Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА

Безкоровального Ярослава Олеговича

на здобуття ступеня вищої освіти магістра

Метод ідентифікації фішингових атак в електронних листах

Галузь знань 12 – Інформаційні технології
Спеціальність 125 – Кібербезпека та захист інформації
Освітня програма Кібербезпека та захист інформації

Шифр КРМКБЗІ.2301131.23.01.01 ПЗ

Виконав студент 2 курсу група КБЗІм-23-1 Ярослав БЕЗКОРОВАЛЬНИЙ
Керівник докт. техн. наук, професор Михайло КАСЯНЧУК
Нормоконтролер старший викладач Сергій МОСТОВИЙ

До захисту допускаю:
Завідувач кафедри кібербезпеки Юрій КЛЬОЦ

16 12 2024 р.

Хмельницький 2024

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Кібербезпеки

Рівень вищої освіти Магістр

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека та захист інформації

Освітня програма Кібербезпека та захист інформації

ЗАТВЕРДЖУЮ

Завідувач кафедри кібербезпеки

Юрій КЛЬОЦ

2 09 2024 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Безкоровальному Ярославу Олеговичу

1 Тема роботи Метод ідентифікації фішингових атак в електронних листах

Керівник роботи докт.техн.наук, професор Михайло КАСЯНЧУК

Затверджено наказом ректора університету 26 08 2024 № 60

2 Строк подання студентом кваліфікаційної роботи на кафедру 2.12.2024

3 Вихідні дані до роботи Проаналізувати поняття фішингу та засоби їх виявлення. Обрати параметри електронного листа та здійснити попередню обробку набору даних. Розробити модель ідентифікації фішингових атак. Розробити метод ідентифікації фішингових атак в електронних листах. Здійснити оцінку ефективності розробленого методу.

4 Зміст пояснювальної записки (перелік питань, які потрібно розробити)
Вступ. Аналіз фішингових атак. Постановка задачі. Вибір параметрів електронних листів та підготовка набору даних. Модель ідентифікації фішингу. Метод ідентифікації фішингових атак в електронних листах. Розрахунок ефективності пропонованого методу.

5 Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

6 Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7 Дата видачі завдання 2 09 2024 р.

КАЛЕНДАРНИЙ ПЛАН

Назва етапів (розділів) кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
Дослідження проблеми виявлення фішингових атак		Виконано
Визначення змісту, структури кваліфікаційної роботи		Виконано
Аналіз життєвого циклу атаки та наявні рішення щодо виявлення		Виконано
Опрацювання статті за результатами дослідження		Виконано
Розоробка моделі виявлення фішингових електронних листів та вибір параметрів для реалізації		Виконано
Розробка методу виявлення фішингових атак велектронних листах		Виконано
Оцінка ефективності розробленого методу		Виконано
Підготовка та опрацювання ілюстративного матеріалу		Виконано
Оформлення магістерської роботи графічної та текстової частини		Виконано
Попередній захист магістерської роботи		Виконано
Захист магістерської роботи на засіданні ЕК		Виконано

Студент _____ Ярослав БЕЗКОРОВАЛЬНИЙ

Керівник кваліфікаційної роботи _____ Михайло КАСЯНЧУК

АНОТАЦІЯ

Тема кваліфікаційної роботи: Метод ідентифікації фішингових атак в електронних листах

Автор роботи: студент групи КБЗІм-23-1 Безкоровальний Я.О.

Керівник роботи: докт. техн. наук, професор Касянчук М.М.

Загальний обсяг роботи: 93 сторінок, 16 рисунків, 10 таблиць, 1 додаток, 60 посилань.

Ключові слова: фішингові атаки, електронна пошта, кібербезпека, методи виявлення.

Дана кваліфікаційна робота присвячена розробці та впровадженню методу ідентифікації фішингових атак в електронних листах, що має на меті підвищити рівень захисту користувачів від кіберзагроз. У процесі дослідження проаналізовано життєвий цикл фішингових атак, особливості атак нульового дня, а також сильні та слабкі сторони існуючих методів захисту. Запропонований метод базується на поєднанні машинного навчання з евристичним аналізом тексту і структурних елементів електронних листів, що забезпечує виявлення загроз із високою точністю. Практичне значення роботи полягає у можливості інтеграції розробленого методу в антивірусні системи, поштові сервери та інші засоби захисту, що знижує ризик витоку конфіденційної інформації та фінансових втрат.

2.12.2024

ANNOTATION

Theme of qualification work: Method of identifying phishing attacks in e-mails

Author of the work: student of KBZIm-23-1 Bezkorovalnyi Ya.O.

Mentor: Dr. Technical Sciences, Professor Kasyanchuk M.M.

Total volume of work: 93 pages, 16 figures, 10 tables, 1 appendix, 60 references.

Keywords: phishing attacks, e-mail, cyber security, detection methods.

This qualification work is devoted to the development and implementation of a method of identifying phishing attacks in e-mails, which aims to increase the level of protection of users against cyber threats. During the research, the life cycle of phishing attacks, features of zero-day attacks, as well as strengths and weaknesses of existing protection methods were analyzed. The proposed method is based on a combination of machine learning with heuristic analysis of text and structural elements of e-mails, which ensures detection of threats with high accuracy. The practical value of the work lies in the possibility of integrating the developed method into anti-virus systems, mail servers and other means of protection, which reduces the risk of leakage of confidential information and financial losses.

2.12.2024

ЗМІСТ

ВСТУП

Фішинг є однією з найпоширеніших форм кіберзлочинності, яка щороку завдає значних збитків як приватним особам, так і великим організаціям у різних секторах. Головна мета фішингових атак – отримання конфіденційної інформації користувачів, включаючи паролі, фінансові дані, особисті та корпоративні відомості, які зловмисники можуть використати для фінансового шахрайства, компрометації облікових записів, або отримання несанкціонованого доступу до систем. Незважаючи на значний прогрес у сфері інформаційної безпеки, фішинг продовжує залишатися серйозною загрозою, оскільки зловмисники постійно адаптують свої методи, щоб обійти захисні системи, створюючи все складніші та переконливіші атаки.

Фішингові атаки сьогодні відбуваються в багатьох формах, включаючи електронну пошту, текстові повідомлення, соціальні мережі та навіть телефонні дзвінки. Проте електронна пошта залишається одним із головних каналів для розповсюдження фішингових загроз. Шахрайські електронні листи зазвичай містять фальшиві посилання, які на перший погляд виглядають як легітимні. Однак вони ведуть на підроблені сайти, де користувачів обманом змушують вводити свої облікові дані або іншу конфіденційну інформацію. Додатково, фішингові листи часто використовують техніки соціальної інженерії, щоб викликати у користувачів відчуття терміновості або страху, стимулюючи їх до необдуманих дій.

Проблема виявлення фішингових атак ускладнюється тим, що зловмисники постійно вдосконалюють свої методи, використовуючи соціальну інженерію, інструменти для маскування посилань і підробки вебсторінок, які виглядають практично так само, як офіційні. Це означає, що класичні методи кіберзахисту, такі як чорні списки та антивірусні програми, не завжди можуть ефективно запобігти фішинговим атакам, особливо якщо мова йде про атаки нульового дня, які базуються на нових вразливостях, ще не відомих системам захисту. Більше того, атаки нульового дня зазвичай характеризуються коротким

життєвим циклом фішингових сайтів, що дозволяє зловмисникам використовувати один ресурс обмежений час, але дуже інтенсивно. Водночас, кожна фішингова атака є досить масштабованою — фішингові листи можуть бути надіслані сотням і тисячам користувачів, і якщо навіть невеликий відсоток адресатів надасть свої дані зловмисникам, це вже призведе до значних втрат.

В умовах зростаючої кількості фішингових атак є потреба у вдосконаленні систем виявлення та запобігання шахрайству, які могли б адаптуватися до змінних тактик зловмисників. Відповідно, існує потреба в нових методах захисту, які використовували б сучасні технології обробки даних та аналізу текстів, зокрема евристичні підходи та штучний інтелект. Враховуючи швидкі темпи еволюції фішингу, зростає актуальність досліджень, що орієнтуються на розробку методів, здатних працювати з новими і невідомими раніше схемами шахрайства. Ці методи мають враховувати багатоаспектність фішингових атак, які не обмежуються одними лише посиланнями на підроблені сайти, а можуть включати візуальні підказки, обманні повідомлення і навіть емоційний тиск на користувача.

Об'єктом дослідження в даній роботі є фішингові атаки, які здійснюються через електронну пошту з метою викрадення конфіденційної інформації користувачів. Фішинг залишається серйозною кіберзагрозою, особливо у сфері комунікацій через електронні листи, де зловмисники застосовують різні техніки соціальної інженерії та підроблені посилання. Метою зловмисників є обман користувачів та схилення їх до розкриття особистих даних, таких як паролі, фінансова інформація, дані про банківські рахунки тощо. Саме через широке використання електронної пошти у бізнесі та особистих комунікаціях, об'єкт дослідження набуває критичної актуальності.

