DOI: 10.58254/viti.8.2025.06.79

УДК 004.8+004.49

канд. техн. наук Залужний О. В. ORCID: 0000-0002-8722-4087 (ВІТІ ім. Героїв Крут)

ДОСЛІДЖЕННЯ ВПЛИВУ МЕТОДІВ ВІДБОРУ ОЗНАК НА ЕФЕКТИВНІСТЬ МОДЕЛЕЙ МАШИННОГО НАВЧАННЯ ДЛЯ ВИЯВЛЕННЯ КІБЕРАТАК

Об'єми даних, що використовуються для навчання моделей машинного навчання систем кіберзахисту, часто вимірюються в гігабайтах та містять десятки ознак, які можуть приймати сотні тисяч значень, окрім того, дані постійно оновлюються та поповнюються в процесі функціонування систем. Збільшення об'єму даних призводить до виникнення певних проблем, серед яких збільшення часу на навчання та тестування моделі, прокляття вимірності, малі вибірки, зашумлені або надмірні ознаки, а також упереджені дані. Відбір ознак (Feature Selection) є фундаментальним для розв'язання таких проблем.

Метою статті є дослідження впливу методів відбору ознак на якість класифікації кібератак моделями машинного навчання для визначення найбільш ефективних підходів до формування набору ознак.

У статті наведено короткий опис та проведено дослідження ефективності використання відомих методів відбору ознак таких як: кореляційний метод, метод на основі статистичного критерію χ^2, статистичний метод ANOVA, метод на основі розрахунку взаємної інформації, метод ReliefF, метод на основі генетичного алгоритму та метод рекурсивного відбору ознак.

Оцінка ефективності методів відбору ознак здійснювалась в поєднанні з різними методами машинного навчання за такими критеріями, як повнота, точність, точність класифікації та параметром F2-scor. Окрім того, в статті наведені графіки залежності кількості пропущених атак і хибних спрацювань від кількості ознак для кожного із досліджуваних методів.

Експериментальні дослідження проведено з використанням інструментів Python та бібліотеки scikit-learn. Вони показали, що застосування методів відбору ознак покращує показники якості роботи моделей машинного навчання та зменшити час на їх навчання. Ефективність кожного із методів залежить від об'єму даних, який потрібно опрацювати, методів машинного навчання з якими використовуються ті чи інші методи відбору ознак, часових та обчислювальних обмежень. Використання алгоритму з χ^2, методів ANOVA та Mutual Information Filter в моделі, що побудована на основі випадкового лісу, дозволяє отримати кращі результати ніж генетичний алгоритм та рекурсивне вилучення ознак. Проте, останні у поєднанні з методом k найближчих сусідів є найефективнішими з усіх досліджуваних комбінацій методів відбору ознак та методів машинного навчання за кількістю детектованих атак та хибних спрацювань.

Ключові слова: *кібербезпека, виявлення кібератак, методи машинного навчання, методи відбору ознак, зменшення розмірності, кореляційний метод відбору ознак, статистичні методи відбору ознак, генетичний алгоритм, рекурсивного вилучення ознак.*

O. Zaluzhnyi. Study on the impact of feature selection methods on the performance of machine learning models in cyberattack detection

Data volumes used to train machine learning models for cyber security systems are often measured in gigabytes and contain dozens of features that can take hundreds of thousands of values. Also, the data is constantly updated and replenished during the operation of the systems. The increase in data volume leads to certain problems, including increased model training and testing time, the curse of dimensionality, small sample sizes, noisy or redundant features, and biased data. Feature selection is fundamental to solving such kinds of problems.

The purpose of this article is to study the impact of feature selection methods on the quality of cyberattack classification by machine learning models in order to determine the most effective approaches to form a features set.

The article provides a brief description and studies the effectiveness of using well-known feature selection methods such as: the correlation method, the method based on the χ^2 statistical criterion, the ANOVA statistical method, the method based on mutual information calculation, the ReliefF method, the method based on a genetic algorithm, and the recursive feature selection method.

The effectiveness of feature selection methods was evaluated in combination with various machine learning methods according to criteria such as Recall, Precision, Accuracy, and F2-score. In addition, the article provides graphs showing the dependence of the missed attacks number and false positives on the number of features for each method.

Experimental studies were conducted using Python and the scikit-learn library. They showed that the feature selection methods' usage improves the machine learning models performance and reduces the time required for their training. The effectiveness of each method depends on the data amount to be processed, the machine learning methods with which certain feature selection methods are used, and time and computational constraints. Using the χ^2 algorithm, ANOVA methods, and Mutual Information Filter in a model based on random forest allows for better results than Genetic Algorithm and Recursive feature elimination. However, the latter, in combination with the k-Nearest Neighbours method, are the most effective of all the

combinations of feature selection methods and machine learning methods studied in terms of the number of detected attacks and false positives.

Keywords: *cybersecurity, cyberattack detection, machine learning methods, feature selection methods, dimensionality reduction, correlation feature selection method, statistical feature selection methods, genetic algorithm, recursive feature elimination.*

Постановка проблеми

Сучасні тенденції розвитку систем кіберзахисту пов'язані з широким застосуванням технологій машинного навчання (МН) для виявлення і запобігання кіберзагрозам. Об'єми даних, що використовуються для навчання моделей часто вимірюються в гігабайтах та містять десятки ознак (Features), які приймають сотні тисяч значень, окрім того дані постійно оновлюються та поповнюються в процесі функціонування систем кіберзахисту. Збільшення об'єму даних призводить до виникнення певних проблем, серед яких збільшення часу на навчання та тестування моделі, прокляття вимірності, малі вибірки, зашумлені або надмірні ознаки, а також упереджені дані. Відбір ознак (Feature Selection) є фундаментальним для розв'язання таких проблем, оскільки відомо, що дозволяє покращити часові показники роботи моделей МН, підвищити повноту та точності прийнятих ними рішень.

Аналіз останніх досліджень і публікацій

Застосування методів машинного навчання в системах кіберзахисту зумовило значний прогрес у сфері виявлення і запобігання кіберзагрозам. Опубліковані численні наукові статті, в яких досліджуються різні підходи та алгоритми для підвищення ефективності таких систем. Зокрема, у роботі [1] проаналізовано результати провідних досліджень, що присвячені виявленню програм-вимагачів за допомогою алгоритмів машинного навчання та глибокого навчання, де зокрема значна увага приділяється підходам до відбору релевантних ознак.

У роботі [2] було проведено ґрунтовні дослідження моделей машинного навчання для виявлення шкідливого програмного забезпечення (ШПЗ), що побудовані з використанням різних методів машинного навчання. Автори підкреслили важливість відбору релевантних ознак для підвищення ефективності детектування ШПЗ, проте не проводили аналізу різних методів відбору ознак.

У статті [3] запропоновано метод відбору ознак на основі ансамблю із чотирьох методів: Information Gain (IG - інформаційний виграш), Gain Ratio (відносний інформаційний виграш), хі-квадрат та алгоритм ReliefF. Досліджено його ефективність для детектування атак типу "розподілена відмова в обслуговуванні" (DDoS). Для аналізу продуктивності системи було використано набір даних NSL-KDD в якому зменшено кількість ознак із 41 до 13 та навчено модель машинного навчання, де використано алгоритм класифікації який побудований на основі дерева рішень.

У роботі [4] запропоновано новий підхід до класифікації шкідливого програмного забезпечення за допомогою методу k-найближчих сусідів (k-Nearest Neighbors, KNN) в поєднанні з середовищем Cuckoo (відокремлене середовище, "пісочниця", для динамічного аналізу та спостереження за поведінкою неперевіреного або потенційно шкідливого коду без ризику для цілісності основної системи). Для автоматизованого відбору ознак запропоновано використання алгоритму Boruta, який побудований на основі алгоритму класифікації випадковий ліс.

Дослідження, що проведені в [5], свідчать про ефективність використання для відбору ознак, при багатокласовій класифікації, методів що ґрунтуються на взаємній інформації.

