КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 2 (30), 2025
Спеціальний випуск ISSN 2663 – 4023
Special issue
DOI 10.28925/2663-4023.2025.30.906
УДК 004.056.5:004.8

Денисюк Владислав Михайлович
аспірант
Національний технічний університет
«Харківський Політехнічний Інститут», Харків, Україна
ORCID: 0009-0001-6334-1473
vladyslav.denysiuk@cs.khpi.edu.ua

ПРОГНОЗУВАННЯ ЕКСПЛУАТАЦІЇ CVE НА ОСНОВІ ВІДКРИТИХ ДАНИХ NVD ТА KEV ДЛЯ РИЗИКО-ОРІЄНТОВАНОЇ ПРІОРИТИЗАЦІЇ

Анотація. Зі зростанням кількості публічно розкритих вразливостей програмного забезпечення команди безпеки стикаються з дедалі більшими труднощами у визначенні пріоритетів проблем, що потребують термінового усунення. Хоча такі системи, як Загальна система оцінювання вразливостей (CVSS), надають оцінки серйозності, вони не вказують, чи буде вразливість використано на практиці. Дослідження пропонує підхід на основі машинного навчання для прогнозування експлуатаційності вразливостей із використанням структурованих публічних даних з Національної бази даних вразливостей (NVD) та каталогу відомих експлуатованих вразливостей (KEV), що підтримується CISA. Сформовано маркований набір даних із понад 300 000 CVE, де випадки експлуатації ідентифікуються за KEV. Вилучені ознаки охоплюють вектори CVSS, ідентифікатори CWE, метадані постачальника/продукту та часові характеристики. Через екстремальний дисбаланс класів (експлуатовані CVE становлять ~0,45%) застосовується метод надсемплінгу і налаштування порогу прийняття рішень. Логістичну регресію, навчену в ML.NET, використано для побудови інтерпретованої моделі; показано, що вона вивчає змістовні патерни, які відрізняють експлуатовані вразливості. Оцінювання на спектрі порогів демонструє високу повноту та зростання точності, пропонуючи прозорий і відтворюваний інструмент для пріоритизації виправлень. Додатково окреслено обмеження, пов’язані з неповнотою KEV як «джерела істини», та окреслено напрями вдосконалення: інтеграцію NLP-ембеддингів описів CVE, калібрування ймовірностей і часово-орієнтовану валідацію для запобігання витоку даних. Такий підхід підсилює ризик-орієнтоване прийняття рішень у кібербезпеці та може бути безпосередньо інтегрований у процеси керування вразливостями в організаціях різного масштабу.

Ключові слова: CVE; CVSS; KEV; прогноз експлуатації; машинне навчання; логістична регресія; дисбаланс класів; пріоритизація виправлень; ML.NET; кібербезпека.

ВСТУП

Кількість публічно розкритих вразливостей програмного забезпечення зростає з безпрецедентною швидкістю. Національна база даних вразливостей (NVD) щорічно реєструє десятки тисяч нових поширених вразливостей та ризиків (CVE), створюючи величезний обсяг даних безпеки для аналізу організаціями. Хоча Загальна система оцінювання вразливостей (CVSS) забезпечує стандартизований спосіб оцінки серйозності, вона не вказує на те, чи дійсно вразливість експлуатується в реальних умовах. Як результат, багато вразливостей високого рівня (наприклад, CVSS ≥ 9.0) привертають негайну увагу, тоді як деякі вразливості з нижчим рейтингом, але активно експлуатуються, можуть бути знехтовані. Ця невідповідність ускладнює процес прийняття рішень для команд безпеки, які намагаються визначити пріоритети своїх зусиль щодо реагування.

На практиці проблема ускладнюється масштабом та різнорідністю продуктів програмного забезпечення: розподілені гібридні інфраструктури, контейнери й безсерверні середовища, залежності від відкритого коду та ланцюжків постачання збільшують поверхню атаки й прискорюють появу похідних уразливостей. Обмежені вікна для встановлення оновлень, міжкомандні залежності (SecOps, IT Ops, DevOps) і бізнес-обмеження щодо простоїв додатково ускладнюють своєчасне усунення ризиків. Лише наявності "статичного" балу CVSS часто недостатньо для реалістичного ранжування задач: експлуатаційність визначається не лише технічними параметрами, а й динамікою загроз, доступністю експлойтів, мотивацією зловмисників і масовістю встановленої бази.

Відповідь спільноти на цю прогалину охоплює як ретроспективні ініціативи (наприклад, каталоги на кшталт KEV, що фіксують факт активної експлуатації), так і прогностичні підходи, які намагаються оцінити ймовірність експлуатації у найближчій перспективі. Проте ретроспективні джерела за своєю природою не передбачають майбутні атаки, а прогностичні системи часто обмежуються вузькими наборами ознак або мають недостатню прозорість щодо методів і даних. У підсумку організаціям потрібні інструменти, які поєднують відкритість джерел, інтерпретованість моделей і здатність працювати з дисбалансом класів, властивим експлуатаційним подіям.

У цьому контексті доцільним є ризик-орієнтований підхід до керування вразливостями, який підсилює традиційні оцінки CVSS аналізом ознак, що корелюють із реальною експлуатацією. Центральним завданням стає побудова прозорої моделі, здатної на основі публічних структурованих даних виділяти вразливості з підвищеною ймовірністю експлуатації, зберігаючи при цьому відтворюваність і практичну корисність для операційних процесів (планування виправлень, тимчасові пом'якшення ризиків).

Запропонований підхід ґрунтується на формуванні міченого набору даних із джерел NVD та KEV, інженерії релевантних ознак (атрибути CVSS, CWE, вендор/продукт, часові характеристики), а також на застосуванні інтерпретованих методів машинного навчання з урахуванням екстремального дисбалансу класів. Особлива увага приділяється налаштуванню порогів прийняття рішень і оцінюванню на широкому діапазоні порогових значень, що дозволяє керовано балансувати повноту й точність у залежності від операційних вимог.

Метою статті є дослідження та аналіз здатності прогнозувати, які CVE, ймовірно, будуть використані в реальних сценаріях. Аналізуючи закономірності в історичних даних про вразливості та загрози – оцінити ймовірність використання точніше, ніж лише за допомогою статичних систем оцінювання та розробити демонстраційний варіант моделі машинного навчання здатної на це.

