АКТУАЛЬНІ ПИТАННЯ ТЕХНІЧНИХ НАУК

УДК 004.056.55
DOI https://doi.org/10.32782/2663-5941/2025.1.2/43

Дика А.І.
Національний університет «Одеська юридична академія»

Трофименко О.Г.
Національний університет «Одеська юридична академія»

Ковальжі А.С.
Національний університет «Одеська політехніка»

ІНТЕЛЕКТУАЛЬНЕ ВИЯВЛЕННЯ ФІШИНГОВИХ СТОРІНОК У СЕРЕДОВИЩІ ВЕБЗАСТОСУНКІВ ІЗ ЗАСТОСУВАННЯМ ВІЗУАЛЬНОГО ПОДАННЯ

Сучасний фішинг перетворився на одну з найскладніших кіберзагроз, яка поєднує методи соціальної інженерії з передовими технологіями штучного інтелекту. Сучасний фішинг характеризується стрімкою еволюцією та появою нових складних форм атак, серед яких: deepfake-фішинг, клонування легальних сайтів, фармінг тощо. Особливу небезпеку становлять саме вебсторінки, які є кінцевим ланцюжком більшості випадків фішингу, адже вони передбачають взаємодію користувача з підробленим вебінтерфейсом. Навіть попереднє виявлення фішингового повідомлення не гарантує безпеки, якщо користувач потрапляє на шкідливий сайт, візуально ідентичний легальному. Тому розробка ефективних методів аналізу саме вебсторінок, а не лише URL-адрес чи текстів, стає критично важливою складовою сучасних систем кібербезпеки. У статті подано новий підхід до виявлення фішингових вебресурсів, який ґрунтується на поєднанні алгоритмів машинного навчання (метод опорних векторів, SVM) з інноваційною технікою графічного подання HTML-коду. Основна ідея методу полягає в перетворенні байтової послідовності вебсторінки на зображення з подальшим виділенням ключових візуальних ознак, що дозволяє виявляти навіть складні випадки імітації легальних сайтів. Експериментальні дослідження показали високу ефективність запропонованого підходу – точність класифікації досягла 92.45% при збалансованих показниках precision (0.92) та recall (0.92). Особливу цінність методу становить його здатність аналізувати структуру сторінки незалежно від мови розмітки або специфіки кодування, що суттєво розширює сферу застосування порівняно з традиційними текстовими методами. Порівняльний аналіз із сучасними аналогами показує конкурентні переваги запропонованого рішення. На відміну від методів, які аналізують лише URL-адреси або текстовий вміст, наш підхід дозволяє виявляти складні види фішингу, зокрема клоновані сайти та інші форми візуальної імітації. Важливою практичною перевагою є можливість інтеграції методу в наявні системи кібербезпеки, як-от веббраузери або серверні рішення для моніторингу трафіку, без значних змін в інфраструктурі. Результати дослідження свідчать, що запропонований метод є перспективним інструментом для боротьби з сучасними фішинговими атаками, особливо за умов стрімкого розвитку генеративних технологій. Подальші напрямки роботи можуть стосуватися оптимізації алгоритму для роботи в реальному часі, розширення набору ознак для виявлення нових типів атак та інтеграції з системами аналізу поведінки користувачів. Запропонований метод може бути ефективно інтегрований у різні вебзастосунки та системи. Наприклад, його можна впровадити у вигляді браузерного розширення для онлайн-перевірки сторінок безпосередньо під час перегляду, серверного модуля для сканування підозрілих посилань у корпоративних мережах, API-сервісу для інтеграції з наявними системами безпеки електронної пошти, компонента вебфільтрації в маршрутизаторах або проксі-серверах. Швидкість обробки (за рахунок алгоритму SVM) дозволяє використовувати його в системах реального часу без суттєвого впливу на продуктивність застосунків. Крім того, метод може бути доповнений механізмами колективного захисту, автоматично оновлюючи бази даних фішингових сайтів на основі результатів аналізу.

Ключові слова: *штучний інтелект, ШІ, машинне навчання, SVM, візуалізація даних, вебзастосунок, фішинг, кібербезпека, аналіз вебсторінок, HTML-код, класифікація.*

Постановка проблеми. Фішинг є однією з найпоширеніших форм сучасних кіберзагроз, що активно використовуються зловмисниками для несанкціонованого доступу до конфіденційної інформації. Так, за даними досліджень [1] 2024 року 94% організацій стикалися з фішинговими атаками. Значний вплив має фішинг через електронну пошту, який є джерелом 90% атак програм-вимагачів. При цьому середній розмір збитків від однієї фішингової атаки на підприємство становить $4.65 мільйона, а глобальні фінансові втрати від фішингових атак за 2024 рік оцінили у $17.4 мільярда, що на 45% більше проти попереднього року [2].

З появою штучного інтелекту (ШІ) у кіберзлочинців з'явилася можливість створювати цільові фішингові кампанії, використовуючи генеративні моделі та алгоритми обробки природної мови. Генеративні можливості сучасних великих мовних моделей (Large Language Model, LLM), генеративно-змагальних мереж (Generative Adversarial Networks, GAN), моделі синтезу мовлення (Text-to-Speech, TTS) та клонування голосу, моделі обробки природної мови (Natural Language Processing, NLP) з контекстною адаптацією, моделі комп'ютерного зору (Computer Vision, CV) в комплексі суттєво розширили арсенал засобів, які використовують зловмисники у фішингових атаках. Зокрема, автоматизоване створення переконливих фішингових повідомлень, deepfake-контенту (аудіо, відео, зображень), а також фальшивих профілів користувачів дозволяє зловмисникам з високим ступенем правдоподібності імітувати людську поведінку. Це, у свою чергу, суттєво ускладнює виявлення таких загроз традиційними методами. Саме тому 43% користувачів не здатні правильно ідентифікувати фішинговий електронний лист, що свідчить про критичну потребу як у покращенні навчання з питань кібергігієни, так і в розробленні інноваційних автоматизованих систем виявлення фішингових атак [3].