У роботі [6] автори застосували генетичний алгоритм (Genetic Algorithm, GA) у поєднанні з обгорткою логістичної регресії для відбору ознак у наборах даних UNSW-N15 та KDDCup99, що використовуються при виявленні мережевих вторгнень. Результати досліджень показали, що у поєднанні з класифікатором дерево рішень такий підхід дозволяє правильно виявляти 81,42 % мережевих вторгнень, при цьому хибні спрацювання становлять 6,39 % (було відібрано 20 ознак з 42 у датасеті UNSW-NB15). Що до датасету KDDCup99, то

такий підхід показав результат виявлення 99,90 % і хибні спрацювання склали 0,105 % для 18 ознак.

В роботі [7] застосовано алгоритм Extreme Gradient Boosting (XGBoost) для відбору релевантних ознак в наборі даних UNSWNB15. Використовуючи зменшений простір ознак було навчено та протестовано моделі машинного навчання, що побудовані на основі методу опорних векторів (support vector machine, SVM), KNN, логістичної регресії, штучної нейронної мережі та дерева рішень. У експериментах було розглянуто як бінарну, так і багатокласову класифікацію. Результати показали, що метод відбору ознак на основі XGBoost дозволяє підвищити точність моделі DecisionTree з 88,13 % до 90,85 % для бінарної класифікації.

Таким чином, аналіз літератури показує, що в сучасних системах кіберзахисту активно використовуються технології машинного навчання. Продуктивність таких систем значною мірою залежить від обраних методів машинного навчання та методів відбору релевантних ознак. Незважаючи на те, що за цією тематикою постійно публікуються нові наукові праці, недостатньо дослідженим залишається напрямок, який пов'язаний із проведенням досліджень ефективності методів відбору ознак в поєднанні з різними методами машинного навчання для розв'язання задач класифікації кібератак.

Метою статті є дослідження впливу методів відбору ознак на якість класифікації кібератак моделями машинного навчання для визначення найбільш ефективних підходів до формування набору ознак.

Виклад основного матеріалу дослідження

Короткий опис датасету

Для проведення досліджень було використано загальнодоступний датасет "Internet of Things network intrusion dataset 20" (IoTID20) [8]. Набір даних IoTID20 сфокусовано на виявленні аномальної активності в мережах IoT, проте в деяких роботах його також використовують для розробки Intrusion Detection System (IDS) [9]. Набір даних IoTID20 складається з 83 ознак, що описують мережеву активність, і трьох ознак, які використовуються для маркування класів або типів подій (мітки). Розподіл міток на бінарні, категорійні та підкатегорійні наведено в таблиці 1[8].

Таблиця 1

Бінарні, категорійні та підкатегорійні мітки датасету IoTID20

Binary	Category	Subcategory
Normal, Anomaly	Normal, DoS, Mirai, MITM, Scan	Normal, Syn Flooding, Brute Force, HTTP Flooding, UDP Flooding ARP Spoofing Host Port, OS

Розподіл зразків у наборі даних IoTID20, який використовується для задач виявлення вторгнень, наведено в таблиці 2 відповідно до трьох рівнів класифікації: бінарного (нормальний трафік/атака), категоріального (основні типи атак), а також підкатегоріального (конкретні підтипи атак).

Таблиця 2

Розподіл безпечних та зловмисних зразків у наборі даних IoTID20

Binary label distribution		Subcategory distribution	
Normal	40073	Type	Instances
Anomaly	585710	Normal	40073
		DoS	59391
Category label distribution		Mirai Ack Flooding	55124

Binary label distribution		Subcategory distribution	
Type	Instances	Mirai Brute force	121181
Normal	40073	Mirai HTTP Flooding	55818
DoS	59391	Mirai UDP Flooding	183554
Mirai	415677	MITM	35377
MITM	35377	Scan Host Port	22192
Scan	75265	Scan Port OS	5307

Короткий опис методів машинного навчання, що використовуються для досліджень

Для тестування ефективності методів відбору ознак були використані відомі методи машинного навчання, а саме: дерево рішень (Decision Tree, DT), випадковий ліс (Random Forest, RF), SVM та KNN.

Метод DT

Структурно проста модель, у якій кожен вузол, зверху вниз, відповідає етапу прийняття рішення. Вибір рішення у вузді здійснюється на основі певних умов наприклад, значень ознаки вибірки під час прогнозування. Кожен кінцевий вузол відповідає певній мітці (класу) до якої веде відповідний шлях у дереві рішень [10]. Алгоритм рекурсивно формує дерево зверху вниз, виконуючи розбиття простору ознак у кожному вузлі на основі певної функції домішок (наприклад, ентропії), оцінюючи при цьому, наскільки кожна ознака сприяє розрізненню класів. Розподіл триває до тих пір, доки це дозволяє підвищити точність розмежування класів або поки у вузлі не залишиться надто мало прикладів для подальшого розбиття. Коли процес завершується, кожен кінцевий вузол відповідає певному класу, який визначається більшістю прикладів, що потрапили до цього вузла під час навчання.

Метод

Random Forest – це ансамблевий алгоритм, який використовує концепцію "колективної мудрості" з метою зменшення дисперсії результатів, що отримані за допомогою окремих класифікаторів. Алгоритм формує сукупність дерев рішень, кожне з яких навчається на різних підвибірках навчальних даних, сформованих методом бутстрепу. Для кожного окремого випадку процес класифікації здійснюється кожним деревом у лісі. А потім прогнозований клас визначається на основі принципу більшості голосів [10].

Метод SVM

Основний принцип SVM полягає у побудові гіперплощини або набору гіперплощин в *n*-вимірному просторі, що розділяють об'єкти за класами з максимально можливою відстанню від гіперплощин до найближчих об'єктів кожного класу. Коли дані не є лінійно роздільними у методі використовують функцію втрат в поєднанні з підходом з м'яким рішенням та L2-регуляризацією [10]. Обчислення класифікатора еквівалентні мінімізації виразу:

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1-y_i(\vec{\omega}\cdot\vec{x}_i-b)\right)\right]+\lambda\|\vec{\omega}\|^2,$$

де $\vec{\omega}$ – це вектор нормалі до гіперплощини;

$\frac{b}{\|\vec{\omega}\|}$ – зсув площини від початку координат вздовж $\vec{\omega}$;

n – загальна кількість навчальних зразків усіх класів датасету;

x_i – вектор ознак *i*-го зразка;

y_i – його клас, що визначає, по який бік від гіперплощини знаходиться цей зразок ($y_i \in \{-1, +1\}$);

λ – коефіцієнт регуляризації (контролює баланс між точністю та узагальненням);

$\|\vec{\omega}\|^2$ – квадрат норми вектора нормалі до гіперплощини.

Метод KNN

Метод KNN є непараметричним алгоритмом, який не обмежений фіксованим числом параметрів. Зазвичай вважають, що цей алгоритм взагалі не має параметрів, а реалізує просту функцію від навчальних даних. Насправді в ньому навіть немає справжнього етапу навчання. Просто на етапі тестування, бажаючи отримати вихід *y* для тестового зразка *x*, шукають в навчальному наборі $\boldsymbol{X}$ k-найближчих сусідів $\boldsymbol{x}$ та визначають його клас на основі більшості їхніх міток. Ця ідея працює для будь-якого виду навчання з учителем, за умови, що можна визначити поняття середньої мітки. У разі класифікації усереднювати можна за унітарними кодовими векторами $\boldsymbol{c}$, у яких $\boldsymbol{c}_y = 1$ і $\boldsymbol{c}_i = 0$ для всіх інших *i*. Середнє за такими векторами можна інтерпретувати, як розподіл імовірності класів. Алгоритм KNN, будучи непараметричним, може досягати дуже високої ємності. Припустімо, наприклад, що є завдання багатокласової класифікації та міра продуктивності з бінарною функцією втрат. У такому разі метод одного найближчого сусіда сходиться до подвоєної байєсівської частоти помилок, коли кількість навчальних прикладів наближається до нескінченності. Перевищення над байєсівською помилкою пов'язане з тим, що випадковим чином вибираємо одного з рівновіддалених сусідів. Якщо кількість навчальних прикладів нескінченна, то у всіх тестових точок $\boldsymbol{x}$ буде нескінченно багато сусідів із навчального набору на нульовій відстані. Якби алгоритм дозволяв сусідам проголосувати, а не вибирав випадкового сусіда, то процедура сходилася б до байєсівської частоти помилок. Висока ємність алгоритму KNN дає змогу отримати гарні результати, якщо є великий навчальний набір. Але за це доводиться розплачуватися високою тривалістю обчислень, а за малого навчального набору алгоритм погано узагальнюється. Одне зі слабких місць алгоритму KNN – невміння зрозуміти, що одна ознака є більше характерною, ніж інша [11].