Огляд літератури. Сучасні роботи сходяться на тому, що класичні метрики на кшталт CVSS недостатні для оперативної пріоритезації: вони добре описують «потенційну» небезпеку, але не відображають реальну імовірність атак. Узагальнену рамку для конструювання метрик вразливостей (із розведенням понять «серйозність» і «ризик» та місцем моделей на кшталт EPSS у практичній пріоритезації) пропонує дослідження (Almahmoud et al., 2025), яке підкреслює потребу поєднувати технічні ознаки CVE з контекстом загроз і динамічними сигналами з екосистеми кіберзагроз.

Інша робота (Ferdous et al., 2025) аналізує якість і структуру базових джерел даних для побудови моделей. Велике емпіричне дослідження в Information and Software Technology детально розглядає «анатомію» сховищ вразливостей (NVD тощо), включно з еволюцією полів CVSS/CWE, практикою оновлень і типовими джерелами похибок. Ці результати прямо впливають на стратегії побудови датасетів, очистку, злиття записів і

керування «шумом» міток – критично важливі аспекти для коректного навчання й валідації моделей експлуатаційності.

З погляду методів, тренд останніх років – рух від «плоских» табличних ознак до семантично багатших подань і багатоджерельних підходів. Текстові моделі на базі представлень описів CVE (від TF-IDF до трансформерів) доповнюються знання-орієнтованими структурами. Зокрема, у новій ACM публікації запропоновано компактний граф знань для ризик-орієнтованого аналізу вразливостей: поєднання довідників (CVE/CWE/CPE), контексту домену та залежностей полегшує виведення непрямих сигналів ризику і підтримує сценарії пріоритезації з урахуванням взаємозв'язків компонентів. Хоча випадок доменно-специфічний, підхід є загальним і застосовуваним до програмного забезпечення.

Ще одна важлива тема – агрегування та нормалізація зовнішніх джерел експлойтів: репозиторії PoC/exp (Exploit-DB, Metasploit), телеметрія, сигнали соцмереж. Оглядова робота (J. Lyu et al., 2021) систематизує екосистему сховищ експлойтів і підсвічує практичні проблеми: фрагментація, різна якість метаданих, дублікати, часові зсуви між публікацією CVE, появою PoC і фіксацією «в польових умовах». Ці спостереження пояснюють, чому моделі, що враховують «живі» джерела, зазвичай перевершують чисто CVSS-базові бенчмарки.

ТЕОРЕТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ

Загальна система оцінювання вразливостей (CVSS) – це широко прийнята система для кількісної оцінки серйозності вразливостей програмного забезпечення. Версія CVSS 3.1, найновіша на момент написання, обчислює складений бал від 0,0 до 10,0 на основі набору базових, часових та екологічних показників. Ці показники включають:

1) Вектор атаки (AV): Описує, наскільки віддаленим має бути зловмисник (наприклад, локальний, суміжний, мережевий);
2) Складність атаки (AC): Вимірює умови, що знаходяться поза контролем зловмисника та повинні існувати;
3) Необхідні привілеї (PR): Визначає рівень привілеїв, необхідний для успішної атаки;
4) Взаємодія з користувачем (UI): Вказує, чи потрібна взаємодія людини;
5) Вплив на конфіденційність, цілісність та доступність (CIA): Вимірює потенційну шкоду, якщо вразливість буде використана.

Ці значення об'єднуються за допомогою визначеної математичної формули для отримання базового балу, який відображає внутрішні характеристики вразливості, що є постійними з часом та в різних середовищах. Хоча CVSS є цінним методом стандартизації способів повідомлення про вразливості та їх порівняння, він має суттєві обмеження при застосуванні для прогнозування експлойтів або визначення пріоритетів на основі ризиків:

1) Теоретичний та емпіричний ризик: CVSS відображає гіпотетичний сценарій, за якого вразливість може бути використана, без доказів того, чи була вона коли-небудь використана в реальних умовах. Як наслідок, оцінка може спотворювати фактичний ризик;
2) Надмірний акцент на технічних властивостях: CVSS не враховує критичні реальні фактори (поведінка та мотивація зловмисників, вік вразливості)
3) Хибна пріоритизація: Команди безпеки, що покладаються виключно на CVSS, можуть надавати пріоритет «критичним» вразливостям (оцінка ≥ 9,0), які ніколи

не експлуатуються, ігноруючи «середні» або «низькі» вразливості, які активно атакують зловмисники.

Емпіричні дослідження (наприклад, [Sabottke et al., 2015], [Bozorgi et al., 2010]) продемонстрували слабку кореляцію між оцінками CVSS та експлуатацією в реальних умовах, що підкреслює необхідність більш динамічних та контекстно-залежних моделей [1].

Протягом останнього десятиліття було проведено низку досліджень, присвячених застосуванню методів машинного навчання для прогнозування експлуатації вразливостей. Ці підходи можна умовно поділити на чотири основні категорії:

1) Моделі на основі тексту (NLP) засновані на обробці природної мови (NLP), використовують методи аналізу тексту для вилучення семантичних ознак із описів CVE та іншої неструктурованої інформації. Поширеними підходами є TF-IDF (термінова частота – обернена частота документів), векторні подання слів (word embeddings) та трансформерні моделі, такі як BERT. Мета цих моделей – виявити мовні шаблони або ключові слова, які можуть вказувати на ймовірність реальної експлуатації вразливості, наприклад, фрази "public exploit", "used in the wild" або "proof of concept". Наприклад, (Sabottke et al., 2015) продемонстрували, що аналіз згадок CVE у Twitter значно підвищує точність прогнозування експлуатації, підкреслюючи потенціал використання зовнішніх текстових сигналів у цій галузі.

2) Інженерія ознак на основі структурованих полів. Інший напрямок робіт зосереджений на використанні структурованих даних із баз вразливостей, таких як категоріальні поля (вектор атаки, складність доступу, класифікація CWE, постачальник та продукт). Ці атрибути часто поєднуються з числовими ознаками, такими як базова оцінка CVSS або кількість днів від моменту публікації. Для того, щоб зробити ці дані придатними для машинного навчання, їх зазвичай попередньо обробляють за допомогою one-hot кодування, нормалізації або використання embedding-шарів. Така структурована репрезентація дозволяє моделям виявляти закономірності, засновані на відомих технічних характеристиках вразливостей.