Аналіз останніх досліджень і публікацій. Дослідженню проблем виявлення кібератак за допомогою алгоритмів ШІ та машинного навчання присвячені численні публікації науковців і дослідників. Так, дослідження [4] спрямоване на аналіз впливу технологій машинного навчання на зміну та адаптацію кіберзагроз, зокрема на способи використання ШІ зловмисниками для підвищення ефективності, масштабу та складності атак. Автори статті [5] запропонували методику виявлення зловмисних команд, прихованих у графічних файлах за допомогою стеганографічних методів. Робота [6] висвітлює різні методи машинного навчання для тестування на проникнення. У дослідженні [7] запропоновано архітектуру та методологію впровадження системи Threat Guard AI для пом'якшення цифрових загроз. Дослідження [8] розглядає комбінований метод з алгоритмами глибокого навчання, машинного навчання та ройового інтелекту для виявлення фішингових атак. У роботі [9] запропоновано до розгляду програмний застосунок, який для аналізу введених URL-адрес щодо виявлення потенційно небезпечних посилань використовує три алгоритми машинного навчання: випадковий ліс (Random Forest, RF), логістичної регресії (Logistic Regression) та опорно-векторного кластерування на основі методу опорних векторів (Support Vector Machine, SVM). У дослідженні [10] для виявлення фішингових вебсайтів використано дещо інший набір алгоритмів: бустинг градієнта (Gradient Boosting), RF, CatBoost та логістичної регресії. Автори статті [11] для навчання розробленої ними моделі для виявлення у мережевому трафіку різного роду кібератак на вебсайти використали відкритий набір даних CSE-CIC-IDS2017 та алгоритми машинного навчання: RF, наївний баєсів класифікатор, k-найближчих сусідів (K-Nearest Neighbor, KNN), SVM з використанням гауссівського ядра, адаптивний бустинг, бустинг градієнта.

Тим не менш, незважаючи на велику кількість публікацій, присвячених боротьбі з фішинговими атаками, дану проблему не можна назвати вирішеною остаточно, про що свідчить безперервне зростання збитків, спричинених саме цим видом атак.

Метою даної роботи є підвищення ефективності виявлення фішингових атак шляхом поєднання машинного навчання (SVM) та інноваційного підходу до візуалізації HTML-коду.

Виклад основного матеріалу

1. ***Класифікація поширених видів фішингу***

Фішингові атаки ґрунтуються на соціальній інженерії, оскільки зловмисники використовують фальсифіковані повідомлення, здебільшого у вигляді електронної пошти або комунікацій у соціальних мережах, з метою спрямування користувача на спеціально створені фішингові вебресурси. За статистичними даними щомісяця створюється приблизно 1,5 мільйона нових фішингових вебсайтів [12]. Такі вебресурси зазвичай видають себе за відомі компанії, як-от Microsoft та Google, обманом змушуючи людей надавати особисту інформацію. Імітуючи легітимні сторінки авторизації або онлайн-форми вони здійснюють

несанкціоноване збирання конфіденційної інформації. Отримані дані в подальшому використовуються для реалізації шахрайських схем або крадіжки особистих відомостей. Так, 79% випадків захоплення облікових записів розпочинались саме з фішингових електронних листів [13]. Крім того, гіперпосилання, вміщені у подібних повідомленнях, нерідко слугують каналом доставки шкідливого програмного забезпечення (ПЗ) до інформаційної системи користувача.

Розрізняють різні види фішингу (рис. 1).

– *Фішинг електронною поштою (Email Phishing)* є класичним найпоширенішим різновидом, з яким зіткнулося вже 86% компаній у всьому світі [1]. Шахрайство через електронну пошту зросло на 70% проти минулого року, завдяки здатності ШІ імітувати тон, підробляти внутрішні електронні листи та персоналізувати повідомлення з вражаючою точністю [14].

– *Смішинг (SMS Phishing)* є фішинговою шахрайською схемою через текстові повідомлення, які спонукають людину перейти за посиланням, завантажити шкідливе ПЗ або надати особисту інформацію. Популярність смішингу зумовлена тим, що повідомлення виглядають переконливо та вимагають миттєвих дій, що є приманкою і робить зайняту на роботі людину легкою здобиччю. Поширено зловмисники використовують фальшиві повідомлення про неоплачені збори або доставки, щоб спонукати користувачів перейти за шкідливими посиланнями. Приблизно три з чотирьох організацій у всьому світі стикалися з цим типом атаки, причому 2024 року кількість зареєстрованих смішингових атак зросла на 25% [15].

– *Вішинг (Vishing)* або голосовий фішинг є шахрайством через телефонні дзвінки або голосові повідомлення, щоб обманом виманити у людей конфіденційні дані. Наприклад, зловмисник телефонує, представляється працівником банку та намагається дізнатися PIN-код або CVV-код картки. Зловмисники використовують інструменти на базі генеративного ШІ, щоб видавати себе за керівників компаній або державних службовців, обманюють своїх жертв, змушуючи їх думати, що вони розмовляють з кимось, кого вони знають і кому довіряють [16]. Цей вид фішингу є класичним прикладом атак соціальної інженерії. Сім з десяти організацій стикалися з шахрайськими схемами вішингу, причому 2024 року кількість атак вішингу на фахівців компаній зросла на 28% [1].

– *Цільовий фішинг (Spear Phishing)* є атакою на конкретну людину або організацію задля збору особистої інформації. Наприклад, персоналізований фішинг-лист, надісланий бухгалтеру з підробленої адреси нібито від керівника з використанням імен, посад, деталей про компанію. Такий лист від «вашого колеги» з проханням подивитися важливий договір (а насправді – шкідливий файл) виглядає максимально переконливим і спрямований він на те, щоб змусити жертву клікнути на шкідливе посилання, відкрити файл або розкрити конфіденційні дані. Не дивно, що 65% успішних фішингових атак пов'язані саме з цільовим фішингом, причому 2024 року спостерігалося збільшення цього виду атак на 30% [17].

– *Фішинг у соціальних мережах (Social Media Phishing)* задля того, щоб отримати логін/пароль або змусити встановити шкідливе ПЗ, використовує фальшиві профілі або повідомлення від друзів у Facebook, Instagram тощо. Приблизно 33% спроб фішингу 2024 року були спрямовані через соцмережі [3].

Рис. 1. Поширені різновиди фішингу

– *Клонування фішингу (Clone Phishing)* стосується копіювання реального листа, наприклад, з повідомленням від банку, але посилання чи вкладення в ньому замінені на шкідливі. Приблизно 19% зареєстрованих фішингових електронних листів є спробами клонування фішингу, причому їхня кількість 2024 року зросла на 25% проти попереднього року [18].