Предметом дослідження є методи та алгоритми, що забезпечують виявлення фішингових атак у електронних листах. Сюди входить аналіз структури фішингових листів, виділення ключових ознак, які можуть вказувати на шахрайство, а також розробка класифікаційних моделей, здатних автоматично розпізнавати фішинг. Важливою частиною дослідження є застосування

алгоритмів машинного навчання, евристичних методів та обробки текстів для створення системи, яка може ідентифікувати фішингові електронні листи, незалежно від їх унікальних або нових технік маскування.

Наукова новизна даного дослідження полягає у поєднанні методів обробки текстів, аналізу даних та алгоритмів машинного навчання для створення комплексної системи, здатної точно виявляти фішингові листи навіть у разі атак нульового дня. Розроблений метод використовує сучасні алгоритми для аналізу електронних листів та виділення ознак, що підвищують ймовірність ідентифікації фішингових загроз. Важливою особливістю є здатність методу адаптуватися до нових загроз, завдяки чому вона залишається ефективною навіть за умов швидкої еволюції технік фішингу. Також дослідження пропонує новий підхід до оптимізації ознак, що дозволяє зменшити обсяг оброблюваних даних без втрати точності, що є значним внеском у розробку інтелектуальних систем кібербезпеки.

Практичне значення роботи полягає в можливості інтеграції розробленого методу в існуючі поштові сервери, системи кібербезпеки та антивірусні платформи для виявлення та запобігання фішинговим атакам. Він може значно знизити ризик витоку конфіденційної інформації та фінансових втрат для користувачів та організацій, які використовують електронну пошту для комунікації. Запропонований метод також може слугувати основою для створення більш складних систем захисту від фішингу, які можуть бути адаптовані для інших каналів зв'язку, таких як соціальні мережі або месенджери, розширюючи таким чином можливості захисту від кіберзагроз.

1 АНАЛІЗ ФІШИНГОВИХ АТАК

1.1 Визначення фішингу та його життєвий цикл

Основна мета фішингових атак полягає у викраденні конфіденційної інформації користувачів Інтернету, такої як паролі, номери соціального страхування та дані кредитних карт. Фішинг є серйозною загрозою, яка щорічно завдає бізнесу збитків на мільярди доларів.

Термінологія, що стосується фішингу, має різні визначення в науковій літературі, від описових до наукових та загальних. У попередніх дослідженнях фішинг не завжди чітко визначався, інколи його описували на прикладах або ж автори припускали, що читачі вже знають, що таке фішинг, чи вважали, що точні визначення будуть надто складними для розуміння. Багато дослідників запропонували власні визначення фішингу, що призвело до великої кількості різних визначень у науковій літературі. Найпоширеніші визначення наведені нижче:

– за визначенням Оксфордського словника англійської мови, фішинг – це «шахрайська практика надсилання електронних листів, що нібито надходять від надійних компаній, щоб змусити людей розкрити особисту інформацію, таку як паролі та номери кредитних карт»[1];

– PhishTank, один з основних центрів співпраці щодо даних про фішинг в Інтернеті, визначає фішинг як «шахрайська спроба, зазвичай через електронну пошту, викрасти вашу особисту інформацію»[2];

– APWG, найбільш відома організація, що координує глобальні зусилля в боротьбі з кіберзлочинністю, дає довше визначення: фішинг – це «кримінальний механізм, який використовує як соціальну інженерію, так і технічні хитрощі для викрадення особистих даних споживачів і фінансових реквізитів [3]. Схеми соціальної інженерії використовують підроблені електронні листи, які начебто походять від законних компаній і установ, щоб перенаправити споживачів на підроблені веб-сайти, які обманом змушують їх розкрити свої фінансові дані, такі як імена користувачів і паролі»;

– відповідно до Закону України "Про електронні комунікації" від 16 грудня 2020 року N 1089-IX, фішинг - неправомірні дії в мережі Інтернет, наслідком яких є або може бути виманювання персональних даних та інших даних абонентів, в тому числі реквізитів платіжних карток та паролів, ідентифікаційних номерів, номерів банківських рахунків тощо [4].

Ці визначення свідчать про те, що різні організації розробили власні тлумачення терміну «фішинг». Отож, фішинг – це масштабована дія обману, де використовується маскування для отримання інформації від цілі [5-7]. Це визначення вважається найбільш точним і узагальненим, охоплюючи всі види фішингових атак.

Типова фішингова атака починається з надсилання електронного листа клієнту. Це є початком атаки, яку пропонується зупинити шляхом захисту користувача від обману. Приклад фішингового листа показаний на рисунку 1.1.

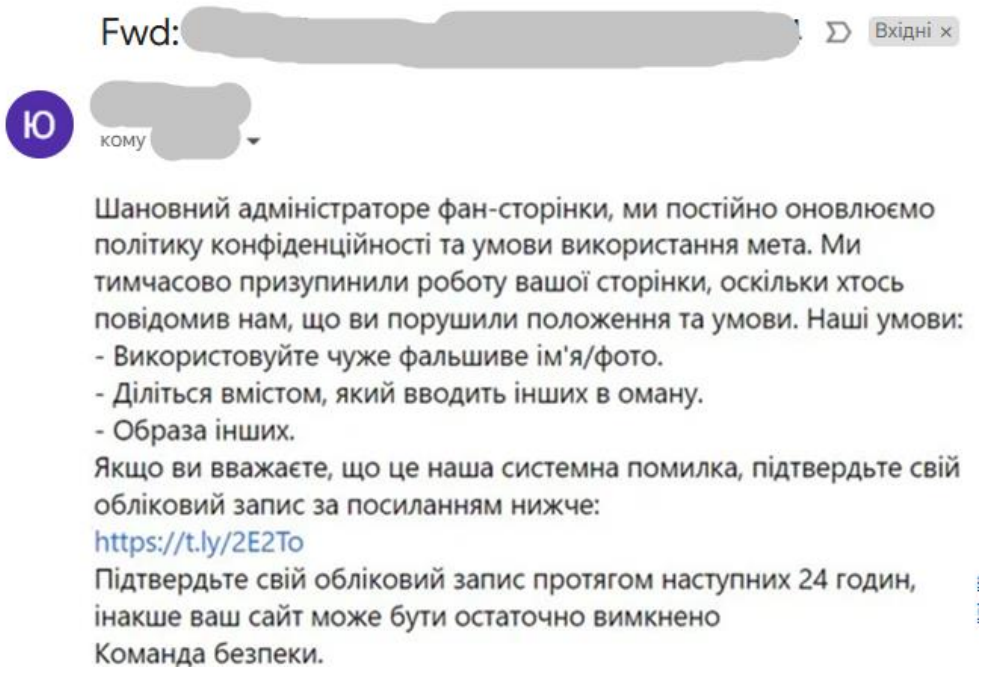

Рисунок 1.1 - Приклад фішингового електронного листа

Його основними складовими є (рис. 1.2) [8-10]:

– шаблонні, загальні привітання, як-от "Шановний користувач", замість конкретного імені одержувача. Це ознака того, що лист надсилається масово і не

адресований конкретному отримувачу;

– сайт не має захищеного протоколу HTTPS (відображається як "http://" замість "https://"), оскільки HTTPS шифрує дані під час їх передачі між користувачем і сайтом, тому фішингові сайти часто використовують небезпечний протокол HTTP, щоб отримати доступ до даних;