Короткий опис методів відбору ознак, що використовуються

У роботі досліджувались найбільш поширені фільтраційні методи відбору ознак (англ. filter methods), зокрема: кореляційний метод (Correlation Feature Selection, CFS), метод на основі статистичного критерію χ^2, статистичний метод ANOVA (Analysis of Variance – аналіз дисперсії), метод на основі розрахунку взаємної інформації (Mutual Information Filter, MIF) та метод ReliefF. Серед методів обгортки (wrapper methods) були розглянуті: генетичний алгоритм (Genetic Algorithm, GA), де базовим алгоритмом є дерево рішень **(**DT-GA**)**, а також метод рекурсивного відбору ознак **(**Recursive feature elimination, RFE**)**, у якому базовим алгоритмом також є дерево рішень **(**DT-RFE).

Метод CFS

CFS – це простий алгоритм фільтрації, який ранжує підмножини ознак відповідно до кореляційної евристичної функції оцінки. Функція оцінки зміщена в бік підмножин, що містять ознаки, які сильно корелюють з класом і не корелюють між собою. Нерелевантні ознаки ігноруються, оскільки вони матимуть низьку кореляцію з класом. Надлишкові ознаки слід відсіяти, оскільки вони будуть сильно корелювати з однією або кількома ознаками, що залишилися. Прийнятність ознаки буде залежати від того, наскільки вона прогнозує класи в областях простору зразків, які ще не були передбачені іншими ознаками. Евристична функція оцінки підмножини ознак [12]:

$$\text{Merit}_S = \frac{k \cdot \overline{r_{cf}}}{\sqrt{k + k(k-1) \cdot \overline{r_{ff}}}},$$

де Merit_S – це евристична оцінка підмножини ознак *S*, що містить *k* ознак;

$\overline{r_{cf}}$ – середня кореляція ознака-клас $f \in S$;

$\overline{r_{ff}}$ – середня взаємна кореляції між ознаками в підмножині.

Основна мета – знайти баланс між високою кореляцією ознак з класом, і низькою взаємною кореляцією ознак між собою (тобто мінімізувати надмірність).

Фільтраційний метод на основі статистичного критерію χ^2.

В цьому методі тест χ^2 використовується для оцінки кореляції між ознаками та мітками. Статистичний тест χ^2 має нульову гіпотезу, тобто ознака та мітка є незалежним, проти альтернативної гіпотези, тобто вони є залежними. Нульова гіпотеза відхиляється, коли $\text{p}\left(\chi^2_{df} > \chi^2_{statistic}\right) < \alpha$, де рівень значущості $\alpha = 0{,}05$. В іншому випадку нульова гіпотеза не може бути відхилена [13]. Статистика χ^2 має ступінь вільності *degree of freedom – df*), який розраховується як $df = (r-1)\cdot(c-1)$, де *r* – кількість рядків у контингентній таблиці (ознак), *c* – кількість стовпців (міток) .

Основна ідея тесту χ^2 полягає в тому, щоб порівняти спостережувані значення в даних з теоретично очікуваними значеннями і перевірити, чи пов'язані ці значення між собою. Для розрахунку значення χ^2 створюється контингентна таблиця. Формула статистики χ^2 наступна [13]:

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(Q_{i,j} - E_{i,j})^2}{E_{i,j}},$$

де $Q_{i,j}$ – спостережувана частота для чарунки i, j;

$E_{i,j}$ – очікуване значення для чарунки i, j, розраховане за умови незалежності змінних.

Теоретично очікуване значення для чарунки i, j, розраховується як:

$$E_{i,j} = \frac{\sum_{k=1}^{c} O_{i,k} \cdot \sum_{k=1}^{r} O_{k,j}}{N},$$

де $\sum_{k=1}^{c} O_{i,k}$ – сума по рядку *i* (загальна кількість спостережень в категорії *i* змінної *x*);

$\sum_{k=1}^{r} O_{k,j}$ – сума по стовпцю *j* (загальна кількість спостережень в категорії *j* змінної *y*);

N – загальна кількість усіх спостережень.

У дослідженнях проводився розрахунок χ^2 та визначення ймовірності $p\left(\chi^2_{df} > \chi^2_{statistic}\right)$, на основі якої ухвалюється рішення про відхилення або прийняття нульової гіпотези. Після чого здійснювалось ранжування ознак та відбір 15 ознак з найкращими показниками.

Статистичний метод ANOVA

Тест дисперсійного аналізу ANOVA використовується для порівняння середніх значень декількох груп у наборі даних та виявлення статистично значущих відмінностей між середніми значеннями різних груп (класів). Статистика для ANOVA – F-статистика, яка обчислюється за такими етапами [14]:

1. Розрахунок варіації між групами:

1.1. Між сумою квадратів (Between sum of squares, BSS):

$$\mathrm{BSS} = \sum_{j=1}^{k} n_j (\overline{x_j} - \bar{x})^2 ,$$

де $\overline{x_j}$ – середнє значення j-ї групи;

n_j – кількість спостережень у j-й групі;

$\bar{x}$ – загальне середнє значення всіх спостережень.

1.2. Середні квадрати між групами (Between mean squares, BMS):

$$\mathrm{BMS} = \mathrm{BSS}/df ,$$

де $df = degreeoff\ \ freedom = k - 1$ (k – кількість груп).

2. Розрахунок варіації всередині груп:

2.1. Сума квадратів всередині груп (Within sum of squares, WSS):

$$\mathrm{WSS} = \sum_{j=1}^{k} (n_j - 1)\sigma_j^2 ,$$

де σ – середньоквадратичне відхилення.

2.2. Середні квадрати всередині груп (Within mean squares, WMS):

$$\mathrm{WMS} = \mathrm{WSS}/df_w ,$$

де $df_w = N - k$, N – кількість спостережень.

Після обчислення F-статистики ознаки сортуються та обирається k найкращих ознак (з найбільшими значеннями F-статистики) [15].

Метод на основі розрахунку взаємної інформації

Маючи взаємну інформацію, можна кількісно оцінити релевантність ознаки на основі того, скільки інформації вона містить про цільовий клас. Таким чином, ми можемо ранжувати ознаки відповідно до отриманої релевантності. Mutual Information Filter дає зважені результати. Для розрахунку взаємної інформації використовується відомий підхід із теорії інформації, який полягає в тому, щоб визначити скільки інформації одна випадкова змінна має про іншу. Обчислення здійснюють за таким виразом [10]:

$$\mathrm{I}(x; y) = \sum_{i=1}^{n} \sum_{j=1}^{n} \mathrm{P}(x_i, y_j) \cdot \log\left(\frac{\mathrm{P}(x_i, y_j)}{\mathrm{P}(x_i) \cdot \mathrm{p}(y_j)} \right),$$

де n – загальна кількість випадків;

x та y – випадкові величини;

x_i та y_i – i-те значення змінної x та y відповідно;

$P(x_i)$ та $P(y_j)$ – ймовірності появи x_i та y_j;

$P(x_i, y_j)$ – об’єднана ймовірність появи x_i та y_j.

Цей метод розглядає ознаки лише окремо, не враховуючи залежності між ними, і, таким чином, є одновимірним підходом.

Метод ReliefF

ReliefF – це популярний алгоритм для відбору ознак, який оцінює, наскільки кожна ознака сприяє розрізненню об’єктів різних класів. Його основна ідея полягає в наступному: для кожного об’єкта x_i у наборі даних алгоритм шукає k найближчих об’єктів того ж класу (так звані *збіги*), а також k найближчих об’єктів для кожного іншого класу (*промахи*) [10].

