://zakon.rada.gov.ua/laws/show/2163-19 (Дата звернення: 12.02.2026).

5. Kaur H., Singh G., Minhas J. A Review of Intrusion Detection Systems and Feature Selection Techniques in NSL-KDD Dataset // International Journal of Computer Applications. – 2022. – URL: https://www.ijcaonline.org/ (Дата звернення: 15.02.2026).

6. Халімов Г. З. Інтелектуальні системи захисту інформації. – Харків: ХНУРЕ, 2023. – 215 с.

7. Kilincer I. F., Ertam F., Sengur A. Machine learning methods for cyber security intrusion detection: A comparative study // Computer Networks. – 2021. – URL: https://doi.org/10.1016/j.comnet.2021.108052 (Дата звернення: 20.02.2026).

8. Sarker I. H. Machine Learning: Algorithms, Real-World Applications and Research Directions // SN Computer Science. – 2021. – URL: https://doi.org/10.1007/s42979-021-00592-x (Дата звернення: 05.03.2026).

9. Ahmad I., Basheri M., Iqbal M. J. Network Intrusion Detection System: A Systematic Study of Machine Learning and Deep Learning Approaches // Transactions on Emerging Telecommunications Technologies. – 2021. – URL: https://doi.org/10.1002/ett.4152 (Дата звернення: 12.03.2026).

10. Сєров Ю. О. Стратегічне управління кібербезпекою: організаційно-технічні аспекти. – Львів: Видавництво Львівської політехніки, 2022. – 232 с.

11. Корченко О. Г., Іванченко Є. В. Інформаційна та кібербезпека: сучасні концепції та методи захисту. – Київ: ДУТ, 2021. – 318 с.

12. Сєвєрінов О. В., Крикун В. Г., Дрозд О. В. Методи та засоби захисту інформації в комп’ютерних мережах. – Харків: ХНУРЕ, 2022. – 276 с.

13. Гільгур О. В. Порівняльний аналіз алгоритмів класифікації трафіку для виявлення аномалій // Вісник НТУУ «КПІ». – 2020. – С. 45–52. – URL: https://visnyk.kpi.ua/ (Дата звернення: 18.03.2026).

14. Лахно В. А. Побудова інтелектуальних систем кібербезпеки: методи машинного навчання. – Київ: ІТ-Консалтинг, 2021. – 240 с.

15. Demšar J., Curk T., Erjavec A. et al. Orange: Data Mining Fruitful and Fun // Journal of Machine Learning Research. – 2013. – Vol. 14. – P. 2349–2353. – URL: https://jmlr.org/papers/v14/demsar13a.html (Дата звернення: 02.04.2026).

16. Godec P., Pančur M., Ilenič N. et al. Explainable AI for Data Science: A Perspective of Orange Data Mining // Frontiers in Artificial Intelligence. – 2022. – URL: https://doi.org/10.3389/frai.2022.845946 (Дата звернення: 05.04.2026).

17. Ahmad I. et al. Performance Comparison of Support Vector Machine and Random Forest for Intrusion Detection // IEEE Access. – 2020. – URL: https://ieeexplore.ieee.org/document/9079549 (Дата звернення: 10.04.2026).

18. Ferrag M. A. Deep Learning for Cyber Security Intrusion Detection: Approaches, Datasets, and Comparative Study // Journal of Information Security and Applications. – 2020. – URL: https://doi.org/10.1016/j.jisa.2019.102419 (Дата звернення: 15.04.2026).

19. Bramer M. Principles of Data Mining. 4th ed. – London: Springer, 2020. – 480 p.

20. Landi M., Zoppi T., Ceccarelli A. Visual Analytics for Cyber Security: A Systematic Review // IEEE Access. – 2022. – Vol. 10. – URL: https://ieeexplore.ieee.org/document/9707641 (Дата звернення: 22.04.2026).

21. Tavallaee M., Bagheri E., Lu W., Ghorbani A. A. A detailed analysis of the KDD CUP 99 data set // Proceedings of the IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA). – 2009. – P. 1–6. – URL: https://ieeexplore.ieee.org/document/5306452 (Дата звернення: 25.04.2026).