– *Фармінг (Pharming)* використовує переспрямування користувачів з легітимних вебсайтів на підроблені шкідливі. Атаки фармінгу часто використовують вразливості в DNS-серверах або шкідливе ПЗ на пристрої користувача і перенаправляють користувачів з легітимних вебсайтів на підроблені шкідливі. 2024 року 12% фішингових атак використовували цю тактику, здебільшого спрямовану на вебсайти онлайн-банкінгу або електронної комерції [3].

– *Фішинг «людина посередині» (Man-in-the-Middle, MitM Phishing)* – це різновид атак з перехопленням даних вебсесії, наприклад, між користувачем та онлайн-платіжним сервісом, який намагається обдурити кожну з двох сторін, змусивши їх думати, що він є іншою стороною, захоплює та/або змінює інформацію, яка передається між жертвою та законною стороною, приміром, під час доступу до онлайн-банкінгу, коли зловмисник може перехопити запити або змінити дані в реальному часі (особливо у незахищених мережах Wi-Fi). Атаки MitM часто використовують вразливості в мережах Wi-Fi або скомпрометованих маршрутизаторах. Зловмисники можуть використовувати методи перехоплення пакетів, захоплення сеансів і підміни HTTPS для перехоплення каналів зв'язку та отримання несанкціонованого доступу до конфіденційних даних. Атаки MitM стосуються перехоплення та маніпулювання зв'язком між двома сторонами без їхнього відома. При цьому зазвичай жертва навіть не здогадується, що її з'єднання перехоплено, а тому це вкрай небезпечний тип атак. Наприклад, користувач здійснює великий платіж надійному постачальнику, але стороння особа перехоплює транзакцію та непомітно змінює банківські реквізити. Користувач вважає, що платіж безпечний, але його перенаправляють на рахунок злочинця. 14% випадків фішингу припадає на цей тип атак, коли зловмисники перехоплюють зв'язок між жертвою та довіреною особою і в реальному часі крадуть дані, зокрема під час онлайн-банківських транзакцій. 2024 року атаки MitM становили 23% кіберінцидентів, пов'язаних з ідентифікацією [19]. Через свою примарну природу їх легко недооцінювати, але не варто цього робити. Атаки MitM можуть призвести до різних злочинних дій: від масштабного шахрайства до захоплення облікових записів та фінансових втрат.

– *Хмарний фішинг (Cloud Phishing)* є серйозною проблемою, оскільки все більше компаній покладаються на хмарні сервіси. Такого роду атака для отримання облікових даних імітує інтерфейси Google Drive, Microsoft OneDrive, Dropbox тощо. Наприклад, користувач отримує посилання, схоже на сторінку входу в Google Workspace або Microsoft 365, де пропонується підтвердити доступ і ввести пароль. Але при цьому відбувається переспрямування на підроблену сторінку входу до хмарного сервісу. 37% фішингових атак 2024 року були спрямовані саме на хмарні сервіси.

– *Фішинг через QR-коди (Quishing)* популярний через повсюдне поширення QR-кодів. Зловмисник поширює підроблений QR-код у ресторанах, листівках, оголошеннях тощо. Після сканування користувач переходить на шкідливий сайт або дає дозвіл на встановлення ПЗ. При цьому здебільшого відбувається обхід фільтрів безпеки, оскільки зазвичай системи не аналізують QR-коди так, як звичайні URL. Наразі приблизно 5% сучасних атак здійснюються через підроблені QR-коди, які спрямовують користувача на шкідливі вебсайти, а 2024 року їхня кількість зросла на 25% проти попереднього року [20].

– *Бренд-фішинг (Brand Phishing)* є видом атак, коли зловмисники імітують відомий бренд (наприклад, Google, PayPal, Microsoft, Netflix, Apple), тобто створюють підроблені вебсайти, логотипи, домени тощо. Метою є переконання користувача у тому, що він взаємодіє з офіційним джерелом, щоб отримати його облікові дані або змусити здійснити платіж. За даними Check Point, 2024 року 40% фішингових атак були пов'язані з імітацією відомих брендів: Microsoft у фішингових атаках склав 32% усіх [21].

– *Deepfake-фішинг* – новий, небезпечний вид фішингу, в якому використовуються штучно згенеровані відео, аудіо або зображення. Зловмисники, застосовуючи генеративний ШІ ChatGPT, Sora, ElevenLabs, імітують голос або зовнішність відомих осіб, зокрема керівників компаній, державних службовців або колег. Така атака може здійснюватися через відеодзвінки, голосові повідомлення або навіть Zoom-конференції. Через розвиток ШІ 2024 року кількість deepfake-атак зросла на понад 50% [22].

Зважаючи на зазначені статистичні дані, фішинг є однією з найбільш актуальних та небез-

печних кіберзагроз. Використання ШІ значно підвищує ефективність фішингових атак, робить їх більш переконливими та важкими для виявлення. Тому важливою складовою стратегії кібербезпеки є впровадження комплексних заходів захисту, включно з навчанням користувачів, використанням багатофакторної автентифікації та постійним оновленням систем безпеки.

Як видно з наведеної класифікації, значна частина сучасних фішингових атак передбачає взаємодію користувача саме з вебінтерфейсом – підробленою вебсторінкою, яка імітує легітимний ресурс (банківський сайт, хмарний сервіс, корпоративний портал тощо). Такі фішингові сайти є центральною ланкою багатьох типів атак: клацнувши на посилання у фішинговому листі, повідомленні чи QR-коді, користувач потрапляє на візуально правдоподібну, але шкідливу вебсторінку. Згідно з даними за 2024 рік понад 70% фішингових атак у підсумку зводяться до взаємодії жертви з підробленим вебінтерфейсом. Це означає, що навіть попереднє виявлення фішингового повідомлення не гарантує повної безпеки, якщо користувач все ж відкриє сайт і візуально не зможе відрізнити його від справжнього.