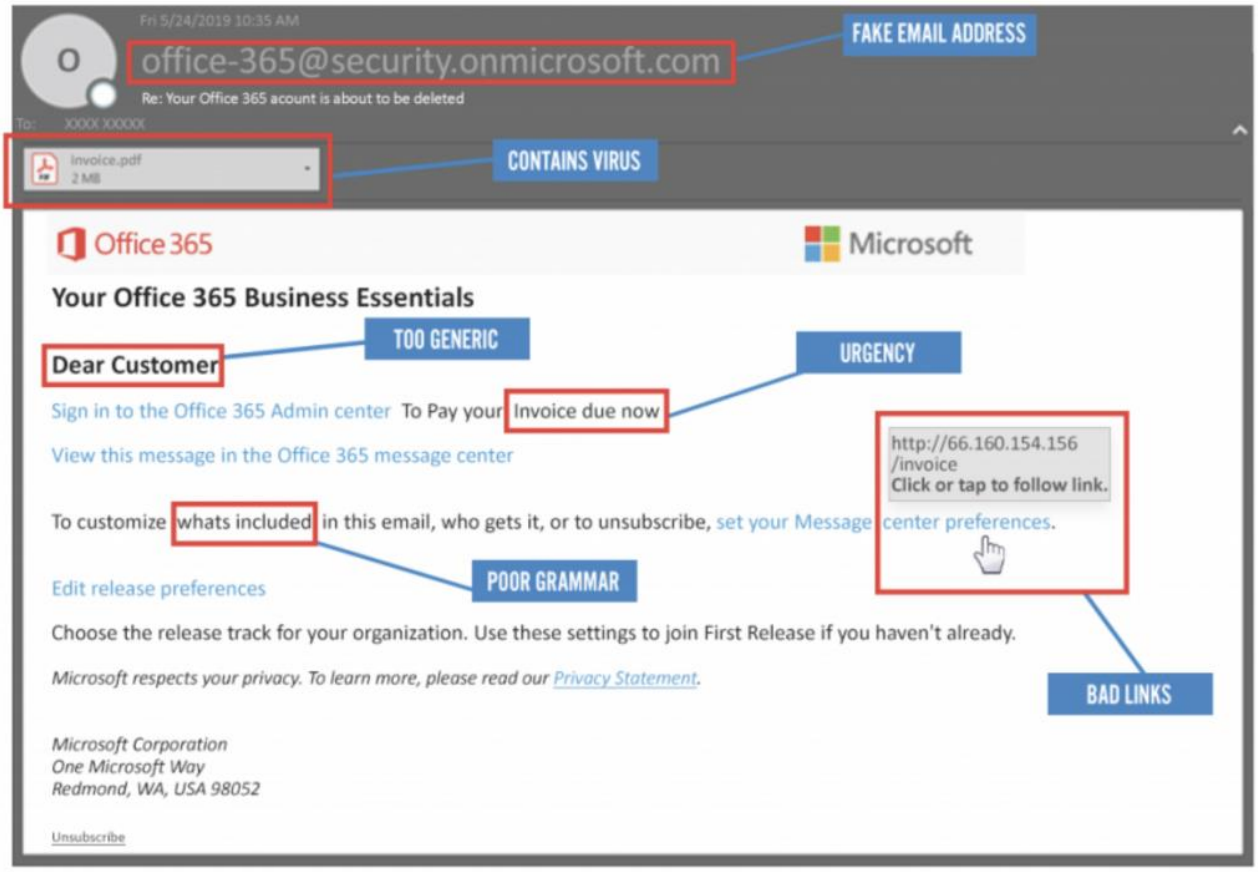

Рисунок 1.2 - Приклад фішингового електронного листа з описами його основних частин

– просять надати конфіденційну інформацію, таку як паролі, номери кредитних карток або іншу особисту інформацію на відміну від легітимних компаній, як правило, ніколи не запитують таку інформацію через електронну пошту;

– відправник може бути вказаний дуже загально або використовувати незрозумілу або підроблену адресу. Наприклад, від імені відомого банку, але адреса відправника виглядає підозріло або не відповідає офіційним доменам

компанії;

– у фішингових листах часто є фальшиві посилання, які виглядають як справжні, але насправді перенаправляють на шахрайські сайти. Такі посилання можуть містити неправильно написані URL або приховувати фальшиві веб-адреси за звичними словами, як-от "натисніть тут";

– фішингові листи намагаються створити відчуття терміновості, щоб змусити діяти швидко. Вони можуть стверджувати, що акаунт буде заблоковано або що відбувся несанкціонований доступ до облікового запису, щоб змусити негайно реагувати, не думаючи про наслідки.

Підроблене посилання в листі веде на фальшивий сайт, як показано на рисунку 1.3. Цей сайт розроблено таким чином, щоб виглядати як оригінальний веб-сайт.

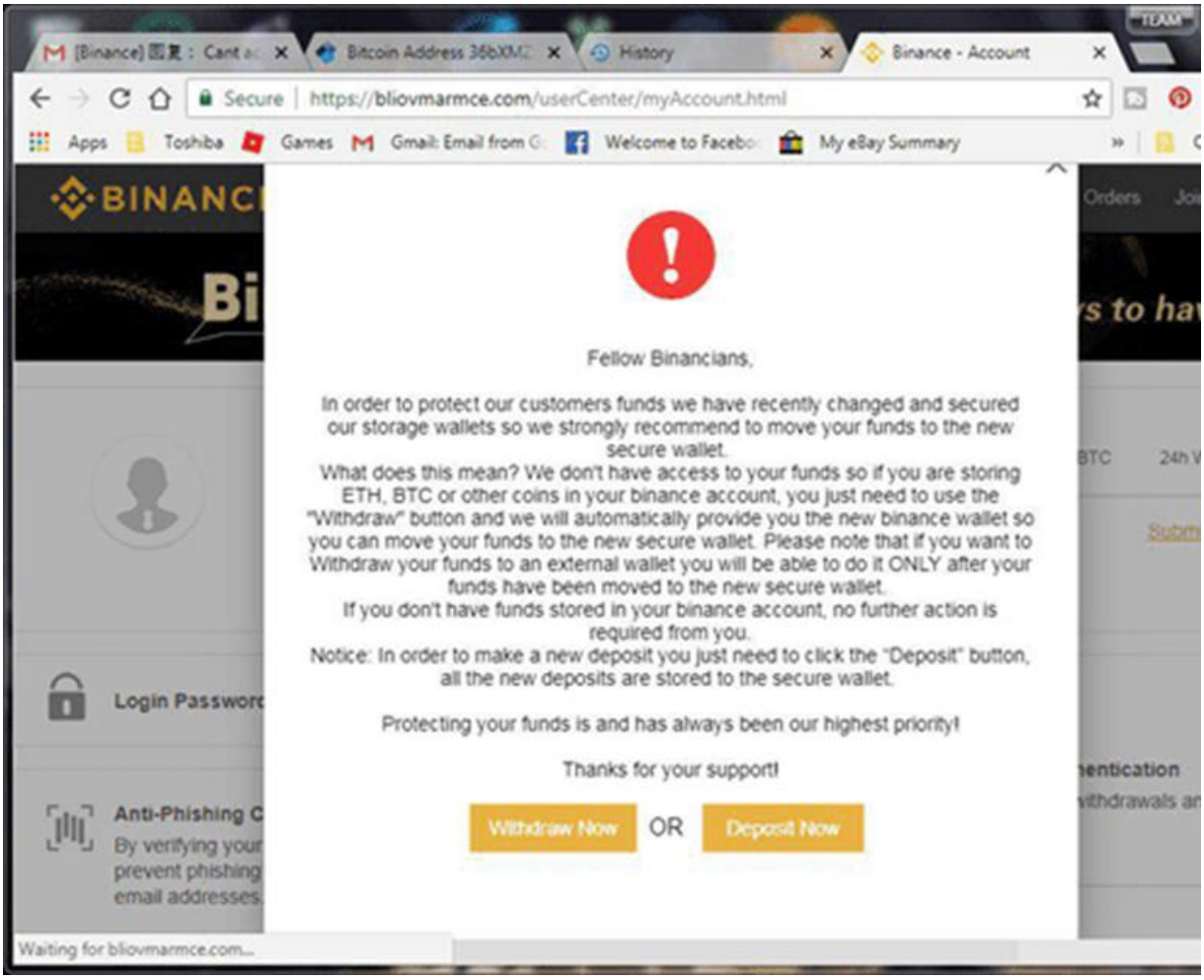

Рисунок 1.3 - Фішинговий веб-сайт

Життєвий цикл фішингової атаки нагадує процес риболовлі (рис. 1.4), звідки й походить слово «фішинг» [11-13]. Рибалка готує приманку, щоб риба добровільно клюнула, і так само зловмисник створює приманку у вигляді фішингового веб-сайту. Потім він надсилає електронного листа з посиланням на цей сайт багатьом користувачам і чекає, доки хтось із них введе свої дані на підробленому сайті. Далі фішер збирає інформацію жертви, таку як дані банківського рахунку, логіни до соцмереж або електронної пошти, та використовує ці дані для крадіжки грошей чи інформації.