3) Техніки класифікації при дисбалансі класів. Однією з найбільших проблем прогнозування експлуатації вразливостей є крайній дисбаланс класів – експлуатовані вразливості складають менше 1% від загальної кількості CVE. Для подолання цієї проблеми використовуються різні стратегії, зокрема випадкове надсемплювання меншості (позитивного класу), підсемплювання більшості (негативного класу) та синтетичні методи, такі як SMOTE (Synthetic Minority Oversampling Technique) [2], який генерує нові синтетичні позитивні приклади. Деякі моделі також застосовують методи навчання з урахуванням вартості помилок (cost-sensitive learning), де помилка у передбаченні позитивного прикладу карається сильніше, ніж помилка при передбаченні негативного, завдяки чому модель стає більш чутливою до випадків експлуатації.

4) Гібридне та багатоджерельне навчання. Більш сучасні підходи використовують гібридні методи, інтегруючи структуровані дані з NVD з іншими джерелами, такими як стрічки розвідки загроз, бази даних експлойтів (наприклад, Exploit-DB, Metasploit) і навіть соціальні медіа платформи, такі як Twitter або Reddit. Ці джерела можуть забезпечити ранні індикатори активної експлуатації або обговорення серйозності вразливості. Деякі дослідження також включають графові ознаки, наприклад, залежності між програмними компонентами, або використовують часові моделі для врахування динаміки розкриття та експлуатації вразливостей. Такі багатоджерельні моделі дозволяють покращити актуальність та точність прогнозів,

використовуючи ширший контекст. Дане дослідження спирається на ці напрацювання, поєднуючи:

1) Структуровані поля CVSS з мітками експлуатації з KEV;
2) Застосування надсемплювання та оптимізації порогового значення для подолання дисбалансу;
3) Використання інтерпретованих моделей (логістична регресія) для забезпечення прозорості;
4) Побудову відтворюваного процесу на основі відкритих наборів даних та інструментів (ML.NET, NVD feeds).

У сукупності, ці підходи показали, що моделі, засновані на даних, можуть перевершувати CVSS у передбаченні експлуатації. Втім, багато моделей все ще страждають від проблем відтворюваності, обмеженої доступності даних або перенавчання через малі обсяги вибірок.

МЕТОДИКА ДОСЛІДЖЕННЯ

Набір даних, використаний у цьому дослідженні, було сформовано шляхом агрегування інформації з двох основних відкритих джерел: Національної бази даних вразливостей (NVD) та каталогу Відомих Експлуатованих Вразливостей (KEV), який підтримується Агентством з кібербезпеки та інфраструктурної безпеки США (CISA). NVD надає структуровані дані про всі опубліковані вразливості та експозиції (CVE), які розповсюджуються через регулярно оновлювані JSON-стрічки. Кожен запис у NVD містить детальну метаінформацію, таку як ідентифікатор CVE, текстовий опис, дати публікації та модифікації, а також набір показників безпеки. Особливий інтерес становлять вектори Common Vulnerability Scoring System (CVSS), зокрема базова оцінка серйозності, вектор атаки та необхідні привілеї – ці атрибути є ключовими індикаторами технічної серйозності вразливості та умов її експлуатації.

Доповнюючи дані NVD, каталог KEV надає кураторську інформацію про вразливості, які були підтверджені як активно експлуатовані у реальному світі. Цей набір даних слугує джерелом "істинних" міток для визначення, чи була вразливість фактично експлуатована на практиці. Записи KEV включають посилання на ідентифікатори CVE і часто містять додатковий контекст, такий як рекомендації постачальників, графіки виправлення вразливостей і дати експлуатації. Об’єднавши дані KEV і NVD, ми створили мічений набір даних, який дозволяє виконувати бінарну класифікацію для прогнозування експлуатації вразливостей.

З об’єднаного набору даних NVD-KEV було спроєктовано набір ознак, які відображають як технічні, так і часові характеристики кожної вразливості. Базову оцінку CVSS було витягнуто безпосередньо, яка відображає оцінену серйозність вразливості з урахуванням впливу на конфіденційність, цілісність та доступність, а також показників експлуатованості. Атрибут вектору атаки вказує, яким чином зловмисник може взаємодіяти з уразливою системою – розрізняючи мережеві, суміжні, локальні чи фізичні вектори атаки. Ще одним категоріальним показником є необхідність привілеїв, який визначає, чи потребує зловмисник підвищених привілеїв для успішного використання вразливості.

Крім того, було отримано ідентифікатор Common Weakness Enumeration (CWE) (якщо він наявний), який надає нормалізовану класифікацію типу програмної вразливості (наприклад, переповнення буфера, некоректна валідація введення). Імена постачальників та продуктів були отримані шляхом зіставлення записів NVD з тими, що

згадуються у записах KEV, створюючи контекстну відповідність між вразливостями та ураженим програмним забезпеченням. Нарешті, було розраховано часовий показник - "кількість днів від дати публікації", який визначає інтервал часу між датою публікації CVE та датою збору даних. Цей числовий показник дозволяє врахувати зрілість або вік вразливості, що може корелювати з ймовірністю її експлуатації.

Кожну вразливість CVE було промарковано за допомогою підходу бінарної класифікації, де цільова змінна **IsExploited** приймала значення **true**, якщо CVE була присутня у каталозі KEV, та **false** в іншому випадку. Такий бінарний підхід спрощує задачу до проблеми контрольованого навчання, де метою є навчити модель розрізняти експлуатовані та неексплуатовані вразливості на основі їх ознак.

Ця стратегія маркування передбачає, що всі CVE, яких немає у KEV, є неексплуатованими, що потенційно може призвести до появи хибнонегативних прикладів через неповноту доступної розвідувальної інформації про загрози. Проте, KEV залишається високоякісним джерелом перевірених даних про активну експлуатацію вразливостей. Завдяки своїй консистентності та офіційному статусу, KEV є практичним механізмом маркування для цілей досліджень.

Усі експерименти та процеси навчання моделей у цьому дослідженні були реалізовані з використанням ML.NET – фреймворку машинного навчання від Microsoft для екосистеми .NET. ML.NET надає гнучкий API для побудови повноцінних моделей машинного навчання безпосередньо в середовищі C#, що дозволяє легко інтегрувати моделі у наявні корпоративні системи. Його розширювана архітектура підтримує перетворення даних, такі як кодування категоріальних ознак, обробка текстових полів і оцінка моделей з використанням як класичних, так і сучасних алгоритмів.