Далі ReliefF аналізує, наскільки значення кожної ознаки відрізняється між цільовим об’єктом та його сусідами. Якщо ознака має значну різницю між об’єктами різних класів (промахами), це свідчить про її високу інформативність – її вага збільшується. Натомість, якщо така ж відмінність спостерігається між об’єктами одного класу (*збігами*), це означає, що ознака нестабільна в межах класу – її вага зменшується.

Важливо, що ваги оновлюються з урахуванням ймовірності появи кожного класу у вибірці. Це дозволяє врахувати можливу диспропорцію між класами, завдяки чому всі промахи разом мають вагу, співрозмірну з вагами збігів.

У результаті ReliefF формує вектор ваг ознак, який показує їхню значущість для розділення класів. Ознаки з найвищими вагами можуть бути обрані як найбільш релевантні для побудови моделі.

Метод відбору ознак на основі генетичного алгоритму з обгорткою дерева рішень

Генетичний алгоритм – це еволюційний метод, який імітує природні процеси відбору та спадковості. У контексті відбору ознак кожен індивідуум (рішення) подається бінарним вектором, де 1 означає включення певної ознаки до моделі, а 0 – її відхилення. Початкова популяція складається з випадково згенерованих індивідуумів. Далі кожен з них оцінюється за функцією пристосованості, яка базується на точності класифікації (*balanced accuracy*) моделі дерева рішень, навченої на відповідній підмножині ознак із використанням п'ятикратної перехресної перевірки ($cv = 5$).

У кожному поколінні застосовуються оператори схрещування та мутації для створення нових рішень. Вибір батьків здійснюється за принципом турнірної селекції. Найкращі особини зберігаються в наступному поколінні завдяки механізму елітарності. Таким чином, з покоління в покоління оптимізується якість класифікації на основі отриманої комбінації ознак [16].

Метод RFE

Метод RFE – це обгортковий метод відбору ознак, заснований на їх ранжуванні. Він полягає в тому, що послідовно видаляються менш значущі ознаки, використовуючи алгоритм машинного навчання як ядро моделі. Метод був вперше запропонований Guyon та ін. [17] і реалізований у пакеті scikit-learn [18]. У дослідженнях використано метод RFE в обгортці дерева рішень (DT-RFE).

Блок-схема алгоритму DT-RFE наведена на рисунку 1. Спочатку, до навчального набору даних використовується дерево рішень для обчислення важливості ознак і їх ранжування, потім щоразу видаляються найменш значущі ознаки, а решта ознак використовуються для повторного тестування точності класифікації (Accuracy) за допомогою дерева рішень. Точність класифікації обчислюється ітеративно, поки не будуть переглянуті всі ознаки. Врешті-решт отримується ранжування всіх ознак за точністю класифікації.

Короткий опис критеріїв оцінювання ефективності методів відбору ознак

Для аналізу того, наскільки ефективним є той чи інший методу відбору ознак, у цій роботі використовуються такі критеріїв оцінки якості бінарної класифікації як: Recall, Precision, Accuracy та F-score.

Рис. 1. Блок-схема алгоритму DT-RFE

Accuracy (точність класифікації)

Точність класифікації є відсотком правильних прогнозів серед усіх прогнозів і розраховується за наступним виразом (1) [10]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{1}$$

де TP – кількість істинних позитивних результатів;
TN – кількість істинних негативних результатів;
FP – кількість хибнопозитивних результатів;
FN – кількість хибнонегативних результатів.

Ця метрика приймає значення в межах від нуля до одиниці, де одиниця означає, що всі прогнози правильні, а нуль – навпаки. Точність класифікації легко обчислити, хоча одним з її головних недоліків є те, що вона не є достатньо достовірною, якщо дані занадто незбалансовані (кількість зразків кожного класу суттєво відрізняється). Тому, при дослідженнях метрика Accuracy розраховувалась за наступним виразом [20]:

$$\text{Balanced}-\text{accuracy} = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right). \tag{2}$$

Формула (2) дозволяє обчислити збалансовану точність та уникнути завищених оцінок на незбалансованих наборах даних. Збалансоване значення точності класифікації – це макро-середнє (macro-average) значення повноти (recall) за класами. Для збалансованих наборів даних цей показник дорівнює звичайній точності класифікації, що розраховується за формулою (1) [20].

Precision (точність позитивних передбачень)

Точність позитивних передбачень (точність) – це показник, який кількісно визначає, скільки випадків, які прогнозуються як позитивні, насправді є позитивними. Він розраховується за формулою [10; 19]:

$$\text{Precision} = \frac{TP}{TP + FP}.$$

Recall (повнота)

Повнота, також відома як чутливість або частота хибнопозитивних результатів моделі. Це метрика, яка кількісно показує, скільки з фактично позитивних зразків були передбачені як позитивні і описується рівнянням [10; 19]:

$$\text{Recall} = \frac{TP}{TP + FN}.$$

F-score

F-score – це зважене гармонійне середнє значення точності та повноти, причому внесок точності в середнє значення зважується за допомогою певного параметра β. Вона визначається за формулою [10; 19]:

$$\text{F}_\beta - \text{score} = \left(1 + \beta^2\right) \cdot \frac{\text{precision} \cdot \text{recall}}{\left(\beta^2 \cdot \text{precision}\right) + \text{recall}}.$$

Щоб уникнути ділення на нуль, коли точність і повнота дорівнюють нулю, використовується еквівалентна формула [10; 19]:

$$\text{F}_\beta - \text{score} = \frac{\left(1 + \beta^2\right) \cdot TP}{\left(1 + \beta^2\right) \cdot TP + FP + \beta^2 \cdot FN}.$$

Коли $\beta = 2$ значення параметра Recall має у чотири рази більший вплив на F-score ніж Precision, що є особливо важливо для задач виявлення кібератак, де пропуск атаки є значно критичнішим за хибне спрацювання.

Як і точність класифікації так і точність позитивних передбачень, повнота та F-score набувають значень в межах від нуля до одиниці.

Аналіз результатів дослідження

Експериментальні дослідження було проведено з використанням інструментів Python та бібліотеки scikit-learn. Обчислення виконувались на системі з 96 ГБ оперативної пам'яті та двоядерним процесором із тактовою частотою 2,3 ГГц на ядро. Для наборів ознак, отриманих методами без вбудованого механізму ранжування, було проведено додаткову оцінку їх значущості за допомогою Permutation Importance.

В таблиці 3 наведено розраховані підчас досліджень значення часу навчання та тестування моделей МН на повному датасеті та на відібраних 15-ти ознаках, значення параметрів: Balanced Accuracy, Precision, Recall та F2-score.

На рисунках 2–8, *а–г* наведено отримані графіки залежності кількості пропущених кібератак та хибних спрацювань від кількості ознак, де:

а – результати обчислень для моделі МН на основі методу SVM;

б – моделі на основі методу RF;

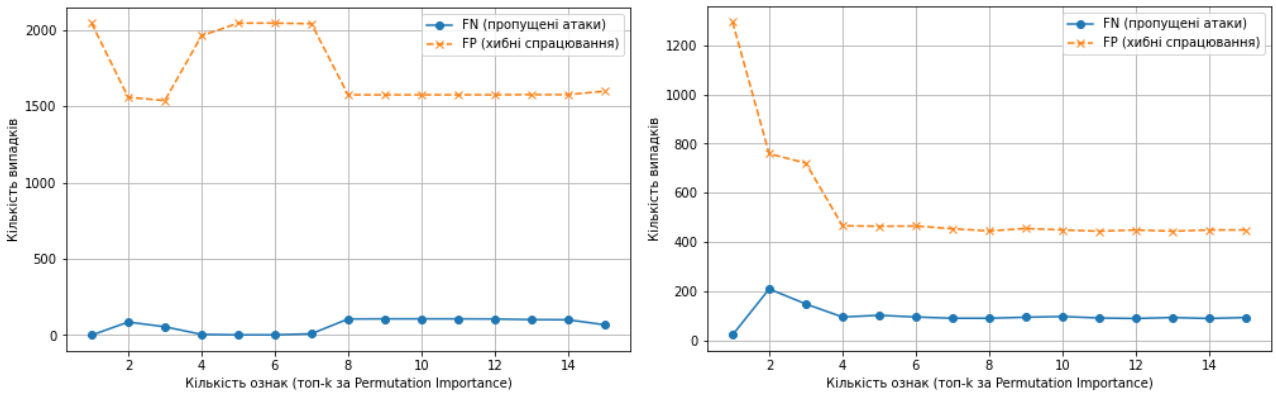
в – моделі на основі методу DT;

г – моделі на основі методу KNN.