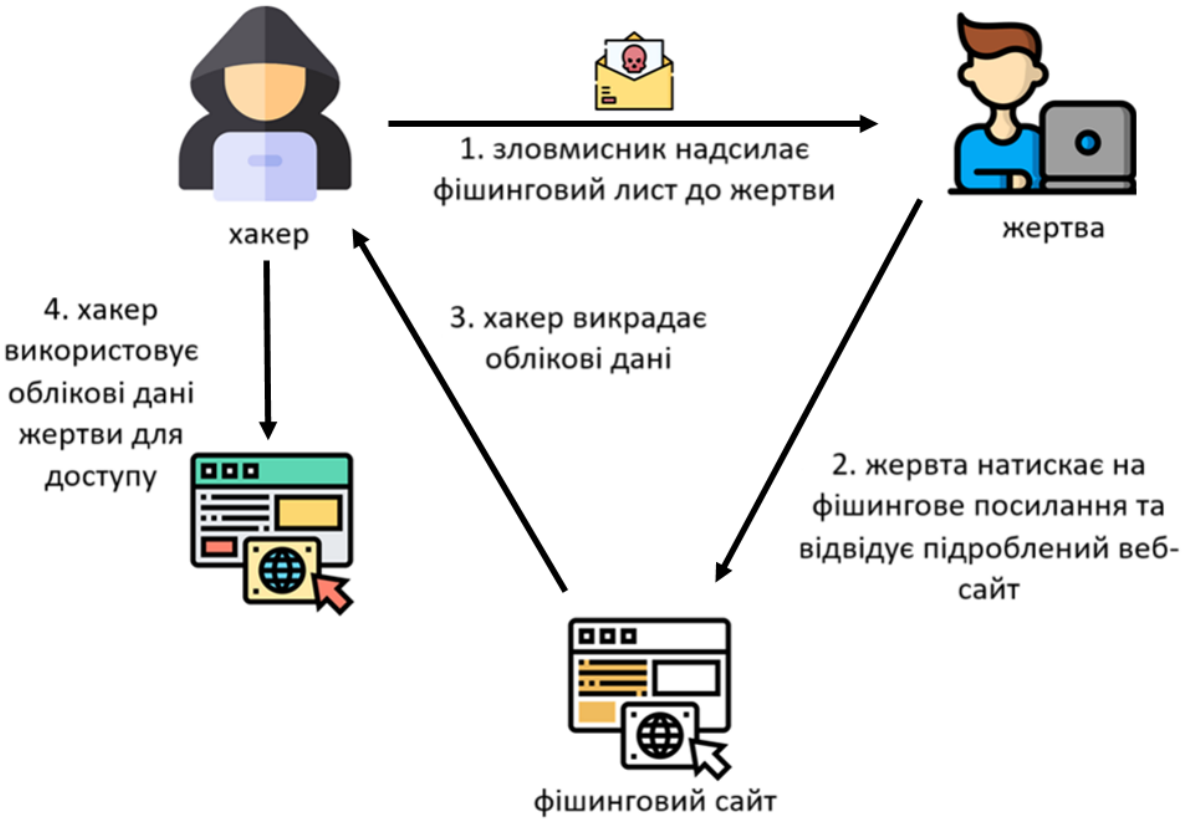

Рисунок 1.4 – Схема фішингової атаки

Фішери використовують різноманітні техніки для фішингових атак. Основні методи включають електронну пошту, SMS, соціальні мережі, VoIP, рекламу, миттєві повідомлення, пошукові системи та шкідливі веб-сайти [14-16]. Вони постійно вдосконалюють свої методи, щоб використовувати всі доступні засоби для досягнення жертв. Найпоширенішим каналом для таких атак є електронна пошта.

1.2 Фішингові атаки нульового дня

Фішингові атаки нульового дня — це такі типи атак, які використовують нові методи чи вразливості, ще не відомі інструментам захисту або фахівцям з безпеки. На відміну від традиційних фішингових атак, що базуються на відомих шаблонах або техніках і можуть бути ефективно виявлені за допомогою наявних інструментів кібербезпеки, атаки нульового дня вимагають більш гнучких та динамічних підходів для виявлення та протидії.

Одна з основних проблем, пов'язаних з виявленням фішингових атак нульового дня, полягає у швидкоплинності самих атак. За статистикою, середній час існування фішингового вебсайту складає всього 21 годину [17-19]. За цей короткий період фішинговий сайт може встигнути обманути велику кількість користувачів до того, як він буде виявлений та внесений до чорного списку. Такий короткий життєвий цикл робить техніки, засновані на чорних списках, малоефективними.

Фішинг нульового дня часто використовує нові домени або IP-адреси, які ще не знаходяться у чорних списках. Ці нові ресурси маскуються під легітимні або відомі вебсайти, використовуючи схожі URL-адреси чи зовнішній вигляд для того, щоб ввести користувачів в оману. Відсутність інформації про такі нові ресурси означає, що традиційні методи, які покладаються на бази даних відомих фішингових вебсайтів або електронних листів, виявляються малоефективними.

Фішингові атаки нульового дня становлять особливу загрозу через їхню здатність обходити існуючі системи безпеки. Вони націлені як на індивідуальних користувачів, так і на великі корпорації, використовують персоналізовані підходи та можуть призвести до катастрофічних наслідків.

Один із прикладів небезпеки таких атак полягає у тому, що зловмисники можуть використовувати атаки нульового дня для крадіжки облікових даних, які потім використовуються для отримання доступу до корпоративних мереж або систем. У випадку корпоративних мереж, це може призвести до втрати конфіденційної інформації, фінансових втрат або навіть повного порушення

операцій компанії.

1.3 Засоби виявлення фішингових атак

Сучасні засоби для виявлення та запобігання фішинговим атакам спрямовані на захист користувачів інтернету, застосовуючи різні методи (рис. 1.5).

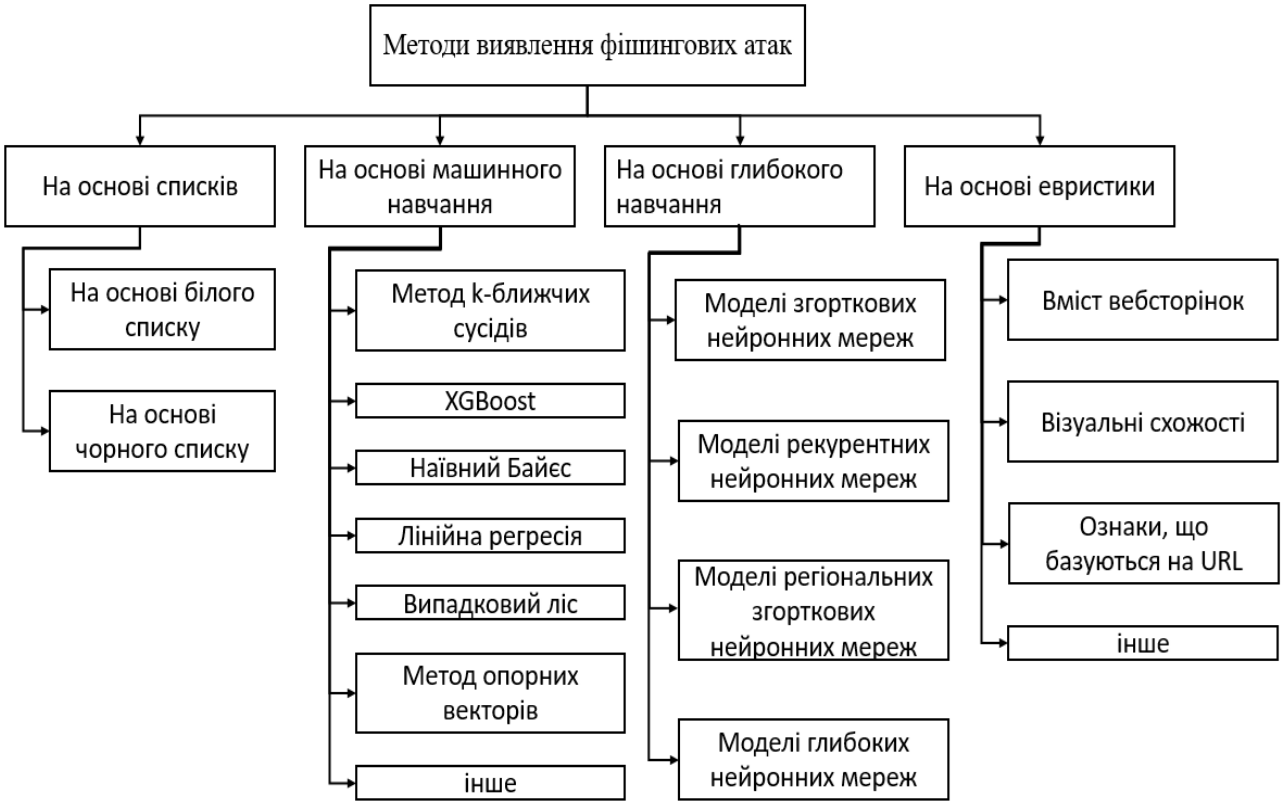

Рисунок 1.5 – Методи виявлення фішингових атак

Попри наявність передових рішень, проблема фішингу залишається через недостатню точність таких інструментів, що призводить до великих фінансових втрат під час онлайн-транзакцій щороку. Технічні підходи до виявлення та запобігання фішингу можна поділити на різні категорії залежно від використаних методів [20-22].

Панелі інструментів для виявлення фішингових листів є важливими

засобами, які допомагають користувачам захистити себе від шахрайських електронних листів та вебсайтів. Вони інтегруються у веббраузери і працюють на стороні клієнта, аналізуючи вміст відвідуваних вебсайтів або електронних листів на наявність підозрілих елементів, характерних для фішингових атак. Ці панелі інструментів стали одними з перших рішень для боротьби з фішингом і продовжують еволюціонувати разом із розвитком загроз в інтернеті. Було створено безліч панелей для захисту користувачів, серед яких Calling ID Toolbar, EarthLink Toolbar, Cloudmark AntiFraud Toolbar, eBay Toolbar та Netcraft Anti-Phishing Toolbar [23-25].

Хоча панелі інструментів для виявлення фішингових листів і вебсайтів є корисними для захисту від шахрайства, вони мають ряд недоліків [26-28]:

– панелі інструментів часто залежать від баз даних відомих фішингових вебсайтів та електронних листів. Якщо фішингова атака нова і ще не внесена в базу, інструмент може її пропустити;

– для ефективної роботи ці панелі повинні регулярно оновлювати свої сигнатури і бази даних фішингових сайтів. Якщо оновлення відбувається з затримкою або користувач не підтримує актуальність інструмента, це знижує рівень захисту;

– деякі панелі можуть позначати безпечні вебсайти або електронні листи як потенційно небезпечні через помилкову інтерпретацію їхніх елементів. Це може спричинити зниження довіри до панелі і втомити користувача постійними попередженнями;

– панелі інструментів, як правило, можуть захищати лише в межах браузера. Вони не забезпечують захист під час взаємодії з поштовими клієнтами або додатками поза веббраузером, тому користувач залишається вразливим до фішингових атак через інші канали;

– оскільки панелі інструментів постійно моніторять трафік і аналізують сайти, це може призвести до сповільнення роботи браузера або зменшення продуктивності системи;

– фішингові сайти стають дедалі складнішими, зловмисники використовують методи для маскування URL-адрес або створення тимчасових сторінок, які можуть обійти стандартні інструменти захисту;

– якщо користувач не правильно налаштовує або вимикає певні функції панелі інструментів, це значно знижує рівень її захисту. Інколи самі користувачі відмовляються від таких інструментів через їхню нав'язливість або сприймають їх як зайві.