У контексті нашого дослідження ML.NET виявився особливо придатним завдяки своїй здатності ефективно працювати зі зрідженими (sparse) даними, швидкими часами навчання, а також вбудованій підтримці логістичної регресії та моделей на основі дерев рішень. Були проаналізовані та використані два алгоритми навчання:

1) **Стохастична Двокоординатна Ацента (SDCA) для Логістичної Регресії** – лінійний алгоритм класифікації, який добре масштабується на великих та зріджених (sparse) наборах даних [3]. Його ефективність та інтерпретованість роблять його надійним кандидатом для задач бінарної класифікації, таких як прогнозування експлуатації вразливостей. Оптимізаційна задача SDCA формулюється наступним чином:

2)

$$\min \frac{1}{n}\sum_{i=1}^{n} \log\left(1+e^{-y_i w^t x_i}\right)+\lambda \mid w\mid_2^2 \qquad (1)$$

де:

1) x_i – вектор ознак для і-го зразка;

2) $y_i \in \{-1, 1\}$ – мітка класу (експлуатована або неексплуатована вразливість);

3) λ – параметр регуляризації, що контролює складність моделі та запобігає перенавчанню.

3) **FastTree Binary Classification (експериментально)** – це алгоритм градієнтного бустингу дерев рішень (GBDT), який будує послідовність дрібних дерев, де кожне наступне дерево навчається на помилках (залишках) попереднього. Алгоритм мінімізує диференційовану функцію втрат (зазвичай логістичну втрату для задач класифікації), поступово покращуючи передбачення моделі. Попри

високу точність моделей GBDT, вони є менш інтерпретованими у порівнянні з лінійними моделями, такими як логістична регресія [6]. Крім того, тренування GBDT моделей потребує більше обчислювальних ресурсів і часу, що зумовило зосередження основної уваги у цьому дослідженні саме на SDCA Logistic Regression як базовій моделі для пояснюваної та оперативної експлуатації.

Набір даних мав значний дисбаланс класів: експлуатовані вразливості складали менше 0,5% від загальної кількості записів. Така нерівномірність призводить до того, що більшість класифікаторів схильні робити прогнози на користь домінуючого класу (неексплуатовані вразливості), ігноруючи рідкісні позитивні зразки, що суттєво знижує показник Recall (чутливість) для класу експлуатованих вразливостей.

Щоб пом'якшити цей ефект, було застосовано метод випадкового надсемплінгу (Random Oversampling). Зокрема, усі наявні позитивні приклади (експлуатовані CVE) були дубльовані до досягнення фіксованої цільової кількості у 100,000 позитивних зразків для тренувальної вибірки. При цьому всі негативні приклади (неексплуатовані CVE) були збережені у повному обсязі без видалення чи зміни їхньої кількості.

В результаті розмір тренувальної вибірки становив:

$$TrainingSetSize = P_{target} + N_{train} = 341000 \quad (2)$$

Цей підхід до надсемплінгу був орієнтований на максимізацію Recall, оскільки він дозволяє моделі більше взаємодіяти з експлуатованими прикладами під час навчання. Це суттєво знижує ризик того, що модель буде ігнорувати важливі патерни, притаманні рідкісному позитивному класу. Особливу увагу було приділено уникненню витоку даних (data leakage). Процедура надсемплінгу застосовувалася виключно до тренувальної вибірки. Валідаційна (тестова) вибірка залишалась незміненою, відображаючи реальну пропорцію класів. Це забезпечує об'єктивне оцінювання здатності моделі узагальнювати нові дані під час тестування.

Таким чином, надсемплінг допоміг компенсувати дисбаланс класів без потреби у складніших техніках синтетичної генерації зразків, таких як SMOTE, та забезпечив кращу чутливість моделі до рідкісних випадків експлуатації. Набір даних містив як категоріальні, так і числові змінні. До категоріальних змінних належали:

1) Attack Vector
2) Privileges Required
3) CWE
4) VendorProject
5) Product

Ці змінні оброблялися за допомогою one-hot кодування, перетворюючи кожне категоріальне значення у бінарний вектор, наприклад:

$$AV = NETWORK \rightarrow [1,0,0,0] \ (\text{one} - \text{hot for Network, Adjacent, Local, Physical}) \quad (3)$$

Числові поля, такі як CVSS Score та Кількість Днів з Моменту Публікації (Days Since Published), використовувалися без нормалізації, оскільки деревоподібні моделі та лінійні класифікатори із вилученими ознаками стійкі до відмінностей у масштабах значень. Щоб забезпечити справедливу оцінку продуктивності моделі та уникнути витоку даних, було застосовано фіксоване стратифіковане розбиття вибірки:

1) Тренувальна вибірка: 1112 експлуатованих (позитивних), 241400 неексплуатованих (негативних);
2) Тестова вибірка: 267 експлуатованих, 60360 неексплуатованих.

Цей підхід гарантував, що тестова вибірка залишалася репрезентативною до реального розподілу CVE, а ефекти від надсемплінгу впливали лише на тренувальну фазу. Фіксоване розбиття забезпечує узгоджену оцінку результатів моделі та ізолює вплив надсемплінгу виключно до етапу навчання.

За замовчуванням логістична регресія видає оцінку ймовірності $\hat{\rho} \in [0,1]$, і використовує поріг 0.5 для перетворення цього значення у бінарне передбачення. Однак у випадках сильного дисбалансу класів типовий поріг може не забезпечувати оптимального співвідношення між точністю (precision) та повнотою (recall). Тому ми систематично оцінювали продуктивність моделі в діапазоні порогових значень $\tau \in [-5.0, +8.0]$ для логіт-виходів $\hat{z}$, де:

$$\hat{\mathrm{p}} = \frac{1}{1+e^{-2}} \tag{3}$$

Для кожного кандидатного порогу обчислювались наступні метрики:

1) Precision: $\frac{TP}{TP+FP}$
2) Recall: $\frac{TP}{TP+FN}$
3) F1 Score: $2\frac{\mathrm{Pr}ecision*\mathrm{Re}call}{\mathrm{Pr}ecision+\mathrm{Re}call}$

Цей процес дозволив нам налаштувати класифікатор у напрямку вищої повноти або збалансованого F1-показника, в залежності від потреб реального використання та маючі сильно незбалансовані дані [4][5]. Для оцінки ефективності кожної моделі було обчислено комплексний набір метрик:

1) Accuracy – корисна, але оманлива при сильному дисбалансі класів;
2) Precision – висока точність означає меншу кількість хибнопозитивних спрацювань;
3) **Recall** – висока повнота гарантує виявлення більшості експлуатованих вразливостей;
4) F1 Score – гармонійне середнє між точністю та повнотою, що відображає баланс між цими показниками;
5) AUC (Area Under ROC Curve) – вимірює ймовірність того, що випадково вибраний позитивний приклад отримає вищий рейтинг, ніж випадково вибраний негативний приклад.