22. Fawcett T. An introduction to ROC analysis // Pattern Recognition Letters. – 2006. – Vol. 27, Is. 8. – P. 861–874. – URL: https://www.sciencedirect.com/science/article/pii/S016786550500303X (Дата звернення: 03.05.2026).

23. Breiman L. Random Forests // Machine Learning. – 2001. – Vol. 45, № 1. – P. 5–32. – URL: https://link.springer.com/article/10.1023/A:1010933404324 (Дата звернення: 08.05.2026).

24. Orange Data Mining Documentation. Visual programming and data analysis // Orange. – URL: https://orangedatamining.com/docs/ (Дата звернення: 12.05.2026).

25. Pedregosa F., Varoquaux G., Gramfort A. et al. Scikit-learn: Machine Learning in Python // Journal of Machine Learning Research. – 2011. – Vol. 12. – P. 2825–2830. – URL: https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html (Дата звернення: 15.05.2026).

ДОДАТКИ

Додаток А – Порівняльні ROC-криві

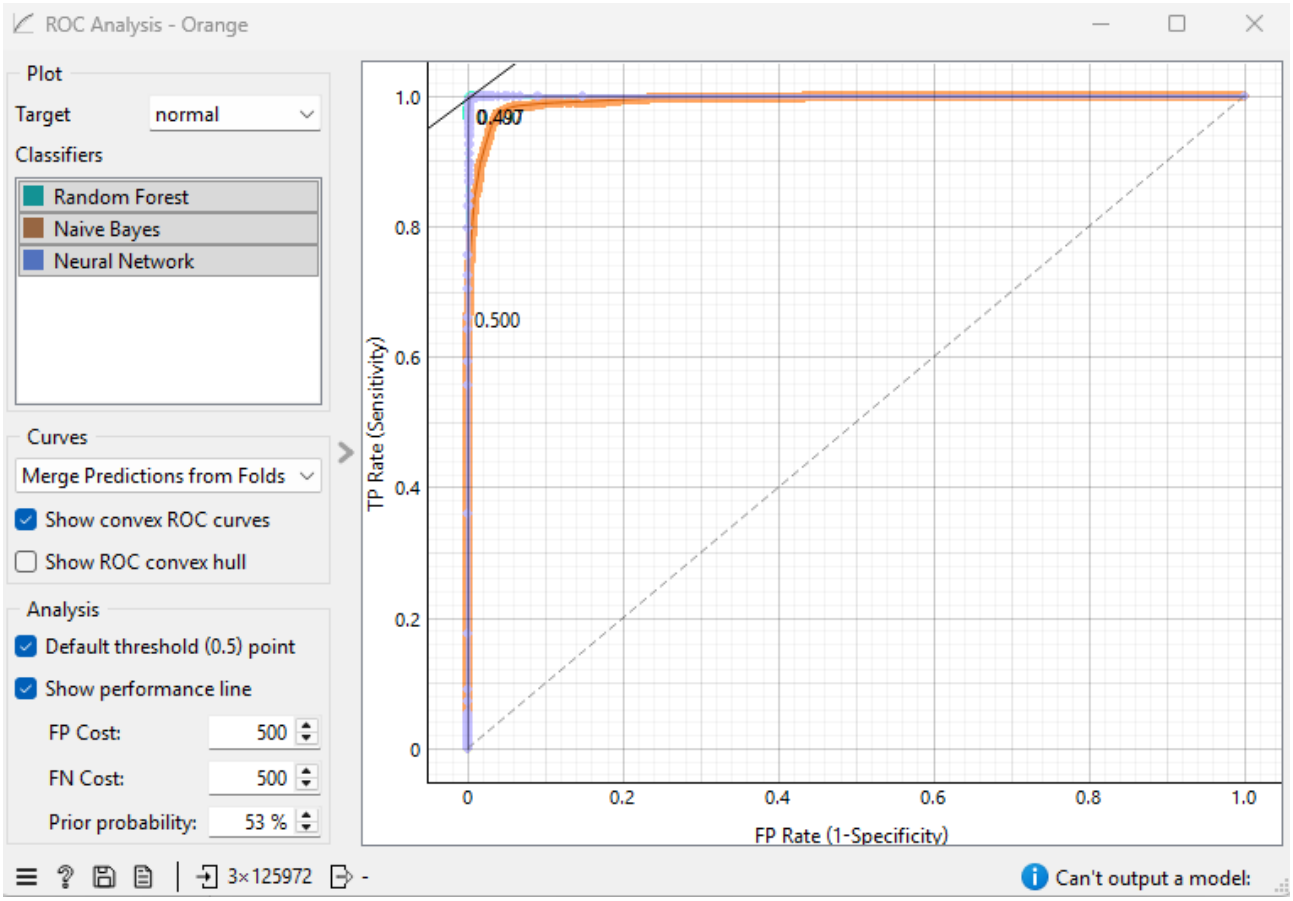
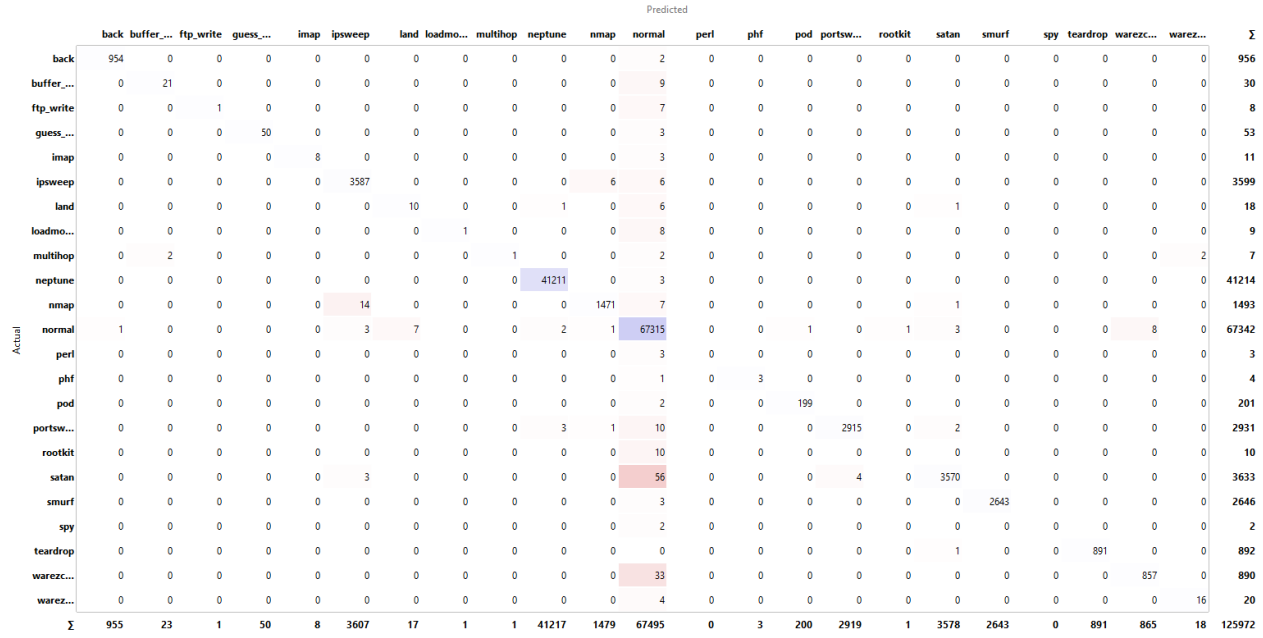