Ці недоліки підкреслюють, що панелі інструментів для виявлення фішингових атак не є універсальним рішенням, і вони повинні використовуватися в поєднанні з іншими методами кібербезпеки.

Білі списки – це перелік довірених вебсайтів, які користувач Інтернету відвідує регулярно [29-30]. Ця методика дозволяє користувачу переходити лише на вебсайти, які були попередньо визнані легітимними. Вона ефективна в боротьбі з фішинговими атаками нульового дня і має нульовий відсоток помилкових спрацьовувань. Основний недолік використання білих списків полягає в тому, що складно врахувати всі вебсайти, на які користувачі можуть заходити в майбутньому. Наприклад, якщо користувач спробує відкрити легітимний сайт, який не занесено до білого списку, система може помилково вважати його фішинговим, що підвищує ризик помилкових негативних результатів. Тому білі списки не мають широкого використання. Отож, метод білого списку може бути корисним при використанні разом з іншими методами, такими як чорні списки або евристичні підходи. Він може прискорити інші методи виявлення фішингу, оскільки відомі легітимні сайти не потребують перевірки.

Метод чорних списків порівнює запитувані URL-адреси з попередньо визначеним чорним списком фішингових сайтів. Якщо користувач спробує відвідати шахрайський сайт, занесений до чорного списку, браузер блокує доступ або попереджає про небезпеку [31-32]. Основні переваги чорних списків – низький відсоток хибно позитивних результатів та простота реалізації. Дослідження показало, що чорні списки неефективні у запобіганні фішинг-

атакам нульового дня, 80% з яких не були виявлені. Основний недолік чорних списків полягає в тому, що вони не можуть охопити всі фішингові сайти, оскільки тривалість їхнього існування дуже коротка.

Антивірусне програмне забезпечення грає важливу роль у боротьбі з фішинговими атаками, і багато організацій інвестують значні кошти в антивірусні рішення для виявлення і запобігання фішингу [33]. Проте ці інвестиції не зупиняють зростання збитків від фішингу у світі. Брандмауери забезпечують додатковий рівень захисту, контролюючи мережевий трафік [34-35]. Вони можуть захистити користувачів від фішингових атак, які використовують трояни і кейлогери для крадіжки чутливих даних. Основний недолік антивірусного програмного забезпечення та брандмауерів полягає в тому, що фішинг-загрози спеціально створені для обходу таких систем захисту. Відповідно, успіх фішингових атак свідчить про недостатню ефективність цих методів у виявленні таких атак.

Евристичні методи перевіряють один або кілька ознак, отриманих з комунікаційного каналу, через який здійснюється атака, таких як електронна пошта, вебсайт, SMS або миттєві повідомлення. Методи на основі евристики використовують штучний інтелект для створення класифікаційної моделі на основі тренувального набору даних [36-37]. Потім ця модель використовується для класифікації отриманих даних як легітимних або фішингових. Вона використовує виявлені ознаки, такі як домен відправника, URL-адреси в електронних листах або код JavaScript. Евристичні методи можуть бути реалізовані на сервері або клієнтських машинах, а також як частина інших застосунків, таких як панелі інструментів браузера, брандмауери або антивірусне програмне забезпечення. Евристичні підходи мають переваги у формі відносно низького рівня помилкових позитивних результатів, високої точності, але мають недоліки у вигляді високих витрат на обробку та необхідності частої адаптації до нових фішингових атак.

1.4 Аналіз попередніх досліджень

У статті [38] представлено метод виявлення підроблених сайтів електронної комерції, використовуючи структури дерев об’єктної моделі документа (DOM). Автори обґрунтовують переваги застосування ядра підшляху для класифікації DOM-дерев, що дозволяє ефективно вирішувати проблему навіть для великих наборів даних. Метод показав високу точність на рівні до 0,998 при навчанні на прикладах фальшивих сайтів. Система виявлення підроблених сайтів була розроблена у співпраці з правоохоронними органами Японії та компанією Rakuten. Окрім технічних переваг, метод стійкий до спроб зловмисників змінити структуру сайтів для обходу виявлення. Таким чином, дослідження пропонує надійний спосіб боротьби з шахрайством в електронній комерції.

Робота [39] присвячена розробці методу для виявлення фішингових URL-адрес, зокрема тих, що виникають у рамках атак нульового дня. Автори пропонують підхід на основі згорткового автокодувальника, який працює з URL-адресами на рівні символів. Це дозволяє ефективніше виявляти аномалії в URL-адресах, які можуть свідчити про фішинг. У статті детально описуються кроки обробки та кодування URL-адрес, побудова моделі, а також застосування різних технік, таких як згорткові нейронні мережі та максимальне об'єднання. Результати досліджень на реальних наборах даних демонструють ефективність методу, що перевищує інші сучасні методи глибокого навчання.

Автори [40] пропонують підхід до виявлення фішингових веб-сайтів на основі глибоких згорткових нейронних мереж та ансамблевого навчання випадкових лісів. Запропонована методика дозволяє автоматично аналізувати URL-адреси без доступу до їхнього вмісту. Алгоритм використовує вбудовування символів для перетворення URL-адрес у фіксовані матриці, які обробляються CNN для витягу багаторівневих ознак. Потім ці ознаки класифікуються за допомогою кількох класифікаторів RF, після чого вибирається найточніший результат за методом голосування. Використання

ансамблевого навчання з декількома класифікаторами підвищує точність та здатність алгоритму до узагальнення, що робить його більш ефективним для виявлення фішингових сайтів у різних сценаріях.

Автори [41] пропонують метод виявлення фішингових сайтів на основі інтерполяції нечітких правил. Зокрема акцентують на уникненні обмежень традиційних методів, що використовують бінарні рішення або бази знань. Запропонована методика покращує стійкість системи та знижує складність процесу. Це досягається за рахунок адаптивної обробки даних, що дозволяє системі виявляти фішингові атаки навіть за неповної інформації або браку експертних знань. Запропонована система дозволяє коректно обробляти випадки, коли певні правила або параметри не повністю охоплені існуючою базою знань. Метод FRI-Incircle має потенціал для підвищення ефективності систем виявлення фішингових атак, пропонуючи більш зрозумілі та адаптивні рішення для адміністраторів. Це дозволяє швидше реагувати на кіберзагрози та забезпечувати більшу безпеку в мережі.

Стаття [42] присвячена використанню генетичних алгоритмів для покращення ефективності виявлення фішингових URL-адрес за допомогою моделей машинного навчання. Основна ідея дослідження полягає в тому, щоб оптимізувати вибір функцій, які використовуються в процесі навчання моделей, з метою покращення їхньої продуктивності. У статті пропонується застосування генетичних алгоритмів, які імітують природний відбір для пошуку найкращих підмножин функцій. Експериментальні результати демонструють, що застосування ГА суттєво підвищує ефективність системи виявлення фішингових URL-адрес, одночасно зменшуючи кількість обраних функцій, що прискорює процес навчання та прогнозування.

У роботі [43] автори пропонують підхід до виявлення фішингових веб-сторінок, що використовує поєднання графових згорткових мереж і трансформерних моделей. Суть полягає в тому, щоб об'єднати аналіз графів HTML DOM та ознак URL для більш точного визначення фішингових сайтів. Використання GCN дозволяє моделі враховувати структурні особливості HTML-

документів, а трансформери ефективно працюють із послідовностями символів та слів в URL-адресах.

Метод виявлення фішингових атак, що описаний у [44], поєднує метод збільшення даних та гібридну графову нейронну мережу. Метод використовує балансування вибірки даних і витягує тимчасові та структурні ознаки транзакцій. Для підвищення точності класифікації використовуються методи Conv1D та GRU-MHA для обробки часових рядів, а також реконструкція графа за допомогою SAGEConv.

Автори [45-49] демонструють різні підходи до виявлення фішингових веб-сторінок із використанням різних методів машинного навчання.

1.5 Постановка завдання

У сучасному кіберпросторі фішингові атаки є однією з найбільш поширених загроз, спрямованих на порушення безпеки користувачів і організацій. Метою даної роботи є розробка ефективного методу виявлення фішингових електронних листів, який здатний адаптуватися до нових типів загроз та підвищувати точність класифікації електронних повідомлень. Для досягнення цієї мети визначено наступні задачі: проведення огляду існуючих підходів і алгоритмів, які використовуються для класифікації електронних листів та виявлення фішингових повідомлень; створення методу, який здатен оновлювати модель класифікації у режимі онлайн, зберігаючи при цьому високу точність і враховуючи нові загрози; відбір характеристик електронних листів, які сприяють ідентифікації фішингових атак, що допоможе зменшити обчислювальне навантаження та підвищити ефективність алгоритму; оцінка роботи запропонованого методу на реальних та симуляційних даних, порівняння отриманих результатів із результатами відомих алгоритмів для підтвердження підвищеної точності та надійності.