2. Запропонований підхід до подання даних під час виявлення фішингового контенту та його подальшого аналізу

Одним із сучасних та перспективних напрямів у галузі аналізу структурованих даних є подання вихідного коду або вебсторінок у формі зображень з подальшим застосуванням методів машинного навчання [23]. Такий підхід дозволяє інтерпретувати код як візуальну структуру, де просторові та текстурні закономірності можуть нести інформативні ознаки, релевантні для класифікації або виявлення аномалій. На відміну від традиційних текстових або синтаксичних аналізаторів, візуальне подання відкриває можливість виявлення шаблонів на основі глобального вигляду документа – його «візуального сліду». Це особливо актуально для задач виявлення фішингових сайтів, де важливу роль відіграє зовнішня схожість зі справжніми ресурсами, а не лише семантичний вміст коду. Подання коду як зображення дозволяє ефективно використовувати перевірені архітектури нейронних мереж, зокрема згорткові нейронні мережі (CNN), а також традиційні алгоритми машинного навчання, які працюють з візуальними ознаками [24].

Для забезпечення можливості аналізу вебсторінок та подальшої візуалізації їхніх характеристик доцільно здійснити попереднє перетворення HTML-документів у бінарне подання. Попри те, що HTML-код має текстову природу, саме перехід до бітового або байтового рівня дозволяє усунути залежність від синтаксичних особливостей мови розмітки, а також створити уніфіковане подання вхідних даних. Робота з бінарною формою дає змогу виявляти низькорівневі аномалії, регулярності або приховані шаблони, які можуть залишитися непоміченими при аналізі на рівні символів або тегів. Крім того, бінарне подання є зручним для перетворення на матрицю або зображення, що відкриває можливості для застосування методів машинного навчання, орієнтованих на роботу з візуальними структурами.

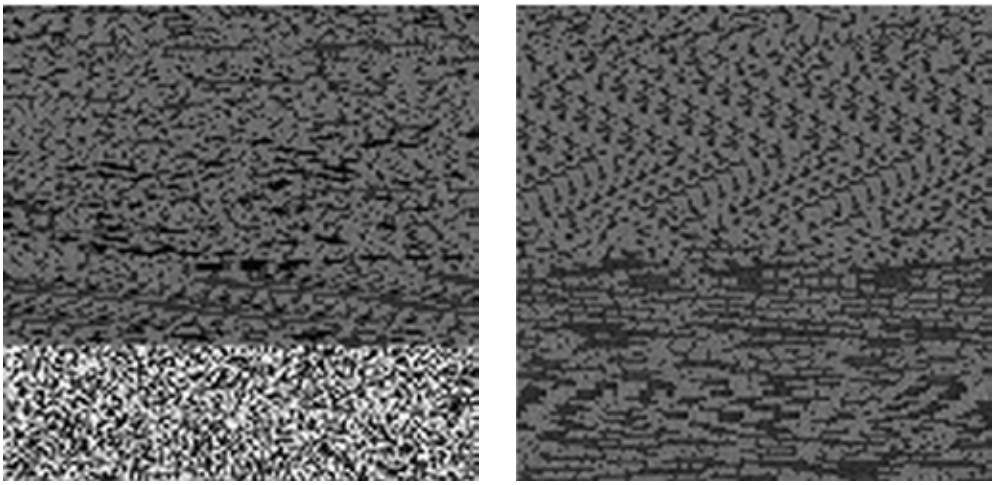
На наступному етапі виконують візуалізацію HTML-файлу. Байтовий потік перетворюється у масив типу NumPy зі значеннями від 0 до 255, кожне з яких відповідає одному байту. Кожен байт коду подається як сірий піксель у форматі RGB (R = G = B), тобто без кольорової інформації. Отримані пікселі формують зображення розміром 128×128 пікселів, що містить 16 384 значення.

На рис. 2 наведено приклад подання фішингового та легітимного файлів у вигляді картинок.

У разі перевищення кількості вихідних даних для формування зображення зазначеного розміру – надлишок обрізається. Тим самим забезпечується єдиний розмір вхідних зображень для подачі на класифікатор.

Після побудови зображення відбувається процедура нормалізації: значення пікселів перетворюються так, щоб мати нульове математичне сподівання та одиничне стандартне відхилення. Це підвищує ефективність подальшої класифікації. Для аналізу використовуються методи машинного навчання, зокрема SVM, який працює зі сформованими векторами ознак, отриманими із зображень.

Після отримання візуального подання HTML-файлу треба здійснити перетворення зображення на вектор ознак, придатний для подачі на вхід моделі машинного навчання. З метою підвищення інформативності аналізу, зображення розміром 128×128 пікселів розбивається на чотири горизонтальні області: верхню, нижню та дві центральні. Така сегментація дозволяє локалізовано досліджувати візуальні патерни, зокрема в центральній частині зображення, де зазвичай концентруються важливі елементи сторінки, як-от: JavaScript-код, функціональні блоки та логіка взаємодії. Натомість початок і кінець HTML-файлу часто містять службові теги або метаінформацію, які не завжди є релевантними для класифікації. Розбиття також



Файл fishtest.html URL: https://www.instagram.com/

Рис. 2. Порівняння візуалізації фішингового та легітимного файлів

дозволяє моделі краще диференціювати регіональні особливості структури сторінки, що може суттєво впливати на якість розпізнавання фішингових шаблонів. Приклад візуального розмежування областей наведено на рис. 3.

Для кожної з виділених горизонтальних областей зображення обчислюється нормалізована гістограма розподілу яскравості пікселів. Нормалізація гістограми здійснюється шляхом ділення кількості пікселів у кожному біні на загальну кількість пікселів у відповідній області, що забезпечує масштабну незалежність ознак. Отримані гістограми з усіх чотирьох областей об'єднуються в єдиний вектор ознак довжиною 1024 елементи, що дозволяє зберегти локальні особливості структури HTML-файлу в різних його сегментах. На наступному етапі вектор ознак піддається стандартизації з використанням відповідної функції нормалізації (скейлера) – трансформації, яка зводить кожну ознаку до нульового середнього значення та одиничного стандартного відхилення. Така стандартизація дозволяє усунути вплив масштабних розбіжностей між окремими компонентами вектора, покращує стабільність роботи моделі та забезпечує її узагальнюючу здатність під час обробки вхідних даних, отриманих з різних HTML-документів. Нормалізований вектор ознак є фінальним результатом етапу перетворення даних і подається на вхід моделі ШІ для подальшої класифікації чи прогнозування.