Кожна з метрик відповідає різним операційним потребам. Наприклад, фахівці з реагування на інциденти можуть віддавати перевагу високій повноті (щоб не пропустити загрозу), тоді як автоматизовані системи вимагатимуть високої точності (щоб зменшити кількість хибних спрацювань).

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Після навчання класифікатора логістичної регресії з використанням алгоритму стохастичної подвійної координатної асценти (Stochastic Dual Coordinate Ascent, SDCA), ми оцінили здатність моделі до розмежування класів шляхом аналізу вихідних оцінок (raw scores) на відкладеному тестовому наборі. Ці вихідні оцінки являють собою логіти – необмежені значення, отримані з лінійної функції прийняття рішення до застосування сигмоїдальної трансформації. У логістичній регресії логіт для певного зразка x обчислюється за формулою 4:

$$\log it(x) = w^t x + b \tag{4}$$

Де, w – це вектор ваг, вивчених під час тренування; b – це зміщення (bias term); x – це вектор ознак, що представляє конкретний випадок CVE. Цей логіт пізніше перетворюється у ймовірнісну оцінку $p = \sigma(logit)$ за допомогою сигмоїдальної функції:

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{5}$$

Проте для задач оптимізації порогу та аналізу відокремлення класів ми працюємо безпосередньо з "сирими" логітами, оскільки вони мають ширший числовий діапазон та забезпечують вищу роздільну здатність при побудові рішень.

Позитивні зразки, (IsExploited = true)

Для 267 тестових зразків, позначених як експлуатовані, "сирі" логіт-оцінки показали компактний та високий розподіл значень. Спостережувані значення логітів знаходилися в діапазоні [5,93, 7.51] , а середнє значення логіт-оцінки становило приблизно $\mu_{pos} \approx 6.62$. Таким чином, майже всі експлуатовані вразливості були передбачені з дуже високим рівнем впевненості як позитивні випадки.

Негативні зразки, (IsExploited = false)

На противагу цьому, для 60,360 тестових зразків, позначених як неексплуатовані, логіт-оцінки варіювались у широкому діапазоні від [-4.68, 3.14] із середнім значенням of $\mu_{neg} \approx -0.23$. Такий центрований у негативній області розподіл є очікуваним результатом для логістичної регресії, яка прагне лінійно розділити класи, де негативні значення свідчать про відсутність належності до позитивного класу. Хоча верхній хвіст розподілу для неексплуатованих зразків досягав позитивних логітів (наприклад, 3.14), основна маса значень знаходилася значно нижче мінімального значення логіту експлуатованих зразків, яке становить 5.93.

Розділення оцінок між експлуатованими та неексплуатованими зразками чітко простежується у статистичному підсумку:

1) Відсутнє перекриття між мінімальним значенням логіту для позитивного класу (5.93) та максимальним значенням для негативного класу (3.14).

2) Середнє значення для позитивних логітів (6.62) знаходиться майже на сім стандартних відхилень вище середнього значення для негативного класу, якщо припустити приблизну нормальність розподілу.

Це суттєве розділення свідчить про те, що модель навчилася визначати добре калібровану та осмислену межу прийняття рішень, яка дозволяє ефективно здійснювати пост-фактум налаштування порогу. Більше того, це демонструє, що логістична регресія, навіть з лінійною межею розділення, є надзвичайно ефективною для даного завдання за умови використання відібраних ознак і правильної стратегії збалансування класів через оверсемплінг. Таке розділення є критичним для подальшого налаштування порогового значення, де ми прагнемо обрати τ таким чином, щоб:

$$P(\text{logit}(x) \geq \tau \mid \text{IsExploited} = 1) >> P(\text{logit}(x) \geq \tau \mid \text{IsExploited} = 0) \quad (6)$$

забезпечуючи як високу повноту (recall), так і високу точність (precision), залежно від операційних потреб.

Щоб перетворити безперервні вихідні оцінки логістичної регресії у практично застосовні бінарні передбачення, ми провели систематичну процедуру налаштування порогу. Замість використання фіксованого значення (зазвичай 0.0 для логіт-оцінок або 0.5 для ймовірностей після сигмоїди), класифікатор було протестовано в діапазоні порогів від -5.0 до +8.0 з кроком 0.01. Такий вичерпний пошук дозволив детально проаналізувати, як змінюються показники ефективності моделі – зокрема точність (precision), повнота (recall) та F1-метрика – в залежності від обраної межі прийняття рішення.

Аналіз показав стабільну закономірність: повнота залишалася на рівні 1.000 у широкому діапазоні низьких та середніх значень порогу, що свідчить про те, що всі дійсно експлуатовані випадки (тобто CVE, які входять до каталогу KEV) були коректно класифіковані як експлуатовані незалежно від обраного порогу. Це свідчить про те, що модель із високою впевненістю присвоює вищі оцінки цим позитивним зразкам. Зі збільшенням порогу точність поступово зростала, що вказує на зменшення кількості хибнопозитивних передбачень у класі експлуатованих. Фрагмент результатів налаштування порогу наведено нижче у таблиці 1:

Таблиця 1

Результати налаштування порогу

Threshold	Precision	Recall	F1 Score
0.10	0.044	1.000	0.085
0.40	0.049	1.000	0.093
0.55	0.096	1.000	0.175
0.75	0.139	1.000	0.244
0.85	**0.189**	**1.000**	**0.317**

Оптимальне значення порогу було обране на основі максимального F1-значення, яке досягалося при порозі 0.85. Повнота залишалася стабільною на рівні 1.000 до цього порогу, що свідчить про повне охоплення всіх дійсно експлуатованих вразливостей. Точність покращувалася з підвищенням порогу, що дозволяло поступово знижувати кількість хибнопозитивних передбачень.