2 ПОПЕРЕДНЯ ОБРОБКА ДАНИХ ТА ВИБІР ПАРАМЕТРІВ

2.1 Основні компоненти електронних листів

Електронна пошта є одним із важливих засобів комунікації в цифровому світі, що активно використовується у різних сферах — від особистих повідомлень до ділових взаємодій. Розуміння її структури дозволяє не тільки ефективніше використовувати цей інструмент, а й захищатися від кіберзагроз, таких як фішинг, який став однією з найпоширеніших форм шахрайства в інтернеті. Заголовок електронного листа відіграє важливу роль у процесі передачі повідомлення. Він містить інформацію про відправника, отримувача, дату відправлення та тему листа. У фішингових атаках зловмисники можуть підробляти ці поля, щоб виглядати як законний відправник. Наприклад, вони можуть використовувати підроблені адреси або спробувати змінити поле "From" для імітації офіційних організацій. Виявлення таких підробок — ключовий елемент у боротьбі з фішингом. Тіло листа містить основний текст повідомлення і може бути представлено у вигляді простого тексту або HTML-формату. Фішингові листи часто використовують HTML для створення візуально правдоподібних копій веб-сайтів, включаючи логотипи, стилі і навіть підроблені форми для збору особистих даних. Простий текст у фішингових повідомленнях, навпаки, може використовуватися для уникнення фільтрів спаму, адже він не викликає підозр у систем перевірки. Вкладення в електронних листах є ще одним інструментом, який часто використовують кіберзлочинці. Файли, прикріплені до фішингових листів, можуть містити шкідливе програмне забезпечення, таке як трояни або віруси, які активуються під час відкриття. Саме тому важливо бути уважними до будь-яких файлів, отриманих від незнайомих або підозрілих джерел.

Метадані, хоча і не видимі безпосередньо для користувачів, також можуть стати важливим аспектом у виявленні фішингових атак. Наприклад, аналіз IP-адреси відправника допомагає виявити, що лист, який нібито прийшов від банку, був надісланий з іншого континенту. Такі невідповідності можуть сигналізувати

про спробу обману. Крім того, спеціальні механізми аутентифікації, такі як DKIM, SPF і DMARC, дозволяють перевірити, чи дійсно лист був відправлений з серверів, що належать легітимним організаціям.

SMTP містить технічну інформацію про маршрут доставки листа, також може використовуватися для виявлення фішингових атак [50]. Зловмисники часто намагаються підробити шлях передачі повідомлення або використати обхідні методи, щоб приховати справжнє походження листа. Тому аналіз даних SMTP може допомогти в ідентифікації підозрілих активностей.

Підпис у кінці електронного листа зазвичай містить контактні дані відправника та іншу важливу інформацію. У фішингових повідомленнях ці підписи можуть бути підробленими або копіюватися з офіційних джерел, щоб додати правдоподібності шахрайському листу. Такі підробки допомагають зловмисникам обманути користувачів, переконуючи їх у легітимності листа.

Кожен із цих елементів — заголовок, тіло листа, вкладення, метадані, SMTP і підпис — може бути використаний у фішингових атаках для обману користувачів або обходу систем захисту. Уміння аналізувати структуру електронних листів є ключовим для виявлення таких загроз і запобігання потенційним збиткам від фішингових атак.

2.2 Попередня обробка даних з листа

Етап попередньої обробки даних складається з двох кроків. Перший крок передбачає вибір ознак, які будуть витягнуті з тексту і заголовка кожного електронного листа; ці ознаки описують різні властивості кожного повідомлення. Другий крок полягає у виборі найбільш ефективних ознак із набору, сформованого на першому етапі. Це дозволяє зменшити кількість ознак, що використовуються в запропонованій моделі, що прискорює її побудову та адаптацію класифікаційної системи. Характеристики обираються з трьох джерел: заголовки електронних листів, зміст листів та зовнішні джерела. Загалом

було обрано 48 ознак, однак їхній вплив на класифікацію листів як фішингових або звичайних не є постійним. Під час розробки системи статус окремої ознаки може динамічно змінюватися, щоб відображати природу фішингових атак типу нульового дня. Отримано чотири групи ознак з різних частин електронного листа: заголовків, URL, HTML та тексту. Для виконання етапу попередньої обробки електронний лист ділиться на заголовок і зміст. Із змісту виділяються URL, HTML та текст, залежно від типу контенту електронного листа. Такий поділ листа допомагає зменшити тривалість попередньої обробки. З електронного заголовка було отримано набір ознак, що дозволяють детальніше аналізувати властивості листів. У таблиці 2.1 подано список ознак.

Таблиця 2.1 - Функції, отримані із заголовка електронної пошти

№ парметру	Назва параметру	Опис
1	2	3
1	CompareMsgSenderDomain	порівнює домен ідентифікатора повідомлення та домен відправника, є двійковою функцією, яка визначає, чи доменні імена, отримані від відправника електронної пошти, збігаються з іменами, взятими з ідентифікатора повідомлення
2	HTMLmail	двійкова функція, яка визначає тип вмісту електронної пошти TEXT/HTML
3	Textmail	перевіряє, чи тип вмісту електронної пошти є текстовим/звичайним у заголовку електронної пошти
4	MultiPartMail	двійкова функція, яка перевіряє, чи тип вмісту електронної пошти є багатокомпонентним

Продовження таблиці 2.1

1	2	3
5	NumberOfReceivers	підраховує, скільки отримувачів включено в атрибут (from:) у заголовку електронного листа
6	NumberOfAttachments	підраховує кількість вкладень
7	SubjectFwdWord	перевіряє, чи тема електронної пошти містить слово (Fwd:), і повертає (true), якщо воно існує, і (false) в іншому випадку
8	SubjectReplyWord	перевіряє, чи містить тема електронного листа слово «Re:», і повертає (true), якщо воно існує, і (false) в іншому випадку
9	SubjectVerifyWord	перевіряє, чи містить тема електронного листа слово (Verify), і повертає (true), якщо воно існує, і (false) в іншому випадку
10	SubjectNumChars	підраховує кількість символів, які містить поле теми, і повертає це число
11	SubjectNumWords	підраховує кількість слів, які містить поле теми, і повертає це число
12	SubjectRichness	обчислює відношення кількості слів до кількості символів у полі теми
13	SendNumWords	підраховує кількість слів, які містить поле відправника, і повертає це число
14	SendDiffReplayto	перевіряє, чи не збігається відправник електронної пошти з полем «Replay», і повертає значення (true), якщо вони збігаються, і (false) в іншому випадку

Кінець таблиці 2.1

1	2	3
15	NumberOfRecipients	підраховує, скільки одержувачів включено в атрибут (To:) у заголовку електронної пошти.
16	NumberOfCcRecipients	підраховує, скільки отримувачів включено в атрибут (Cc:) у заголовку електронної пошти.
17	NumberOfBccRecipients	підраховує, скільки одержувачів включено в атрибут (Bcc:) у заголовку електронної пошти.

Однією з ознак є порівняння доменів відправника та ідентифікатора повідомлення. Ця ознака має двійкове значення і визначає, чи збігаються домени, взяті з адреси відправника та з поля Message-ID. Інша характеристика визначає, чи тип вмісту листа є HTML, а ще одна — чи це звичайний текстовий формат. Щодо кількісних характеристик, то було визначено, скільки одержувачів вказано у заголовку, а також підраховано кількість вкладень у листі. Деякі ознаки стосуються аналізу теми листа. Наприклад, одна ознака перевіряє наявність слова "банк" у темі листа, а інші — шукають такі слова, як "дебет", "Fwd:", "Re:", або "verify". Інша ознака підраховує кількість символів та слів у темі листа, а також співвідношення між кількістю слів і символів. Крім того, було додано ознаку, яка перевіряє, чи відправник листа збігається з полем "Reply-To". Аналізу також підлягає кількість одержувачів, зазначених у полях "To", "Cc" та "Bcc", що допомагає оцінювати потенційні спроби масової розсилки або прихованого копіювання.

У таблиці 2.2 представлено перелік характеристик, пов'язаних з гіперпосиланнями, які містяться в тілі електронного листа. Після отримання всіх URL-адрес із контенту листа до них застосовуються різні ознаки для подальшого аналізу.