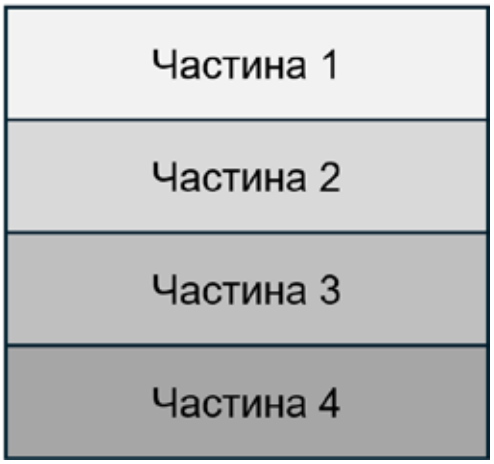


Рис. 3. Розбиття зображення на області

3. Метод виявлення фішингового контенту та навчання нейронної мережі

Ґрунтуючись на запропонованому підході до подання даних сформуємо метод виявлення фішингового контенту у вигляді конкретних кроків.

Крок 1. Отримання вхідних даних. Користувач подає HTML-файл або посилання на вебсторінку. У разі посилання вебсторінка завантажується та зберігається у вигляді HTML-документа.

Крок 2. Визначення та уніфікація кодування. Визначається кодування HTML-файлу. Якщо кодування відмінне від UTF-8 (наприклад, Windows-1251 або KOI8-U), файл перекодується у UTF-8.

Крок 3. Перетворення HTML-коду у байтовий потік. HTML-файл конвертується у потік байтів (значення від 0 до 255). Цей потік буде базою для подальшої візуалізації.

Крок 4. Формування зображення. Кожному байту відповідає сірий піксель формату RGB (R=G=B). Пікселі розміщуються лінійно у зображенні розміром 128×128 пікселів (загалом 16 384 байти). Якщо байтів менше 16 384,

зображення доповнюється байтами із значенням 255. Якщо байтів більше 16 384, надлишок обрізається.

Крок 5. Нормалізація зображення. Яскравість пікселів (значення 0–255) нормалізується: середнє значення дорівнює 0; стандартне відхилення дорівнює 1.

Крок 6. Сегментація зображення. Зображення поділяється на 4 горизонтальні області: верхня, нижня, дві центральні.

Крок 7. Побудова гістограм. Для кожної з 4-х областей обчислюється нормалізована гістограма яскравості пікселів. Гістограма відображає частотний розподіл пікселів за яскравістю. Кожен бін нормалізується відносно загальної кількості пікселів в області.

Крок 8. Формування вектора ознак. Усі чотири гістограми об'єднуються в один вектор ознак довжиною 1024 елементи. Цей вектор відображає візуальну структуру HTML-документа у стислому числовому вигляді.

Крок 9. Стандартизація вектора ознак. Застосовується функція нормалізації (скейлер): кожна ознака трансформується до середнього, яке дорівнює 0 та стандартного відхилення, яке дорівнює 1. Це дозволяє моделі ефективно навчатися на однорідних ознаках.

Крок 10. Класифікація. Отриманий нормалізований вектор подається на вхід моделі машинного навчання (наприклад, SVM).

Відзначимо, що метод може бути використаний у практичних системах виявлення шкідливих вебсторінок, зокрема, у вигляді вбудованого модуля безпеки в браузер, який у фоновому режимі візуалізує HTML-вміст і на основі його зображення оцінює потенційно шкідливу активність. Завдяки перетворенню коду на зображення забезпечується незалежність від конкретної мови розмітки, що розширює можливості застосування моделі у гетерогенних середовищах.

Для побудови класифікатора була використана модель типу Support Vector Machine з ядром RBF. З відкритого джерела [25] отримано 3000 легітимних та 1700 фішингових HTML-файлів. Після попередньої обробки 4700 файлів було об'єднано в єдиний датасет з доданими мітками класів. Отриманий датасет розділено за пропорцією 80/20, де 80% – вектори, на яких модель буде навчатися виявляти патерни, і 20% – вектори, на яких буде проведено тестування точності моделі.

З метою вибору оптимальних параметрів налаштування моделі було проведено Grid-search з кросвалідацією, тобто навчання різних версій моделі з різними параметрами C, gamma та ядра. Після проведення пошуку виявлено, що набір найкращих параметрів має такий вигляд:

$$P = \{'C': 100,\ 'gamma': 'scale',\ 'kernel': 'rbf'\}.$$

Реалізацію та результат пошуку найкращих налаштувань наведено на рис. 4.

Після остаточного навчання моделі SVM, використовуючи оптимальні параметри було досягнуто точності класифікації у 92.45%. Обидва класи даних демонструють схожі значення метрик precision, recall та f1-міри зі значенням ~0.92, що свідчить про коректно налаштований баланс класів, а саме:

1) recall (повнота) – відображає частку фішингових сайтів, які модель змогла успішно виявити та віднести до класу «1»;

2) precision (точність) – відображає, яка частка прикладів, класифікованих як фішингові, дійсно є такою;

3) f1-міра – зважений показник для оцінки балансу між помилками першого та другого роду.

Навчена модель коректно виявляє більшість випадків обох класів, при цьому показник False Positive Rate становить 0.0676, а False Negative Rate – 0.0841.

Для розуміння структури отриманих векторів ознак та перевірки наявності розділюваності між класами (наприклад, шкідливими та легітимними HTML-файлами), доцільно провести їхню візуалізацію у двовимірному просторі. Оскільки початкові вектори мають високу розмірність (1024 ознаки), перед візуалізацією доцільно зменшити розмірність до 2D-простору. Це дозволяє

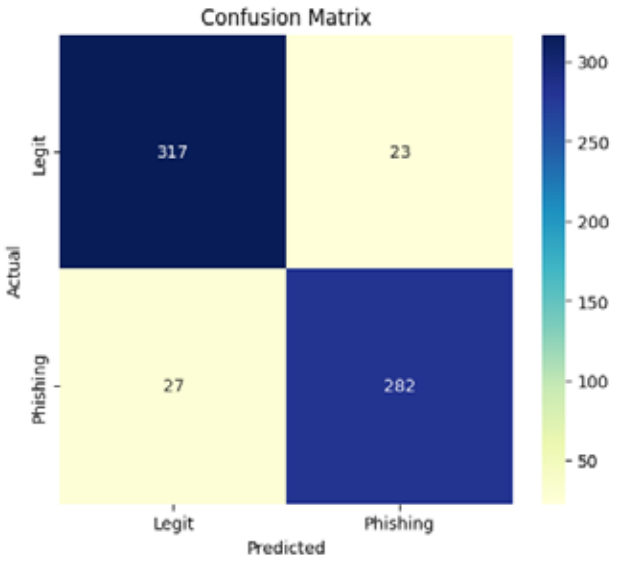


Рис. 4. Матриця конфузії

зберегти ключові характеристики розподілу даних та виявити можливі кластери чи закономірності. Для цього використовують методи зменшення розмірності, наприклад t-distributed Stochastic Neighbor Embedding (t-SNE), які зберігають глобальну або локальну структуру даних, відповідно.