При обраному порозі 0.85 модель досягла повноти (recall) 1.000, точності (precision) 0.189 та F1-метрики 0.317. Площа під ROC-кривою (AUC) становила приблизно 0.85, що свідчить про сильну здатність моделі відокремлювати два класи. Загальна точність

класифікації становила близько 98%, однак цей показник слід інтерпретувати обережно, оскільки він значною мірою завищений через домінування негативних (неексплуатованих) прикладів у датасеті.

Матриця неточностей (confusion matrix), побудована для цієї конфігурації порогу, підтвердила, що всі реальні експлуатовані вразливості у тестовій вибірці були успішно виявлені як істинно позитивні (True Positives). Однак класифікатор також помилково відніс понад тисячу неексплуатованих CVE до класу позитивних, породжуючи хибнопозитивні передбачення (False Positives). Такі результати є очікуваними з огляду на крайню рідкість позитивного класу у реальних наборах даних про вразливості.

З практичної точки зору, така модель є добре придатною для сценаріїв пріоритизації, де забезпечення повного охоплення усіх справді експлуатованих CVE є набагато важливішим завданням, ніж мінімізація кожного хибного сповіщення.

При оптимальному порозі 0.85 модель продемонструвала такі результати:

1) Precision: 0.189;
2) Recall: 1.000;
3) F1 Score: 0.317;
4) AUC (Area Under ROC Curve): ≈ 0.85;
5) Accuracy: ~98% (heavily influenced by class imbalance).

Матриця неточностей (threshold = 0.85) наведена на таблиці 2:

Таблиця 2

Загальна таблиця результатів роботи після прогнозування

	Predicted Exploited	**Predicted Not Exploited**
Actual Exploited	267 (TP)	0 (FN)
Actual Not Exploited	~1,150 (FP)	~59,210 (TN)

Істинно позитивні (TP) – усі дійсно експлуатовані вразливості були виявлені, відмінний результат для сценаріїв, де пріоритетом є висока повнота.

Хибнопозитивні (FP) – були помилково класифіковані кілька тисяч неексплуатованих вразливостей, що очікувано за умов сильного класового дисбалансу.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Експериментальні результати демонструють, що модель машинного навчання здатна ефективно навчитися розрізняти експлуатовані та неексплуатовані вразливості, використовуючи лише відкриті структуровані дані. Незважаючи на те, що загальний рівень експлуатації у датасеті надзвичайно низький (близько 0,45%), застосування методів надвибірки (oversampling) та налаштування порогу класифікації дозволило досягти високої повноти (recall) і поступового покращення точності (precision).

Ключовим висновком з цих експериментів є вирішальна роль налаштування порогу прийняття рішень (threshold tuning). Оцінюючи вихідні значення моделі в межах континууму можливих порогів, ми змогли досягти контрольованого балансу між precision та recall. Цей процес показав, що хоча повнота залишалася стабільно високою (ідеальною при низьких порогах), точність могла бути поступово покращена шляхом вибору оптимального значення порогу, яке мінімізує кількість хибнопозитивних спрацьовувань, не втрачаючи при цьому здатності моделі виявляти всі дійсно експлуатовані вразливості.

Ефективність моделі, навіть при обмеженому наборі ознак (наприклад, атрибути CVSS, такі як вектор атаки та необхідні привілеї), свідчить про те, що певні структуровані характеристики вразливостей мають суттєву прогностичну цінність, коли їх аналізується у сукупності. Це вказує на те, що моделі машинного навчання здатні виявляти приховані закономірності між технічними характеристиками вразливостей та реальними патернами експлуатації, навіть за відсутності багатшого контекстуального контенту (наприклад, текстових описів чи розвідувальних даних).

Основною сильною стороною цього дослідження є його повна орієнтація на відкриті, загальнодоступні джерела даних – зокрема, JSON-стрічки NVD (National Vulnerability Database) та каталог Known Exploited Vulnerabilities (KEV), який підтримується агентством CISA. Відмова від використання закритих даних та комерційних потоків загроз дозволяє забезпечити відтворюваність методології та гарантує, що організації будь-якого масштабу зможуть впровадити та адаптувати запропонований підхід без обмежень ліцензування чи доступу. Це сприяє демократизації прогнозування експлуатованості вразливостей та відкриває можливості для його ширшого застосування в кібербезпековій спільноті.

Ще однією важливою перевагою є прозорість та модульність побудованого пайплайну. Кожен етап робочого процесу – від формування датасету та створення ознак до вибору моделі, навчання та оцінки результатів – розроблений таким чином, щоб бути інтерпретованим і підзвітним. У контексті кібербезпеки така прозорість є критичною, адже здатність відслідковувати логіку прийняття рішень є обов’язковою умовою для формування довіри серед фахівців-практиків.

Крім того, використання інтерпретованих моделей, таких як логістична регресія, на противагу "чорним ящикам" ансамблевих методів, дозволяє командам безпеки розуміти та пояснювати отримані прогнози, що є вкрай важливим у випадках, коли ці прогнози впливають на процеси пріоритизації усунення вразливостей.

Попри обнадійливі результати, дослідження стикається з кількома суттєвими обмеженнями, які потребують вирішення у майбутніх роботах. Першим і найважливішим обмеженням є неповнота каталогу KEV як джерела істинних міток (ground truth). Хоча KEV є авторитетним джерелом, це все ж таки ретроспективний набір даних, який охоплює лише частину активно експлуатованих вразливостей [7]. Багато інцидентів експлуатації, особливо пов’язаних із цільовими атаками або zero-day вразливостями, можуть залишатися незафіксованими через недостатнє висвітлення або прогалини в розвідданих. Така неповнота міток призводить до появи шуму у процесі оцінки моделі, зокрема штучно завищує кількість помилкових позитивних спрацювань, помилково позначаючи реально експлуатовані вразливості як негативні.

По-друге, поточний набір ознак ігнорує багату текстову інформацію, наявну в описах CVE. Опис часто містить критично важливі індикатори ймовірності експлуатації, такі як посилання на proof-of-concept експлойти, публічні розкриття або контекстуальні деталі щодо сценаріїв атак. Обмежуючи модель лише структурованими полями, підхід втрачає потенційно цінну семантичну інформацію, яка могла б значно покращити точність прогнозування.