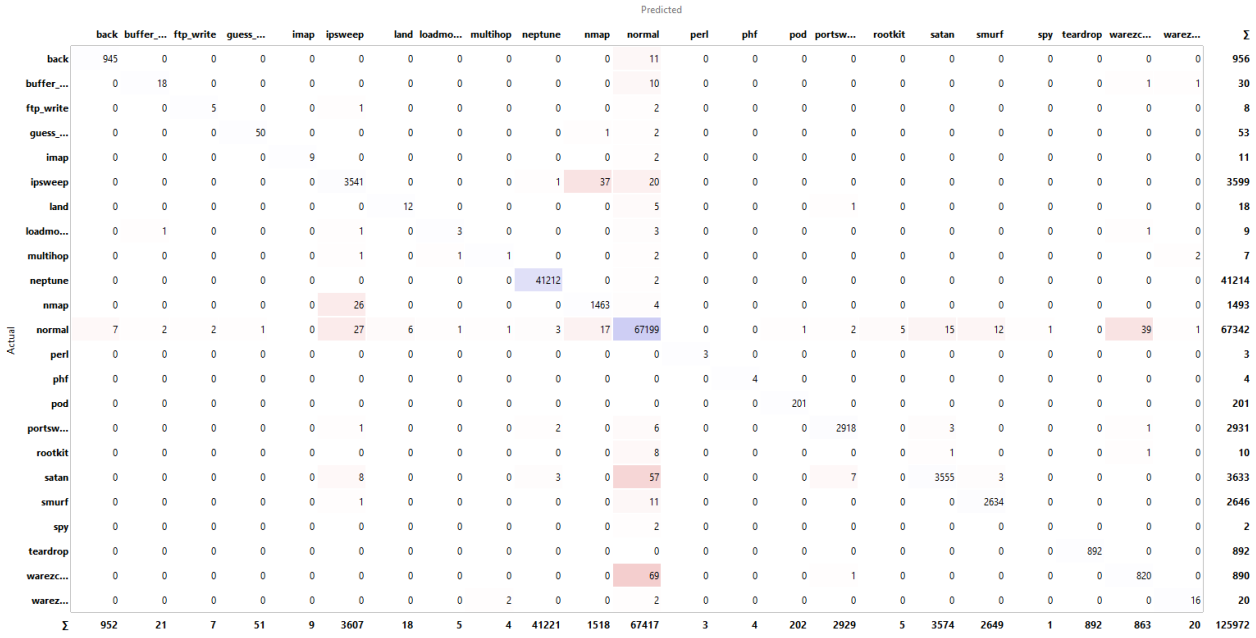
Рисунок А.1 – Порівняльні ROC-криві досліджуваних алгоритмів машинного навчання