Таблиця 2.2 - Функції, отримані із URL-адреси

№ парметру	Назва параметру	Опис
18	NumOfLink	обчислює кількість гіперпосилань, які з’являються в тілі електронного листа
19	NumberOfDi_Domain	підраховує кількість різних доменів, які використовуються у вмісті електронної пошти як гіперпосилання
20	NumDi_LinkText	обчислює кількість гіперпосилань, які містять текст гіперпосилання, який не містить ім’я домену цільового гіперпосилання
21	NumDomainNLSender	обчислює скільки гіперпосилань використовує домен, який не дорівнює домену відправника
22	NumOfDotInDomain	обчислює кількість точок у кожному гіперпосиланні та повертає максимальну кількість
23	NumberLinkContain@	підраховує кількість посилань у тілі електронного листа, які містять символ @
24	NumberOfLink ContainIP	підраховує кількість URL-адрес в електронному листі, які містять IP-адресу
25	NumberOfLink ContainEsc	підраховує кількість URL-адрес у тілі електронної пошти, які містять шістнадцяткові числа або екрановані символи URL-адреси
26	NumberOfLink ContainNSPort	підраховує кількість URL-адрес у тілі електронної пошти, які містять нестандартний порт
27	urlBagLink	двійкова функція, яка повертає (true), якщо будь-яке з наступних слів знайдено в URL-адресі: клацніть, тут, увійдіть і оновіть
28	UrlNumPort	підраховує кількість URL-адрес, які містять порт у розділі повноважень цієї URL-адреси, і повертає це число
29	BlackListURL	двійкова функція, яка повертає (true), якщо будь-яка URL-адреса, яка існує в тілі електронної пошти, існує в чорному списку URL-адрес. Ці URL-адреси з чорного списку збираються з PhishTank, безкоштовного сайту спільноти, де будь-хто може надсилати, перевіряти, відстежувати та ділитися фішинговими даними. Ця функція оновлюється кожні 60 хвилин, щоб забезпечити наявність останніх зареєстрованих фішингових веб-сайтів.

Однією з характеристик є підрахунок кількості гіперпосилань у листі. Це важливий показник, оскільки велика кількість посилань може свідчити про

спроби фішингових атак. Дослідження також включає визначення кількості різних доменів, які використовуються як гіперпосилання в електронному листі. Зазвичай фішингові листи можуть містити посилання на домени, що не відповідають легітимним відправникам. Ще однією важливою характеристикою є кількість посилань, текст яких не збігається з доменом цільового посилання. Це може бути ознакою прихованого шахрайства, коли зловмисники маскують URL-адресу, на яку веде посилання. Важливим аспектом є також визначення доменів, які не збігаються з доменом відправника листа. Така розбіжність може свідчити про спробу підробки або обману.

Кількість крапок у домені URL також може бути маркером небезпеки, адже довгі і складні доменні імена часто використовуються у фішингових атаках для маскування шкідливих посилань. Окрім того, існує характеристика, яка підраховує кількість посилань, що містять символ "@", а також URL-адреси з IP-адресами замість доменних імен, що є типовою практикою серед фішингових сайтів.

Також розглядається наявність у URL шістнадцяткових чисел або символів, що кодуються, що може вказувати на спроби зловмисників приховати справжнє призначення посилання. Аналізу підлягає і кількість посилань, які використовують нестандартні порти, що відрізняються від загальноприйнятих портів 80 або 443.

Окрім того, фішингові листи часто використовують маніпулятивні слова у посиланнях, такі як "натисни", "тут", "зареєструватися", або "оновити". Тому наявність цих слів у URL-адресах також враховується під час аналізу. Ще одним важливим показником є перевірка URL-адрес на наявність у чорних списках, наприклад, у PhishTank — спільноті, яка займається збором і верифікацією фішингових сайтів. Ця характеристика дозволяє оперативно виявляти URL-адреси, що належать до фішингових ресурсів, і постійно оновлюється для забезпечення актуальності даних.

Якщо вміст електронного листа представлено у форматі HTML, алгоритм попередньої обробки виділяє декілька ознак (табл. 2.3), які допомагають виявити

можливі загрози та підозрілу активність у контенті листа. Серед таких ознак є перевірка наявності HTML-форми в тексті листа. Це може бути індикатором фішингової атаки, оскільки HTML-форми часто використовуються для збору конфіденційної інформації, такої як логіни та паролі користувачів. Додатково перевіряється, чи включає електронний лист JavaScript-скрипти, зокрема, наявність спливаючих вікон. Такі скрипти можуть використовуватися зловмисниками для створення нав’язливих або оманливих повідомлень, які спонукають користувача виконати небезпечні дії. Крім цього, оцінюється, чи є у тілі листа посилання на сайти, які використовують шифрування за допомогою самопідписаних сертифікатів. Самопідписані сертифікати часто застосовуються на ненадійних або потенційно шкідливих вебсайтах. Алгоритм також підраховує кількість зображень, що використовуються як гіперпосилання. Це важливо, адже деякі фішингові листи можуть містити багато зображень, які виглядають як кнопки або посилання і ведуть на шкідливі вебсайти. Аналізується також наявність зображень із картами, які використовуються як гіперпосилання, що також може свідчити про потенційно небезпечний контент. Крім того, у процесі попередньої обробки підраховується кількість URL-адрес з нестандартними ASCII-символами. Такий підхід дозволяє виявити адреси, які можуть бути замаскованими або незвичайними, що нерідко використовується для обману користувачів. Ще одним етапом перевірки є порівняння доменного імені з відповідним зворотним DNS-записом. Якщо доменне ім'я та зворотний запис не збігаються, це може бути ознакою підозрілого походження електронного листа. Алгоритм також враховує деякі бінарні ознаки, що сигналізують про наявність специфічних JavaScript-елементів. Серед таких ознак — подія onClick, що може активувати дії після натискання на елемент, наявність спливаючих вікон JavaScript, а також зміни тексту у рядку статусу, ініційовані через JavaScript. Наявність цих ознак може свідчити про спроби приховати реальне призначення або походження листа, що часто використовується у фішингових атаках.

Таблиця 2.3 Функції, отримані з HTML-вмісту електронної пошти

№ парметру	Назва параметру	опис
30	HTMLform	Перевіряє, чи вміст електронної пошти містить елемент форми HTML
31	ContainScript	Перевіряє, чи електронний лист містить спливаюче вікно JavaScript
32	CountSSLLink	Підраховує кількість URL-адрес у тілі електронної пошти, які вказують на веб-сайт, який шифрує з’єднання за допомогою самопідписаного сертифіката.
33	NumOfLinksUsingImage	Обчислює кількість зображень, які використовуються як гіперпосилання
34	NumMapLink	Обчислює кількість зображень із картами зображень, які використовуються як гіперпосилання
35	NumLinkNonASCII	Підраховує кількість URL-адрес, які містять нестандартні символи ASCII
36	NumOfDNSrDNS	Перевіряє, чи доменні імена мають відповідний зворотний запис DNS, і повертає (true), якщо вони рівні, інакше повертає (false)
37	ScriptOnclick	Бінарна функція, яка перевіряє, чи електронний лист містить подію JavaScript onClick, і повертає (true), якщо вона доступна, і повертає (false), якщо ні
38	ScriptPopup	Двійкова функція, яка перевіряє, чи електронний лист містить у своєму вмісті спливаючі вікна JavaScript, і повертає значення true, якщо воно доступне, і false в іншому випадку
39	ScriptStatusChange	Двійкова функція, яка перевіряє, чи електронний лист містить код JavaScript, який змінює текст, який відображається в рядку стану, і повертає (true), якщо він доступний, інакше повертає (false)

Для обробки основного тексту електронного листа та виявлення потенційних загроз необхідно спочатку виділити текст, який відображається отримувачу, а потім застосувати набір параметрів (табл. 2.4).

Таблиця 2.4 – Функції, що отримані з основного тексту електронного листа

№ парметру	Назва параметру	Опис
1	2	3
40	SizeOfDocument	Повертає розмір повідомлення електронної пошти в байтах
41	BodyDearWord	Двійкова функція, яка перевіряє, чи текст електронної пошти містить слово \Dear" і повертає (true), якщо він існує, або в іншому випадку повертає (false)
42	BodyNumChars	Підраховує кількість символів у тілі електронної пошти
43	BodyNumWords	Підраховує кількість слів у тілі електронного листа
44	BodyNumUniqueWords	Підраховує кількість унікальних слів у тілі електронного листа
45	BodyRichness	Обчислює кількість слів, поділену на кількість символів у тілі електронного листа
46	BodyNumFunctionWords	Підраховує кількість службових слів, виявлених у тілі електронного листа. Функціональні слова: обліковий запис, доступ, банк, кредит, клацання, ідентифікація, незручність, інформація, обмежено, журнал, хвилини, пароль, нещодавно, ризик, соціальний, безпека, обслуговування та призупинено

Кінець таблиці 2.4

1	2	3
47	BodySuspensionWord	Перевіряє, чи текст електронної пошти містить слово \suspension" і повертає (true), якщо воно існує, і (false) в іншому випадку
48	BodyVerifyYourAccountPhrase	Двійкова функція, яка перевіряє, чи тіло електронного листа містить речення «підтвердити свій обліковий запис" і повернути (true), якщо він існує, і (false) інакше

Серед цих параметрів — розмір документа в байтах, що дозволяє оцінити обсяг повідомлення. Деякі параметри зосереджені на пошуку конкретних слів або фраз, характерних для фішингових повідомлень. Наприклад, параметр перевірки наявності слова «Шановний» у тексті листа є бінарною ознакою, яка сигналізує про потенційний формальний стиль, часто притаманний фішинговим листам. Інші ознаки включають підрахунок кількості символів і слів у тілі листа, що дозволяє зробити висновки про його обсяг і стиль написання. Зокрема, враховується кількість унікальних слів та багатство тексту, яке розраховується як співвідношення кількості слів до кількості символів. Така оцінка допомагає виявити характерні риси повідомлень, оскільки фішингові листи нерідко мають характерний стиль із високою або низькою щільністю слів. Окрема увага приділяється функціональним словам, які часто використовуються в шахрайських листах. Це такі слова, як «акаунт», «банківський рахунок», «пароль», «логін», «безпека» тощо. Підрахунок цих слів допомагає ідентифікувати фішингові листи, які часто містять подібні терміни, аби спонукати отримувача до виконання небезпечних дій. Крім того, здійснюється перевірка наявності певних підозрілих слів, таких як «suspension», або фрази «verify your account», що свідчать про спробу маніпуляції користувачем для

викрадення його особистих даних. Таким чином, застосування зазначених ознак до тексту електронного листа дозволяє виявити не лише технічні характеристики повідомлення, але й певні лексичні та стилістичні особливості, що підвищують ймовірність ідентифікації фішингового контенту.