Ще одним обмеженням є простота поточного процесу побудови ознак. Хоча такі поля, як вектор атаки, необхідні привілеї та CVSS-оцінка, надають базове розуміння характеристик вразливості, вони не відображають багатогранну та динамічну природу поведінки експлуатації. Фактори, як-от доступність експлойтів на підпільних форумах, потенціал для ланцюжків вразливостей (vulnerability chaining) і час реагування вендорів

на виправлення, не враховуються у поточному наборі ознак, що обмежує глибину розуміння моделі.

Для усунення зазначених обмежень і підвищення прогнозувальних можливостей моделі пропонуються кілька технічних удосконалень. Найбільш вагомим покращенням стане інтеграція методів обробки природної мови (NLP). Використовуючи моделі, такі як BERT, RoBERTa або FastText для генерації щільних семантичних ембеддінгів із описів CVE, система зможе захоплювати тонкі лінгвістичні патерни, що вказують на ймовірність експлуатації. Ембеддінги, отримані від таких моделей, дозволять класифікатору враховувати контекстне значення фраз на кшталт “public exploit available” або “exploited in the wild”, які наразі ігноруються у виключно структурованих пайплайнах.

Ще одним напрямком покращення є калібрування моделі, яке допоможе підвищити інтерпретованість вихідних ймовірностей. Такі методи, як Platt scaling або ізотонічна регресія (isotonic regression)[8], дозволяють узгодити передбачені моделлю ймовірності з фактичними, забезпечуючи командам безпеки більш надійні оцінки ризику. Добре відкалібровані моделі спрощують вибір порогових значень і дозволяють приймати більш обґрунтовані рішення в процесах пріоритизації вразливостей.

Крім того, впровадження часово-орієнтованих (time-aware) стратегій моделювання є критично важливим для усунення витоку даних (data leakage) і симуляції реальних операційних обмежень. Забезпечивши, що під час тренування і тестування моделі використовується лише та інформація, яка була доступна до дати публікації CVE, можна побудувати валідаторну систему, яка точніше відображатиме умови, з якими стикаються спеціалісти з безпеки під час оперативного реагування. Це покращить реалістичність і практичну значущість моделі, сприяючи довірі до її впровадження у робочих середовищах.

Нарешті, розширення набору даних за рахунок зовнішніх джерел розвідувальної інформації про загрози (threat intelligence feeds), репозиторіїв з proof-of-concept експлойтами, а також телеметричних даних від вендорів засобів безпеки дозволить сформувати більш комплексне уявлення про ландшафт експлуатацій. Інтеграція таких джерел у пайплайн моделі – через додаткові ознаки або мультиджерельне навчання - підвищить стійкість моделі та покращить її здатність виявляти нові тренди експлуатації вразливостей.

Це дослідження демонструє доцільність і ефективність застосування методів машинного навчання для прогнозування експлуатованості програмних вразливостей, використовуючи виключно відкриті структуровані дані. Інтегрувавши метадані з Національної бази даних вразливостей (NVD) та підтверджені факти експлуатації з каталогу відомих експлуатованих вразливостей (KEV), який підтримується CISA, було побудовано прозорий і відтворюваний процес, що перетворює сирі записи CVE на практичні оцінки ризику.

Попри суттєві виклики, пов’язані з екстремальним дисбалансом класів – коли менш ніж 0,5% CVE позначені як експлуатовані – застосування стратегій передискретизації (oversampling) та налаштування порогів дозволило досягти високої повноти (recall) із поступовим покращенням точності (precision). Використання логістичної регресії, моделі, яка цінується за інтерпретованість і ефективність, забезпечило зрозуміле і пояснюване розділення класів, що чітко відобразилося у розподілах сирих логіт-оцінок. Ця інтерпретованість є критично важливою для операцій з кібербезпеки, де перевага надається пояснюваним AI-рішенням над "чорними ящиками".

Результати дослідження підтверджують, що навіть мінімалістичний набір ознак, що складається з CVSS-атрибутів, таких як вектор атаки, необхідні привілеї та вік вразливості, містить достатньо прихованої інформації для виявлення патернів експлуатації при навчанні на агрегованих даних. Водночас, виявлені обмеження – зокрема, неповнота KEV як джерела істинних міток та відсутність семантичних ознак із текстових описів CVE – вказують на напрями для подальшого вдосконалення.

Найперспективнішим шляхом розвитку є інтеграція методів обробки природної мови (NLP). Використовуючи ембеддінги текстових описів вразливостей (наприклад, BERT або FastText), модель зможе захоплювати тонкі лінгвістичні індикатори експлуатацій, які не представлені у структурованих полях. Додатково, впровадження методів калібрування моделі допоможе узгодити ймовірнісні вихідні дані з реальними частотами експлуатації, що підвищить надійність прийняття рішень при виборі порогових значень.

Ще одним критичним напрямком розвитку є впровадження часово-орієнтованих (time-aware) методик валідації, які гарантуватимуть, що модель оцінюється в умовах, максимально наближених до реальних, з дотриманням суворих обмежень щодо доступності інформації на момент публікації CVE. Така часово-коректна перевірка дозволить уникнути ретроспективних викривлень і точно імітувати сценарії ухвалення рішень у реальному часі.

У підсумку, хоча це дослідження зосереджувалося на базових аспектах прогнозування експлуатацій на основі структурованих даних, запропонована методологія закладає основу для масштабованої та розширюваної системи. Шляхом поступового включення додаткових джерел даних – таких як репозиторії експлойтів, розвідувальні стрічки загроз та телеметрія безпеки – можна суттєво підвищити як прогнозну точність, так і практичну цінність моделі. Такий підхід здатен подолати розрив між теоретичними системами оцінювання вразливостей і реальними стратегіями пріоритизації ризиків, надаючи командам кібербезпеки інструменти для ефективнішого розподілу ресурсів і точнішого реагування на загрози.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Allodi, L., & Massacci, F. (2012). A preliminary analysis of vulnerability scores for attacks in wild: The EKITS and SYM datasets. *BADGERS '12: Proceedings of the 2012 ACM Workshop on Building analysis datasets and gathering experience returns for security*, 17–24.
2. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953
3. Shalev-Shwartz, S., & Zhang, T. (2014). Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, *155*(1-2), 105–145. https://doi.org/10.1007/s10107-014-0839-0
4. Zadrozny, B., & Elkan, C. (2002). Transforming Classifier Scores into Accurate Multiclass Probability Estimates. *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 694–699. https://doi.org/10.1145/775047.775151
5. Saito, T., & Rehmsmeier, M. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLOS ONE*, *10*(3), Стаття e0118432. https://doi.org/10.1371/journal.pone.0118432
6. Lu, H., & Mazumder, R. (2020). Randomized Gradient Boosting Machine. *SIAM Journal on Optimization*, *30*(4), 2780–2808. https://doi.org/10.1137/18m1223277
7. Li, X., Moreschini, S., Zhang, Z., Palomba, F., & Taibi, D. (2023). The anatomy of a vulnerability database: A systematic mapping study. *Journal of Systems and Software*, 111679. https://doi.org/10.1016/j.jss.2023.111679