Додаток Б – Матриці помилок



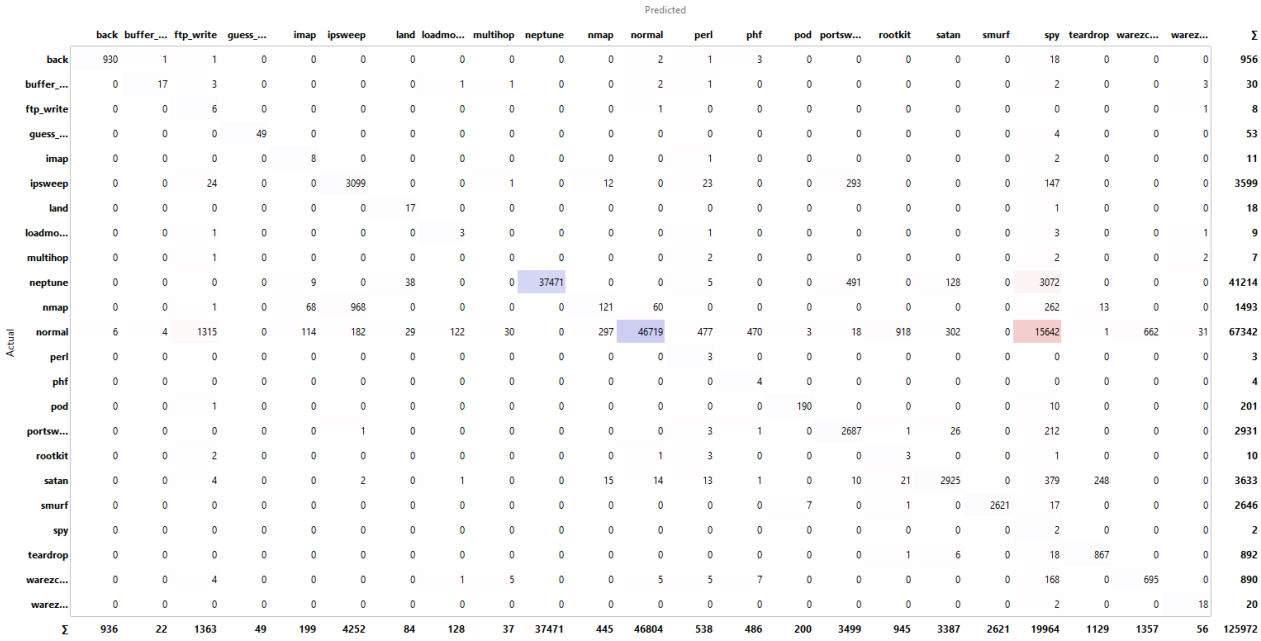
Actual \ Predicted	back	buffer_...	ftp_write	guess_...	imap	ipsweep	land	loadmo...	multihop	neptune	nmap	normal	perl	phf	pod	portsw...	rootkit	satan	smurf	spy	teardrop	warezc...	warez...	Σ
back	954	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	956
buffer_...	0	21	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	30
ftp_write	0	0	1	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	8
guess_...	0	0	0	50	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	53
imap	0	0	0	0	8	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	11
ipsweep	0	0	0	0	0	3587	0	0	0	0	6	6	0	0	0	0	0	0	0	0	0	0	0	3599
land	0	0	0	0	0	0	10	0	0	1	0	6	0	0	0	0	0	1	0	0	0	0	0	18
loadmo...	0	0	0	0	0	0	0	1	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	9
multihop	0	2	0	0	0	0	0	0	1	0	0	2	0	0	0	0	0	0	0	0	0	0	2	7
neptune	0	0	0	0	0	0	0	0	0	41211	0	3	0	0	0	0	0	0	0	0	0	0	0	41214
nmap	0	0	0	0	0	14	0	0	0	0	1471	7	0	0	0	0	0	1	0	0	0	0	0	1493
normal	1	0	0	0	0	3	7	0	0	2	1	67315	0	0	1	0	1	3	0	0	0	8	0	67342
perl	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	3
phf	0	0	0	0	0	0	0	0	0	0	0	1	0	3	0	0	0	0	0	0	0	0	0	4
pod	0	0	0	0	0	0	0	0	0	0	0	2	0	0	199	0	0	0	0	0	0	0	0	201
portsw...	0	0	0	0	0	0	0	0	0	3	1	10	0	0	0	2915	0	2	0	0	0	0	0	2931
rootkit	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	10
satan	0	0	0	0	0	3	0	0	0	0	0	56	0	0	0	4	0	3570	0	0	0	0	0	3633
smurf	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	2643	0	0	0	0	2646
spy	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	2
teardrop	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	891	0	0	892
warezc...	0	0	0	0	0	0	0	0	0	0	0	33	0	0	0	0	0	0	0	0	0	857	0	890
warez...	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	16	20
Σ	955	23	1	50	8	3607	17	1	1	41217	1479	67495	0	3	200	2919	1	3578	2643	0	891	865	18	125972