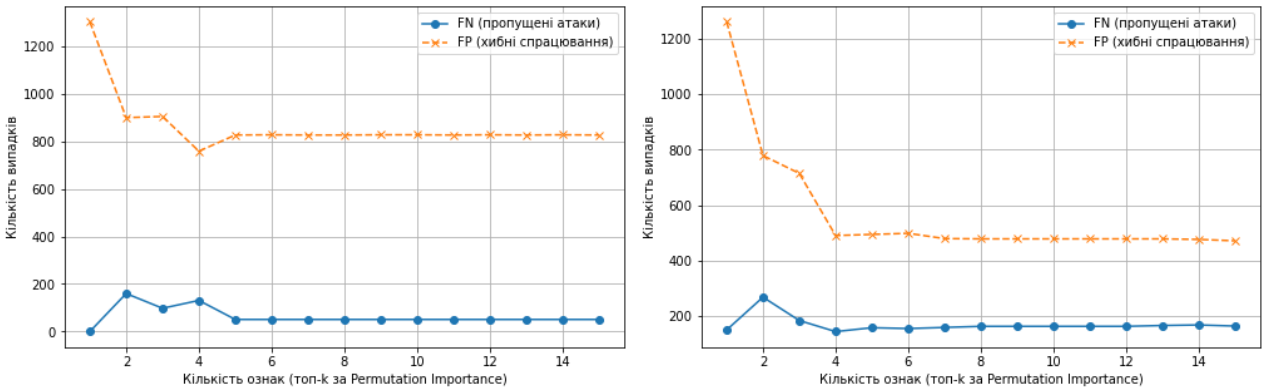
Таблиця 3

Значення показників якості роботи моделей машинного навчання при використанні різних методів відбору ознак

Метод и ML	Параметри оцінки	Усі ознаки	CFS	Алгоритм з χ^2	ANOVA	MIF	Relief	GA	RFE
DT	Час відбору ознак (с)	-	1354	0.06	0.1	32.2	24136	1176	17.9
	Час навчання (с)	0,484	0,078	0,079	0,091	0,185	0,141	0,062	0,152
	Час тестування (с)	0,015	0,01	0,005	0,004	0,005	0,016	0,016	0,004
	Balanced Accuracy	0,965	0,797	0,983	0,983	0,971	0,981	0,965	0,965
	Precision	0,993	0,967	0,981	0,981	0,997	0,984	0,993	0,994
	Recall	0,965	0,797	0,983	0,983	0,971	0,981	0,965	0,965
	F2 score	0,97	0,819	0,983	0,983	0,976	0,981	0,971	0,97
RF	Час навчання (с)	0,672	0,328	0,284	0,313	0,433	0,359	0,219	0,402
	Час тестування (с)	0,063	0,047	0,047	0,064	0,048	0,063	0,031	0,064
	Balanced Accuracy	0,98	0,89	0,991	0,991	0,993	0,986	0,992	0,99
	Precision	0,995	0,966	0,996	0,996	0,999	0,998	0,996	0,999
	Recall	0,98	0,89	0,991	0,991	0,992	0,986	0,992	0,99
	F2 score	0,983	0,903	0,992	0,992	0,994	0,988	0,993	0,992
SVM	Час навчання (с)	74,27	76,72	27,41	173,1	76,83	18,09	19,58	72,94
	Час тестування (с)	40,5	25,73	23,33	21,52	18,87	15,75	18,56	19,17
	Balanced Accuracy	0,635	0,608	0,754	0,753	0,873	0,887	0,893	0,885
	Precision	0,911	0,909	0,82	0,82	0,899	0,872	0,956	0,847
	Recall	0,635	0,608	0,754	0,753	0,873	0,887	0,893	0,885
	F2 score	0,652	0,622	0,764	0,764	0,878	0,884	0,904	0,876
KNN	Час тестування (с)	9,031	4,047	5,741	4,292	2,452	2,422	2,297	2,444
	Balanced Accuracy	0,981	0,882	0,985	0,984	0,989	0,983	0,994	0,99
	Precision	0,988	0,944	0,992	0,992	0,994	0,992	0,995	0,993
	Recall	0,981	0,882	0,985	0,984	0,989	0,983	0,994	0,99
	F2 score	0,982	0,893	0,986	0,986	0,99	0,985	0,994	0,991

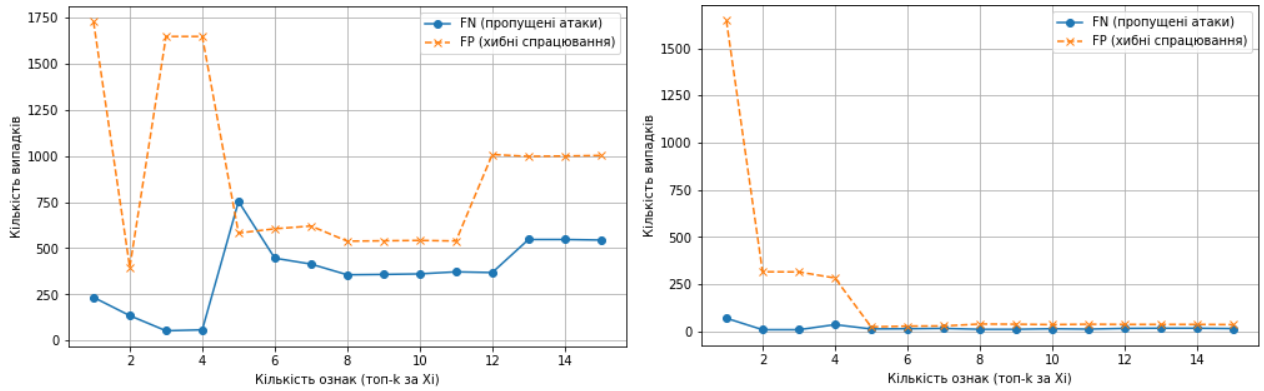


а) *б)*

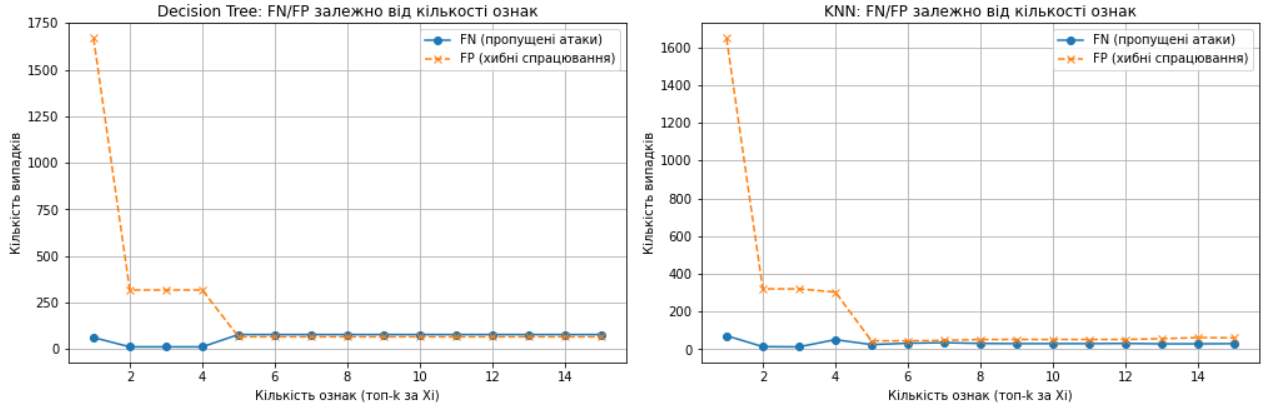


в) *г)*

Рис. 2. Графіки залежності кількості пропущених кібератак та кількості хибних спрацювань від кількості ознак при використанні методу CFS