8. Niculescu-Mizil, A., & Caruana, R. (2005). Predicting good probabilities with supervised learning. *ICML '05: Proceedings of the 22nd international conference on Machine learning*, 625–632. https://doi.org/10.1145/1102351.1102430
9. Almahmoud, Z., Yoo, P. D., Damiani, E., Choo, K.-K. R., & Yeun, C. Y. (2025). Forecasting Cyber Threats and Pertinent Mitigation Technologies. *Technological Forecasting and Social Change*, *210*, 123836. https://doi.org/10.1016/j.techfore.2024.123836
10. Ferdous, J., Islam, R., Mahboubi, A., & Islam, M. Z. (2025). A Survey on ML Techniques for Multi-Platform Malware Detection: Securing PC, Mobile Devices, IoT, and Cloud Environments. *Sensors*, *25*(4), 1153. https://doi.org/10.3390/s25041153
11. Lyu, J., Bai, Y., Xing, Z., Li, X., & Ge, W. (2021). A Character-Level Convolutional Neural Network for Predicting Exploitability of Vulnerability. *International Symposium on Theoretical Aspects of Software Engineering (TASE)*, 119–126. https://doi.ieeecomputersociety.org/10.1109/TASE52547.2021.00014

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 2 (30), 2025
Спеціальний випуск ISSN 2663 – 4023
Special issue

Denysiuk Vladyslav,
postgraduate student
National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine
ORCID: 0009-0001-6334-1473
vladyslav.denysiuk@cs.khpi.edu.ua

PREDICTING CVE EXPLOITATION BASED ON NVD AND KEV OPEN DATA FOR RISK-ORIENTED PRIORITIZATION

Abstract. With the increasing number of publicly disclosed software vulnerabilities, security teams are increasingly challenged to identify key issues that require urgent remediation. While systems such as the Common Vulnerability Scoring System (CVSS) provide severity ratings, they do not indicate whether a vulnerability will be exploited in practice. The study proposes a machine learning-based approach to predict exploitable vulnerabilities using structured public data from the National Vulnerability Database (NVD) and the CISA-maintained Catalog of Known Functional Vulnerabilities (KEV). A labeled dataset of over 300,000 CVEs is generated, where randomly exploited ones are identified by KEV. The extracted features include CVSS vectors, CWE identifiers, vendor/product metadata, and time characteristics. Due to the extreme class imbalance (exploited CVEs are ~0.45%), an oversampling method and decision threshold tuning are used. Logistic regression trained in ML.NET is used to build interpretable models; it learns meaningful patterns that distinguish between exposed vulnerabilities. The threshold spectrum scoring demonstrates high completeness and increasing accuracy, offering a transparent and reproducible tool for prioritization. Additionally, the limitation associated with incomplete KEV as a "source of truth" is addressed and directions for improvement are directed: integration of NLP-embedding of CVE descriptions, probability calibration, and time-based validation to prevent data leakage. This approach increases the risk-based nature of cybersecurity decisions and can be otherwise integrated into vulnerability management processes in organizations of various scales.

Keywords: CVE, CVSS, KEV, exploitation prediction, machine learning, logistic regression, class imbalance, patch prioritization, ML.NET, cybersecurity.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Allodi, L., & Massacci, F. (2012). A preliminary analysis of vulnerability scores for attacks in wild: The EKITS and SYM datasets. *BADGERS '12: Proceedings of the 2012 ACM Workshop on Building analysis datasets and gathering experience returns for security*, 17–24.
2. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953
3. Shalev-Shwartz, S., & Zhang, T. (2014). Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, *155*(1-2), 105–145. https://doi.org/10.1007/s10107-014-0839-0
4. Zadrozny, B., & Elkan, C. (2002). Transforming Classifier Scores into Accurate Multiclass Probability Estimates. *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 694–699. https://doi.org/10.1145/775047.775151
5. Saito, T., & Rehmsmeier, M. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PLOS ONE*, *10*(3), Стаття e0118432. https://doi.org/10.1371/journal.pone.0118432
6. Lu, H., & Mazumder, R. (2020). Randomized Gradient Boosting Machine. *SIAM Journal on Optimization*, *30*(4), 2780–2808. https://doi.org/10.1137/18m1223277
7. Li, X., Moreschini, S., Zhang, Z., Palomba, F., & Taibi, D. (2023). The anatomy of a vulnerability database: A systematic mapping study. *Journal of Systems and Software*, 111679. https://doi.org/10.1016/j.jss.2023.111679

8. Niculescu-Mizil, A., & Caruana, R. (2005). Predicting good probabilities with supervised learning. *ICML '05: Proceedings of the 22nd international conference on Machine learning*, 625–632. https://doi.org/10.1145/1102351.1102430
9. Almahmoud, Z., Yoo, P. D., Damiani, E., Choo, K.-K. R., & Yeun, C. Y. (2025). Forecasting Cyber Threats and Pertinent Mitigation Technologies. *Technological Forecasting and Social Change*, *210*, 123836. https://doi.org/10.1016/j.techfore.2024.123836
10. Ferdous, J., Islam, R., Mahboubi, A., & Islam, M. Z. (2025). A Survey on ML Techniques for Multi-Platform Malware Detection: Securing PC, Mobile Devices, IoT, and Cloud Environments. *Sensors*, *25*(4), 1153. https://doi.org/10.3390/s25041153
11. Lyu, J., Bai, Y., Xing, Z., Li, X., & Ge, W. (2021). A Character-Level Convolutional Neural Network for Predicting Exploitability of Vulnerability. *International Symposium on Theoretical Aspects of Software Engineering (TASE)*, 119–126. https://doi.ieeecomputersociety.org/10.1109/TASE52547.2021.00014