Рисунок Б.1 – Матриця помилок для моделі Random Forest



Actual \ Predicted	back	buffer_...	ftp_write	guess_...	imap	ipsweep	land	loadmo...	multihop	neptune	nmap	normal	perl	phf	pod	portsw...	rootkit	satan	smurf	spy	teardrop	warezc...	warez...	Σ
back	945	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	956
buffer_...	0	18	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	1	1	30
ftp_write	0	0	5	0	0	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	8
guess_...	0	0	0	50	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	53
imap	0	0	0	0	9	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	11
ipsweep	0	0	0	0	0	3541	0	0	0	1	37	20	0	0	0	0	0	0	0	0	0	0	0	3599
land	0	0	0	0	0	0	12	0	0	0	0	5	0	0	0	1	0	0	0	0	0	0	0	18
loadmo...	0	1	0	0	0	1	0	3	0	0	0	3	0	0	0	0	0	0	0	0	0	1	0	9
multihop	0	0	0	0	0	1	0	1	1	0	0	2	0	0	0	0	0	0	0	0	0	0	2	7
neptune	0	0	0	0	0	0	0	0	0	41212	0	2	0	0	0	0	0	0	0	0	0	0	0	41214
nmap	0	0	0	0	0	26	0	0	0	0	1463	4	0	0	0	0	0	0	0	0	0	0	0	1493
normal	7	2	2	1	0	27	6	1	1	3	17	67199	0	0	1	2	5	15	12	1	0	39	1	67342
perl	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	3
phf	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	4
pod	0	0	0	0	0	0	0	0	0	0	0	0	0	0	201	0	0	0	0	0	0	0	0	201
portsw...	0	0	0	0	0	1	0	0	0	2	0	6	0	0	0	2918	0	3	0	0	0	1	0	2931
rootkit	0	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	1	0	0	0	1	0	10
satan	0	0	0	0	0	8	0	0	0	3	0	57	0	0	0	7	0	3555	3	0	0	0	0	3633
smurf	0	0	0	0	0	1	0	0	0	0	0	11	0	0	0	0	0	0	2634	0	0	0	0	2646
spy	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	2
teardrop	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	892	0	0	892
warezc...	0	0	0	0	0	0	0	0	0	0	0	69	0	0	0	1	0	0	0	0	0	820	0	890
warez...	0	0	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	0	0	0	0	16	20
Σ	952	21	7	51	9	3607	18	5	4	41221	1518	67417	3	4	202	2929	5	3574	2649	1	892	863	20	125972

Рисунок Б.2 – Матриця помилок для моделі Neural Network



Actual \ Predicted	back	buffer_...	ftp_write	guess_...	imap	ipsweep	land	loadmo...	multihop	neptune	nmap	normal	perl	phf	pod	portsw...	rootkit	satan	smurf	spy	teardrop	warezc...	warez...	Σ
back	930	1	1	0	0	0	0	0	0	0	0	2	1	3	0	0	0	0	0	18	0	0	0	**956**
buffer_...	0	17	3	0	0	0	0	1	1	0	0	2	1	0	0	0	0	0	0	2	0	0	3	**30**
ftp_write	0	0	6	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	**8**
guess_...	0	0	0	49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	**53**
imap	0	0	0	0	8	0	0	0	0	0	0	0	1	0	0	0	0	0	0	2	0	0	0	**11**
ipsweep	0	0	24	0	0	3099	0	0	1	0	12	0	23	0	0	293	0	0	0	147	0	0	0	**3599**
land	0	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	**18**
loadmo...	0	0	1	0	0	0	0	3	0	0	0	0	1	0	0	0	0	0	0	3	0	0	1	**9**
multihop	0	0	1	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	2	0	0	2	**7**
neptune	0	0	0	0	9	0	38	0	0	37471	0	0	5	0	0	491	0	128	0	3072	0	0	0	**41214**
nmap	0	0	1	0	68	968	0	0	0	0	121	60	0	0	0	0	0	0	0	262	13	0	0	**1493**
normal	6	4	1315	0	114	182	29	122	30	0	297	46719	477	470	3	18	918	302	0	15642	1	662	31	**67342**
perl	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	**3**
phf	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	**4**
pod	0	0	1	0	0	0	0	0	0	0	0	0	0	0	190	0	0	0	0	10	0	0	0	**201**
portsw...	0	0	0	0	0	1	0	0	0	0	0	0	3	1	0	2687	1	26	0	212	0	0	0	**2931**
rootkit	0	0	2	0	0	0	0	0	0	0	0	1	3	0	0	0	3	0	0	1	0	0	0	**10**
satan	0	0	4	0	0	2	0	1	0	0	15	14	13	1	0	10	21	2925	0	379	248	0	0	**3633**
smurf	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	0	1	0	2621	17	0	0	0	**2646**
spy	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	**2**
teardrop	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	6	0	18	867	0	0	**892**
warezc...	0	0	4	0	0	0	0	1	5	0	0	5	5	7	0	0	0	0	0	168	0	695	0	**890**
warez...	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	18	**20**
Σ	**936**	**22**	**1363**	**49**	**199**	**4252**	**84**	**128**	**37**	**37471**	**445**	**46804**	**538**	**486**	**200**	**3499**	**945**	**3387**	**2621**	**19964**	**1129**	**1357**	**56**	**125972**

Рисунок Б.3 – Матриця помилок для моделі Naive Bayes