На рис. 5 наведено приклад такої візуалізації: кожна точка відповідає одному HTML-документу, кольором позначено клас (шкідливий / легітимний). Ця ілюстрація дозволяє наочно оцінити якість отриманих ознак і потенціал моделі до класифікації. Ця візуалізація показує розподіл фішингових і легітимних векторів на окремі кластери, що підтверджує здатність класифікатора розпізнавати відмінності між класами.

У табл. 1 наведено порівняльний аналіз запропонованого методу із сучасними методами, які демонструють високі результати в цій сфері. Для цього порівняно характеристики кількох поширених підходів, зокрема заснованих на аналізі URL, ознаках DOM, методах обробки природної мови та повноцінному рендерингу сторінок. У порівнянні враховувалися такі характеристики методів: точність класифікації, вимоги до ресурсів, чутливість до обходу захисту та придатність до інтеграції в реальні системи (наприклад, у браузери або у пошукові проксі). Це дозволяє об'єктивно оцінити переваги запропонованої моделі та обґрунтувати її практичне застосування як ефективного компонента інфраструктури кіберзахисту.

Запропонований метод на основі SVM та візуалізації HTML демонструє конкурентну точність проти сучасних аналогів, при цьому вирізняється низкою ключових переваг: швидкістю обробки, незалежністю від текстового контенту та можливістю виявляти візуально схожі підроблені сайти. На відміну від методів, які аналізують лише URL [27] або текстовий вміст [28], даний підхід забезпечує комплексний аналіз структури сторінки, що робить його менш вразливим до адаптивних атак. Хоча окремі алгоритми демонструють вищу точність (до 98.12%, як в [26]), вони мають суттєві обмеження – залежність від ознак URL, високу ресурсомісткість або нездатність виявляти клоновані сайти. Тому запропоноване рішення є оптимальним балансом між точністю, продуктивністю та універсальністю для реальних умов застосування.

Висновки. Основні результати проведених досліджень:

1. Виконана класифікація фішингових атак та аналіз сучасних тенденцій у фішингових атаках свідчить про їхню еволюцію та збільшення складності. Застосування ШІ та новітніх технологій

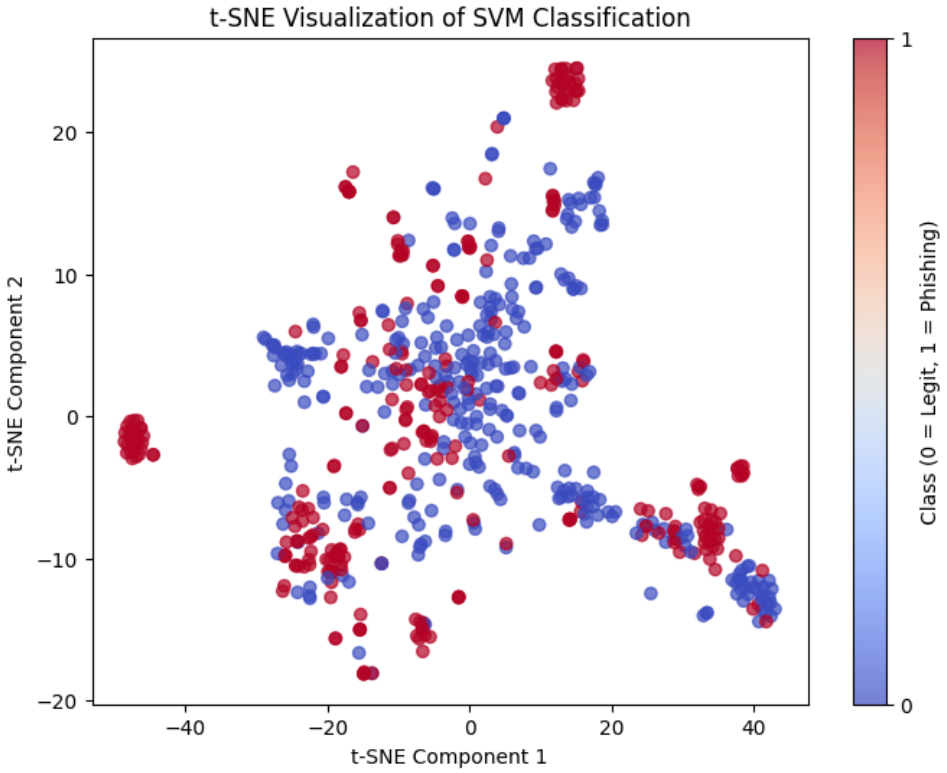


Рис. 5. Візуалізація розподілу векторів ознак

Таблиця 1

Порівняльний аналіз запропонованого методу із сучасними аналогами

Підхід	Точність	Примітки
Запропонований підхід SVM + візуалізація	92.45%	Висока швидкість виконання, побайтовий аналіз контенту сторінки, гнучкість щодо можливої імплементації
Гібридне машинне навчання на основі URL-адреси [26]	98.12%	Метод аналізує тільки URL-адресу (довжину, наявність символів, домен, піддомен тощо). Це робить його вразливим до фішингових сайтів, які використовують легітимну або замасковану URL-адресу. Не аналізує вміст сторінки. Не можна виявляти клон-сайти й атаки через візуальну схожість. Має меншу стійкість до адаптивних атак
Функція на основі URL та HTML з використанням моделі штучних нейронних мереж (ANN) з MobileBERT [27]	86…96%	Метод вимагає значні обчислювальні потужності (часто задіюється GPU), що утруднює його реальне застосування. Структура сторінки може враховуватися не повністю через особливості реалізації методу
Машинне навчання [28]	82.7…98.3%	Метод ґрунтується на аналізі тексту, що залишає зловмисникові простір для можливих маніпуляцій, структура файлу не враховується
Машинне навчання [29]	89.2%	Не застосовується аналіз структури сторінки

дозволяє зловмисникам створювати переконливі, важкі для виявлення фішингові схеми. Надалі організаціям необхідно впроваджувати комплексні стратегії кібербезпеки, включно з навчанням користувачів, використанням багатофакторної автентифікації та постійним оновленням систем безпеки, щоб ефективно протистояти загрозам.