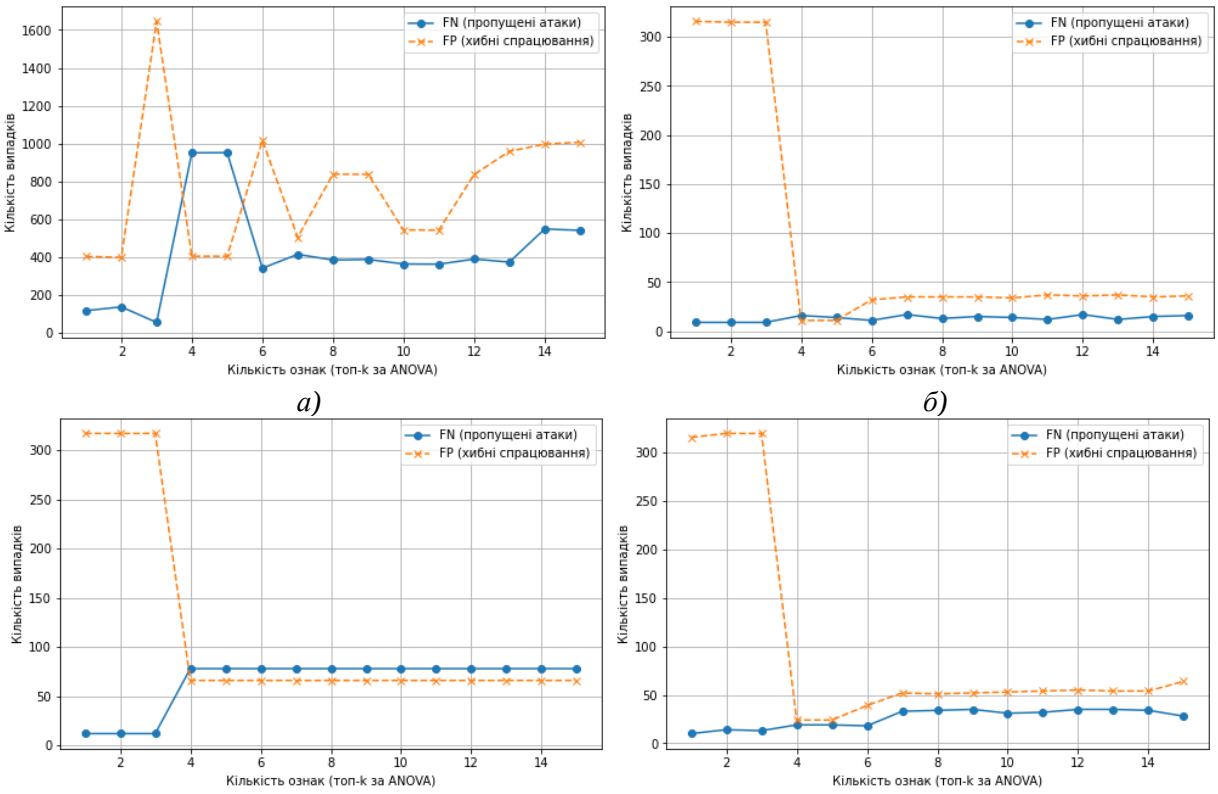


а) *б)*



в) *г)*

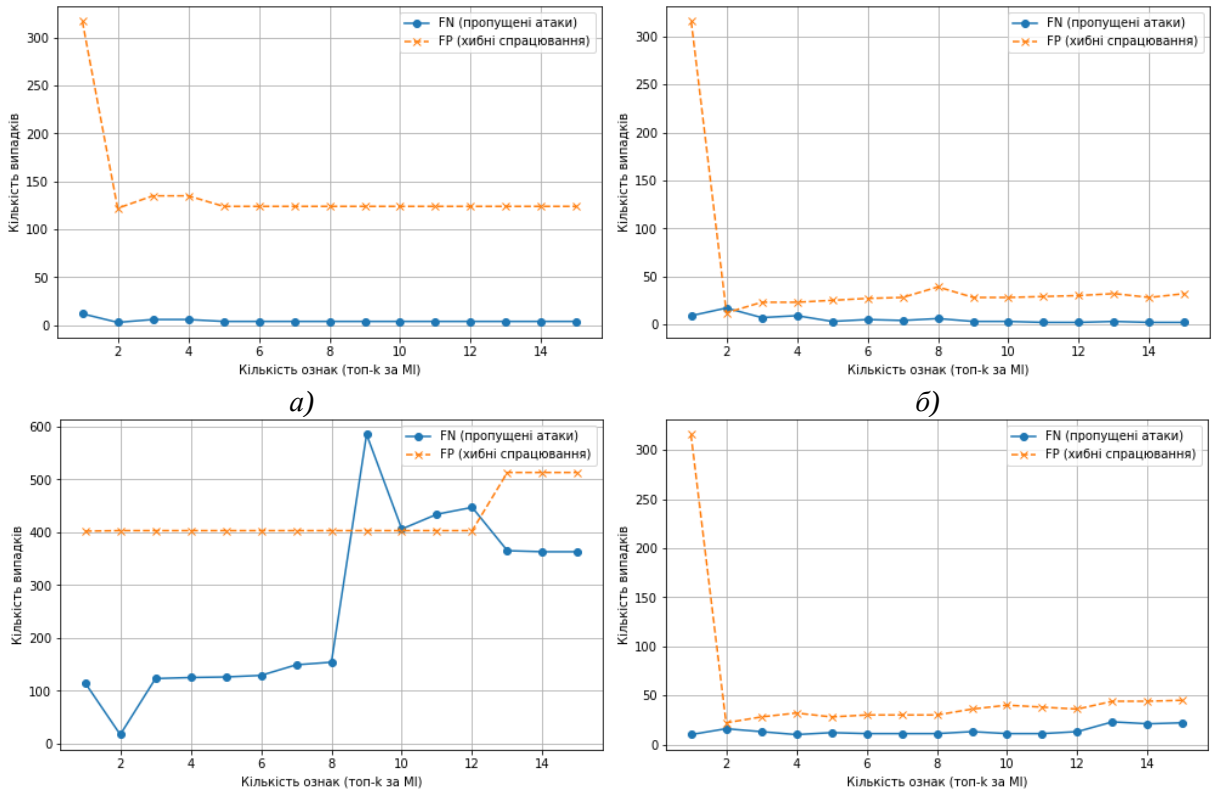
Рис. 3. Графіки залежності кількості пропущених кібератак та кількості хибних спрацювань від кількості ознак при використанні статистичного критерію χ^2