2. Запропонований метод поєднує машинне навчання на основі SVM з інноваційним графічним поданням байтів вебсторінки, демонструє високу ефективність із точністю класифікації до 92.45%. Його ключова особливість полягає у перетворенні HTML-коду на візуальне подання, що дозволяє аналізувати структуру сторінки через призму графічних шаблонів. Такий підхід усуває обмеження традиційних текстових методів аналізу та забезпечує адаптивність до різноманітних типів фішингових атак, включно зі складними випадками імітації легальних ресурсів. Метод відрізняється практичною спрямованістю – він може бути ефективно інтегрований у сучасні браузери або серверні системи для оперативної перевірки вебконтенту. Його універсальність забезпечується аналізом на рівні байтів, що дозволяє виявляти аномалії незалежно від специфіки кодування чи структури документа. Важливою перевагою є збалансованість результатів, де високі показники точності та повноти свідчать про здатність системи мінімізувати кількість помилкових спрацьовувань та пропусків загроз. Цей підхід відкриває нові можливості для боротьби з сучасними фішинговими атаками, особливо за умов стрімкого розвитку генеративних технологій, які ускладнюють виявлення штучно створених шкідливих ресурсів. Його практична цінність полягає у поєднанні високої точності, адаптивності та можливості впровадження в реальні системи кібербезпеки.

3. Розробка ефективного методу виявлення фішингових HTML-сторінок може розглядатися як один із компонентів загальної інфраструктури боротьби з фішингом. У реальних умовах подібні методи інтегруються у більш складні системи – зокрема, вони можуть бути частиною веббраузера, розширенням до нього або системою серверного захисту вебзастосунків. Така інтеграція дозволяє забезпечити своєчасну перевірку підозрілих сторінок під час перегляду користувачем та запобігти переходу на потенційно небезпечні ресурси. Крім безпосереднього блокування шкідливих сторінок, подібна система може автоматично формувати чорні списки URL-адрес, виявляти шаблони фішингових атак, а також надсилати сповіщення розробникам браузера чи відповідним правоохоронним або аналітичним структурам. Це дозволяє не лише захищати окремих користувачів, а й оперативно реагувати на масові кампанії фішингу, оновлювати захисні механізми та сприяти координації зусиль у сфері кібербезпеки на міжорганізаційному рівні.

Загалом механізми автоматичного виявлення фішингових сторінок є невід'ємною частиною загальної безпеки вебзастосунків і цифрового простору в цілому. Їх використання дозволяє створювати багаторівневий захист, який охоплює як кінцевого користувача, так і серверну інфраструктуру вебресурсів.

Список літератури:

1. Phishing. Statistics & Facts. URL: https://www.statista.com/topics/8385/phishing/
2. Top 5 Phishing Statistics 2025. URL: https://www.broadband4europe.com/phishing-statistics/
3. Phishing Statistics by Demographic, Healthcare, Industry and Country. URL: https://www.sci-tech-today.com/stats/phishing-statistics-updated/
4. Habrylchuk A. V., Susukailo V. A., Kurii Y. O., Vasylyshyn S. I. Analysis of Cyber Attacks Using Machine Learning on the Information Security Management Systems. *Computer systems and networks.* 2025. Vol. 7, No 1. P. 68-78. DOI: https://doi.org/10.23939/csn2025.01.068
5. Denysiuk D., Sorochynskyi O., Hnatchuk Y., Drozd A. Information technology for detecting malicious codes in information systems based on parallel process analysis. *Herald of Khmelnytskyi National University. Technical Sciences*. 2025. Vol 347. No. 1. P. 576–583. DOI: https://doi.org/10.31891/2307-5732-2025-347-80
6. Zhuravchak A., Piskozub A. Analysis of machine learning methods for automating penetration testing. *Electronic Professional Scientific Journal «Cybersecurity: Education, Science, Technique»*. 2025. Vol. 3, No. 27. P. 54–62. DOI: https://doi.org/10.28925/2663-4023.2025.27.711
7. Bhinge Pr. Threat Guard AI: A Multi-Domain System for Intelligent Fraud Detection Using Machine Learning. *International Journal for Research in Applied Science and Engineering Technology*. 2025. Issue 13. P. 2985–2998. DOI: https://doi.org/10.22214/ijraset.2025.70866.
8. Albahadili A., Akbas A., Rahebi J. Detection of phishing URLs with deep learning based on GAN-CNN-LSTM network and swarm intelligence algorithms. *Signal, Image and Video Processing.* 2024. Issue 18. P. 1–17. DOI: https://doi.org/10.1007/s11760-024-03204-2.
9. Burova N., Oprysk R., Kurii Y., Lakh Y., Susukailo V. (2024). Machine learning as a key tool in defensive cyber operations: the effectiveness of phishing threat detection. *Social Development and Security*. 2024. Vol. 14, No. 5. P. 113-123. https://doi.org/
10. Adane K., Beyene B., Yimer M. Intelligent Phishing Website Detection before and after Multiple Informative Feature Selection Techniques: Machine Learning Approach. *International Journal of Information Science and Management*. 2024. Issue 22. P. 31-62. DOI: https://doi.org/10.22034/ijism.2023.1977974.0.
11. Prokopov V., Meleshko Ye., Yakymenko M., Reznichenko V., Shymko S. The methods of data storing of a recommendation system based on linked lists. *Control, Navigation and Communication Systems. Academic Journal.* 2022. Vol. 2, No. 68. P. 79–84. DOI: https://doi.org/10.26906/SUNZ.2022.2.079
12. Phishing Statistics. 2024. URL: https://truelist.co/blog/phishing-statistics/
13. Most impactful stats from the 2024 Email Security Risk Report. URL: https://www.egress.com/blog/company-news/stats-from-the-email-security-risk-report
14. AI is making phishing emails far more convincing with fewer typos and better formatting: Here's how to stay safe. URL: https://www.techradar.com/pro/security/ai-is-making-phishing-emails-far-more-convincing--with-fewer-typos-and-better-formatting-heres-how-to-stay-safe
15. Business Email Compromise Statistics 2025. URL: https://hoxhunt.com/blog/business-email-compromise-statistics
16. Трофименко О. Г., Задерейко О. В., Логінова Н. І., Шевченко О. В. Інструменти штучного інтелекту в розробці графічних елементів, 3D-моделей та інших компонентів відеоігор. *Інформатика та математичні методи в моделюванні*. 2025. Т. 15, № 2. С. 276–287. DOI: https://doi.org/10.15276/imms.v15.no2.276
17. Top 15 Phishing Stats to Know in 2024. URL: https://news.trendmicro.com/2024/07/22/phishing--stats-2024
18. Phishing Attacks Double in 2024. URL: https://www.infosecurity-magazine.com/news/2024-phishing-attacks-double/
19. 6 Ways to Prevent Man-in-the-Middle (MitM) Attacks. URL: https://www.memcyco.com/6-ways-to-prevent-man-in-the-middle-mitm-attacks/
20. The Latest 2025 Phishing Statistics (updated June 2025). URL: https://aag-it.com/the-latest-phishing-statistics/
21. Phishing Trends Report (Updated for 2025). URL: https://hoxhunt.com/guide/phishing-trends-report
22. Exploring Q4 2024 Brand Phishing Trends: Microsoft Remains the Top Target as LinkedIn Makes a Comeback. URL: https://blog.checkpoint.com/research/exploring-q4-2024-brand-phishing-trends-microsoft-remains--the-top-target-as-linkedin-makes-a-comeback/
23. Baptista I., Shiaeles S., Kolokotronis N. A Novel Malware Detection System Based on Machine Learning and Binary Visualization. *2019 IEEE International Conference on Communications Workshops (ICC Workshops).* 20-24 May 2019, Shanghai, China. P. 1–6. DOI: https://doi.org/10.1109/ICCW.2019.8757060
24. Трофименко О.Г., Лобода Ю.Г., Дика А.І., Мільченко О.О., Стрілець М.І. Штучний інтелект у системному аналізі. *Таврійський науковий вісник. Серія: Технічні науки»*. 2024. № 5. С. 85–97. DOI: https://doi.org/10.32782/tnv-tech.2024.5.9