а) *б)*

в) *г)*

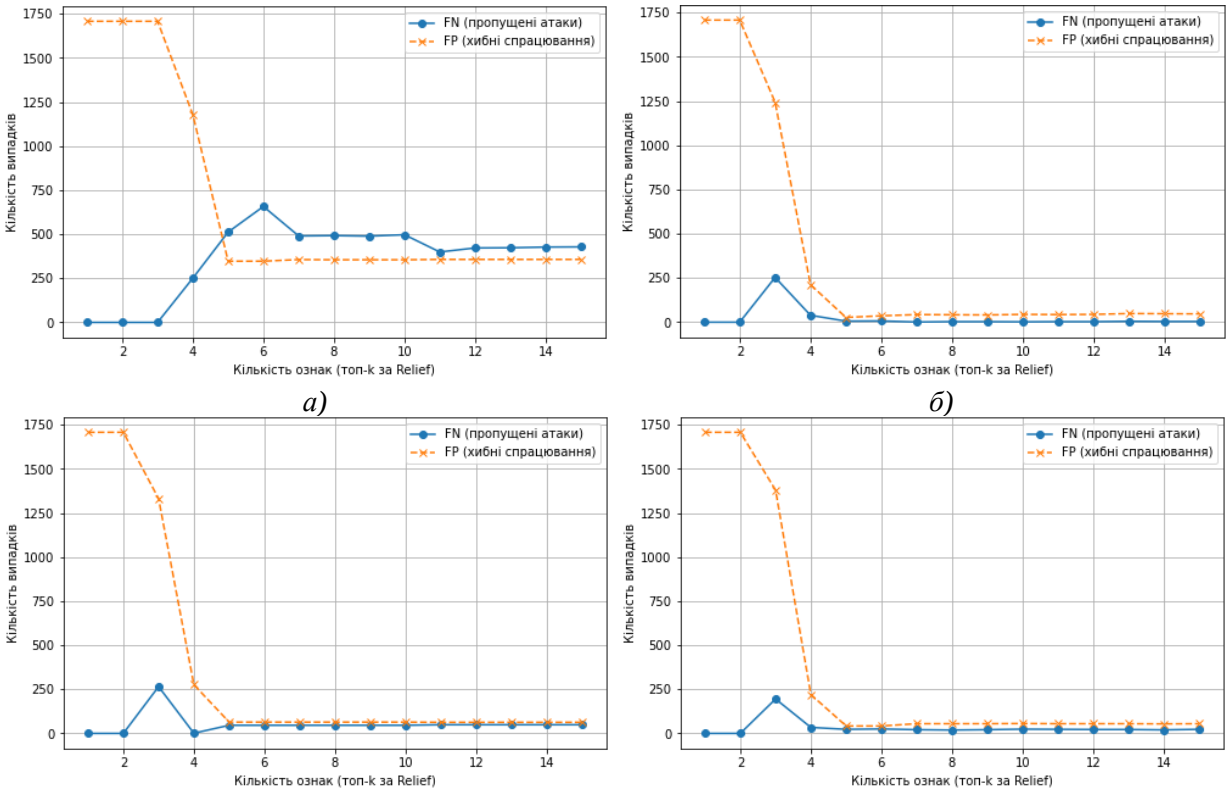
Рис. 4. Графіки залежності кількості пропущених кібератак та кількості хибних спрацювань від кількості ознак при використанні методу ANOVA:



а) *б)*

в) *г)*

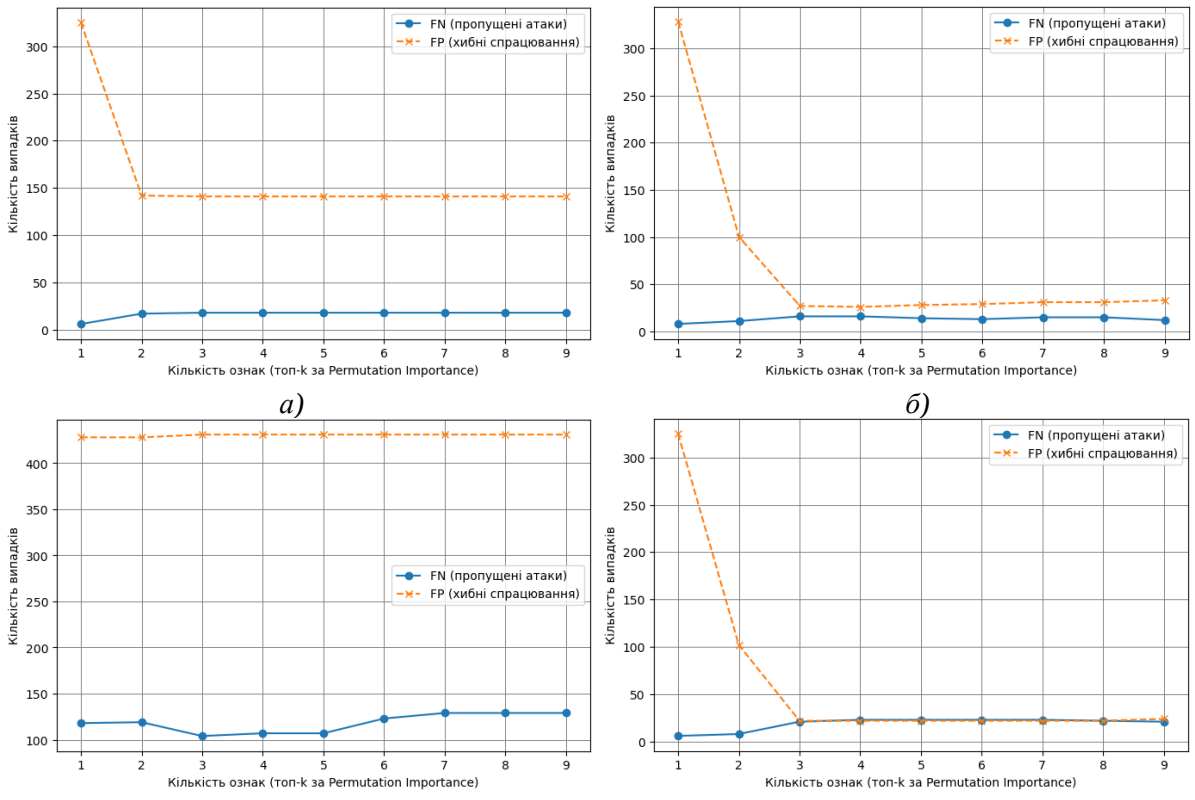
Рис.5. Графіки залежності кількості пропущених кібератак та кількості хибних спрацювань від кількості ознак при використанні методу MIF



а) *б)*

в) *г)*

Рис. 6. Графіки залежності кількості пропущених кібератак та кількості хибних спрацювань від кількості ознак при використанні методу Relief



а) *б)*

в) *г)*

Рис. 7. Графіки залежності кількості пропущених кібератак та кількості хибних спрацювань від кількості ознак при використанні GA

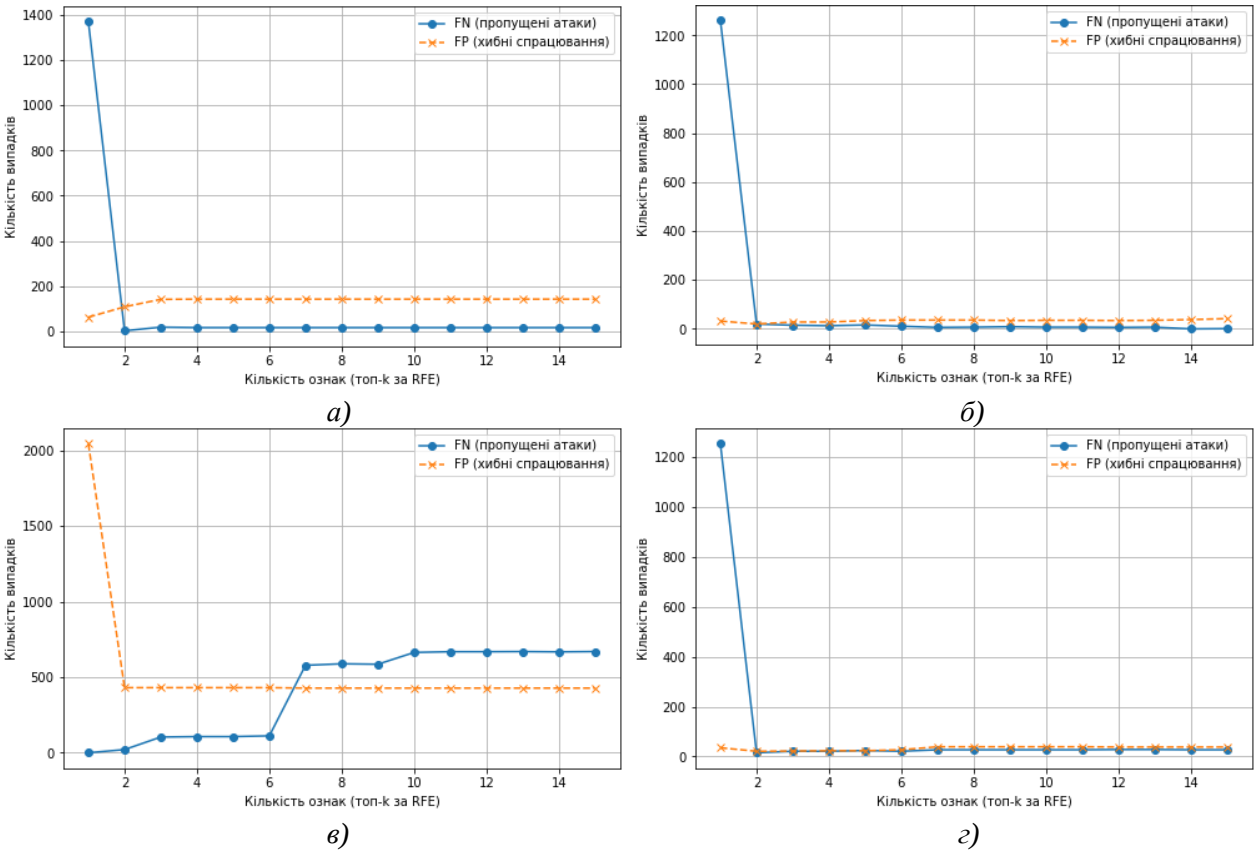


Рис. 8. Графіки залежності кількості пропущених кібератак та кількості хибних спрацювань від кількості ознак при використанні методу RFE

Аналіз результатів показав, що:

1. При використанні методу CFS усі досліджені моделі машинного навчання дають гірший результат за кількістю пропущених атак та кількістю хибних спрацювань ніж при навчанні на повному датасеті. Окрім того, обчислювальна складність цього методу тягне за собою значні затрати часу на відбір ознак, що нівелює виграш від зменшення часу на навчання моделі на меншому наборі ознак.