25. VirusShare.com, Because Sharing is Caring. URL: https://virusshare.com/about

26. Abdul K., Mobeen Sh., Khabib M., et al. Phishing Detection System Through Hybrid Machine Learning Based on URL. *IEEE Access.* 2023. Vol. 11. P. 36805–36822. DOI: https://doi.org/10.1109/access.2023.3252366

27. Shirazia H., Haynesb K., Raya I. Towards performance of NLP transformers on URL-based phishing detection for mobile devices. *Journal of Ubiquitous Systems & Pervasive Networks*. 2022. Vol. 17, No. 1. P. 35–42. DOI: https://doi.org/10.5383/JUSPN.03.01.000

28. Sahingoz O. K, Buber E., Demir O., Diri B. Machine learning based phishing detection from URLs. *Expert Syst. Appl*. 2019. Vol. 117. P. 345–357. DOI: https://doi.org/10.1016/j.eswa.2018.09.029

29. Shahrivari V., Darabi M., Izadi M. Phishing detection using machine learning techniques. *arXiv*. 2020. Vol. 11116. P. 1–9. DOI: https://doi.org/10.48550/arXiv.2009.11116

Dyka A.I., Trofymenko O.G., Kovalzhi A.S. INTELLIGENT DETECTION OF PHISHING PAGES IN WEB APPLICATION ENVIRONMENT USING VISUAL REPRESENTATION

Modern phishing has become one of the most complex cyber threats, combining social engineering methods with advanced artificial intelligence technologies. Modern phishing is characterized by rapid evolution and the emergence of new complex forms of attacks, such as deepfake phishing, cloning of legal sites, pharming, and others. Of particular danger are web pages, which are the final link of most phishing cases and involve user interaction with a fake web interface. Even preliminary detection of a phishing message does not guarantee safety if the user ends up on a malicious site that is visually identical to a legal one. Therefore, the development of effective methods for analyzing web pages, and not just URLs or texts, is becoming a critically important component of modern cybersecurity systems. This paper presents a new approach to detecting phishing web resources, based on a combination of machine learning algorithms (support vector machine learning, SVM) and an innovative technique for the graphical representation of HTML code. The main idea of the method is to transform the byte sequence of a web page into an image, followed by the selection of key visual features, which enables the detection of even complex cases of imitation of legitimate sites. Experimental research has shown the high efficiency of the proposed approach; the classification accuracy reached 92.45% with balanced precision (0.92) and recall (0.92) indicators. A special value of the method is its ability to analyze the structure of the page regardless of the markup language or coding specificity, which significantly expands the scope of application compared to traditional text methods. Comparative analysis with modern analogues demonstrates the competitive advantages of the proposed solution. Unlike methods that analyze only URL addresses or text content, our approach allows us to detect complex types of phishing, including cloned sites, and other forms of visual imitation. An important practical advantage is the ability to integrate the method into existing cybersecurity systems, such as web browsers or server-based solutions for traffic monitoring, without significant changes in the infrastructure. The results of the research indicate that the proposed method is a promising tool for combating modern phishing attacks, especially in the context of the rapid development of generative technologies. Further directions of work include optimizing the algorithm for real-time operation, expanding the set of features to detect new types of attacks, and integrating with user behavior analysis systems. The proposed method can be effectively integrated into various web applications and systems. For example, it can be implemented as a browser extension for online page checking directly during browsing, a server module for scanning suspicious links in corporate networks, an API service for integration with existing email security systems, or a web filtering component in routers or proxy servers. The processing speed (due to the SVM algorithm) allows it to be used in real-time systems without a significant impact on application performance. In addition, the method can be supplemented with collective protection mechanisms, automatically updating databases of phishing sites based on the analysis results.

Key words: *artificial intelligence, AI, machine learning, SVM, data visualization, web application, phishing, cybersecurity, web page analysis, HTML code, classification.*