2. При використанні методу Relief було отримано покращення якості роботи усіх досліджуваних моделей машинного навчання, за усіма показниками, проте, цей метод погано працює на великих наборах даних оскільки має значну обчислювальну складність. Час на відбір 15 ознак із досліджуваного датасету склав більше ніж 6 годин з піковим використанням оперативної пам’яті 87 ГБ.

3. Не ефективною для детектування кібератак є модель на основі методу SVM, що пов’язано з його особливостями. Використання будь яких методів відбору ознак не призвело до значного покращення результатів роботи моделі. Час на навчання та тестування моделі є найвищим із розглянутих методів МН.

4. Найкращих результатів вдалось досягти при використанні алгоритму з χ^2, методів ANOVA та MIF в моделі, що побудована на основі RF. Дещо гірші показники в цьому поєднанні мають методи GA та RFE. Проте використання методів GA та RFE в моделі на основі методу KNN дає найкращий результат з усіх досліджуваних комбінацій методів відбору ознак та методів МН за кількістю детектованих атак та хибних спрацювань. Усі показники якості у такому випадку мають значення що перевищують 99 %.

Висновки

Загалом варто підкреслити, що використання методів відбору ознак дозволяє покращити показники якості роботи моделей машинного навчання, зменшити час на їх навчання. Проте слід зазначити, що ефективність кожного із методів залежить від об’єму та характеру даних, які потрібно опрацювати, методів машинного навчання з якими використовуються ті чи інші методи відбору ознак, часових та обчислювальних обмежень. Наведені в статті результати експериментальних досліджень можуть бути використані для вдосконалення існуючих, або розробки нових моделей машинного навчання, що використовуються в системах кіберзахисту.

Проведений аналіз не є вичерпним і не охоплює усі існуючі методи відбору ознак та їх комбінації, тому напрямками подальших досліджень є аналіз ефективності використання ансамблевих методів відбору ознак та оцінка стабільності різних методів (оцінює, чи залишається обрана підмножина ознак при незначних варіаціях вхідних даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. D. W. Fernando, N. Komninos, T. Chen. A Study on the Evolution of Ransomware Detection Using Machine Learning and Deep Learning Techniques. *IoT*. 2020. Vol. 1, no. 2. Pp. 551–604. URL: https://doi.org/10.3390/iot1020030.

2. M. S. Akhtar, T. Feng. Evaluation of Machine Learning Algorithms for Malware Detection. *Sensors*. 2023. Vol. 23, no. 2. P. 946. URL: https://doi.org/10.3390/s23020946.

3. Osanaiye O., Cai H., Choo K.-K. R., Dehghantanha A., Xu Z., Dlodlo M. Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. *EURASIP Journal on Wireless Communications and Networking.* 2016. Article number: 130. URL: https://doi.org/10.1186/s13638-016-0623-3.

4. Niveditha S., Prianka Rr., Sathya K., Shreyanth S., Nandhagopal Subramani, Balakrishnan Deivasigamani. Predicting Malware Classification and Family using Machine Learning: A Cuckoo Environment Approach with Automated Feature Selection. *Procedia Computer Science*. 2024. Vol. 235, P. 2434–2451. URL: https://doi.org/10.1016/j.procs.2024.04.230.

5. Xiujuan Wang, Yuchen Zhou. Multi-Label Feature Selection with Conditional Mutual Information. *Computational Intelligence and Neuroscience*. 2022. Article ID 9243893, P. 13. URL: https://doi.org/ 10.1155/2022/9243893.

6. Khammassi Chaouki, Krichen Saoussen. A GA-LR Wrapper Approach for Feature Selection in Network Intrusion Detection. *Computers & Security*. 2017. Vol. 70. P. 255–277. URL: DOI:10.1016/ j.cose.2017.06.005.

7. Mambwe Sydney Kasongo, Sun Yanxia. Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset. *Journal Of Big Data*. 2020. URL: https://doi.org/10.1186/s40537-020-00379-6.

8. Ullah I., Mahmoud Q. H. A Scheme for Generating a Dataset for Anomalous Activity Detection in IoT Networks. *Canadian Conference on Artificial Intelligence.* 2020. P. 508–520. URL: https://doi.org/ 10.1007/978-3-030-47358-7_52.

9. Furqan Rustam, Anca D Jurcut. Malicious traffic detection in multi-environment networks using novel S-DATE and PSO-D-SEM approaches. *Computers & Security.* 2023. № 136 (5): 103564. URL: https://doi.org/10.1016/j.cose.2023.103564.

10. M. C. Barbieri, B. I. Grisci, M. Dorn. Analysis and comparison of feature selection methods towards performance and stability. *Expert Systems With Applications.* 2024. Vol. 249, Part B. URL: https://doi.org/ 10.1016/j.eswa.2024.123667.

11. Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep learning. *The MIT Press*. Cambridge, Massachusetts London, England. 2016. 776 p.

12. Mark Hall. Correlation-based feature selection fordiscrete andnumeric class machine learning. *Proceedings of the 17th International Conference on Machine Learning (ICML2000)*. Stanford University, Stanford, CA, USA. 2000. P. 359–366.

13. Handoyo Samingun, Pradianti Nandia, Nugroho Waego, Akri Yusnita. A Heuristic Feature Selection in Logistic Regression Modeling with Newton Raphson and Gradient Descent Algorithm. *International*

Journal of Advanced Computer Science and Applications. 2022. Vol. 13, No. 3. P. 119–126. URL: DOI:10.14569/IJACSA.2022.0130317.

14. Kumar Mukesh, Rath Nitish, Swain Amitav, Rath Santanu. Feature Selection and Classification of Microarray Data using MapReduce based ANOVA and K-Nearest Neighbor. *Procedia Computer Science*. 2015. Vol. 54. P. 301–310. URL: https://doi.org/10.1016/j.procs.2015.06.035.

15. Scikit-learn documentation. SelectKBest. URL: :https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html (дата звернення: 28.10.2025).

16. Xue B., Zhang M., Browne W. N., Yao X. A survey on evolutionary computation approaches to feature selection. *IEEE Trans. Evolutionary Computation*. 2016. Vol. 20, no. 4, P. 606–626. URL: https://doi.org/10.1109/TEVC.2015.2504420.

17. Guyon I., Weston J., Barnhill S., Vapnik V. Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning*. 2002. Vol. 46. P. 389–422. URL: https://doi.org/10.1023/A:1012487302797.

18. Scikit-learn documentation. Recursive feature elimination. URL: https://scikit-learn.org/stable/modules/feature_selection.html#recursive-feature-elimination (дата звернення: 28.10.2025).

19. Scikit-learn documentation. Precision, recall and F-measure metrics. URL: https://scikit-learn.org/stable/modules/model_evaluation.html#precision-recall-f-measure-metrics (дата звернення: 28.10.2025).

20. Scikit-learn documentation. Balanced accuracy score. URL: https://scikit-learn.org/stable/modules/model_evaluation.html#balanced-accuracy-score (дата звернення: 07.11.2025).