Фаза попередньої обробки є важливою в розробленій моделі, оскільки саме на цьому етапі здійснюється отримання інформації з набору даних. Для автоматичного отримання інформації з електронних листів було використано готовий програмний код, яка забезпечує зчитування вмісту листів та виокремлення ознак. Процес попередньої обробки розділено на три етапи, на кожному з яких виконуються різні завдання.

Для реалізації завдань цього процесу електронний лист ділиться на дві основні частини: заголовок і тіло листа. Заголовок містить важливу інформацію, таку як відправник, отримувач, унікальний ідентифікатор повідомлення (message-ID) та тип вмісту. Тіло листа, у свою чергу, містить основну інформацію, призначену для отримувача. Тип вмісту електронного листа залежить від атрибута content-type, зазначеного в заголовку.

На першому етапі (рис. 2.1), з заголовка електронного листа виділяється низка ознак.

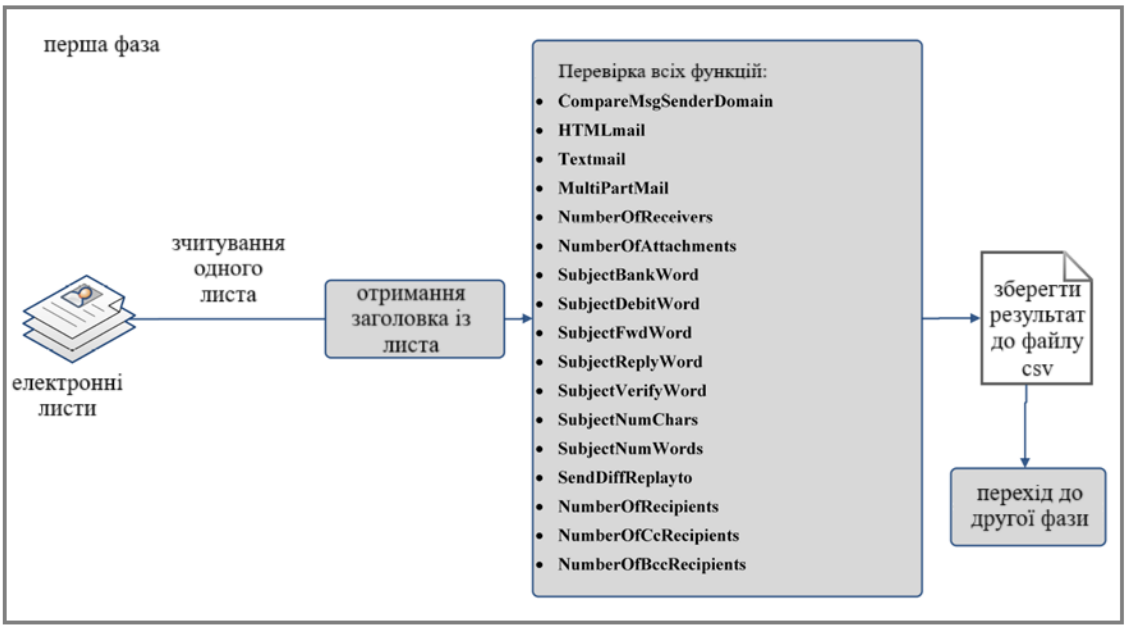

Рисунок 2.1 - Фаза 1 попередньої обробки

Результати для кожної ознаки зберігаються у файлі формату CSV, який згодом використовуватиметься як вхідні дані для алгоритму класифікації після завершення процесу отримання ознак. Після цього програма переходить до другого етапу, де перевіряється, чи вміст листа має тип text, що дозволяє визначити, які саме ознаки слід виділити.

На другому етапі, як показано на рис. 2.2, із вмісту електронного листа отримується набір ознак.

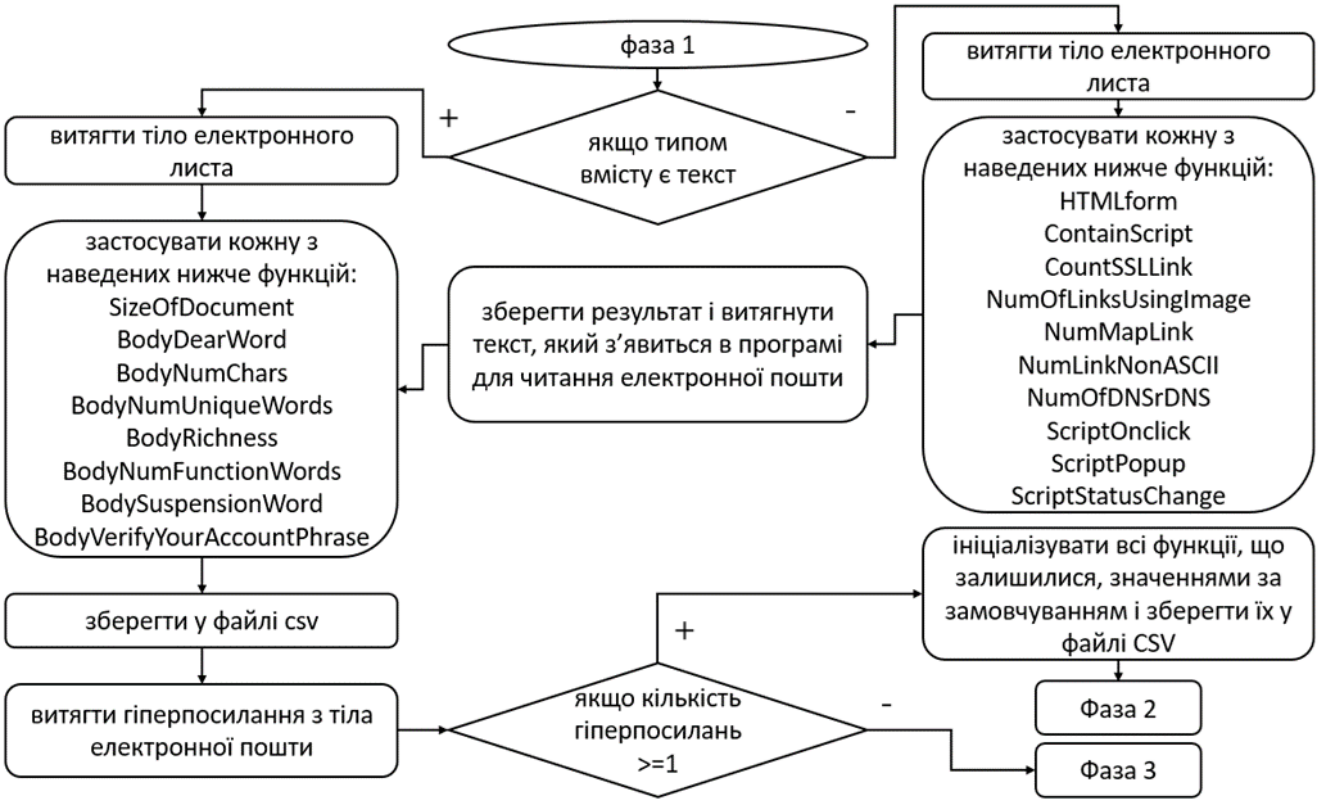

Рисунок 2.2 - Фаза 2 попередньої обробки

Цей набір залежить від типу вмісту електронного листа: якщо він відрізняється від text, то виділяються додаткові ознаки, пов'язані з HTML-вмістом, а також ознаки, що стосуються тексту листа. Наприкінці другого етапу отримуються всі гіперпосилання, які можуть міститися в тілі листа. Якщо гіперпосилань немає, решта ознак ініціалізується значеннями за замовчуванням, і третій етап пропускається. В іншому випадку програма переходить до третього етапу, де виділяються ознаки, пов'язані з гіперпосиланнями, що пришвидшує

процес отримання даних.

На всіх етапах виділені ознаки зберігаються в тому ж файлі CSV, що й результати першого етапу, і перед початком третього етапу витягуються всі гіперпосилання, розміщені в тілі листа. Наприклад, логіка обробки окремих ознак проілюстрована на рис. 2.3, де відображено як працює функція CompareMsgSenderDomain. Даний алгоритм зчитує заголовок листа, виділяє message-ID, відправника, та порівнює домен відправника з доменом із ідентифікатора повідомлення.

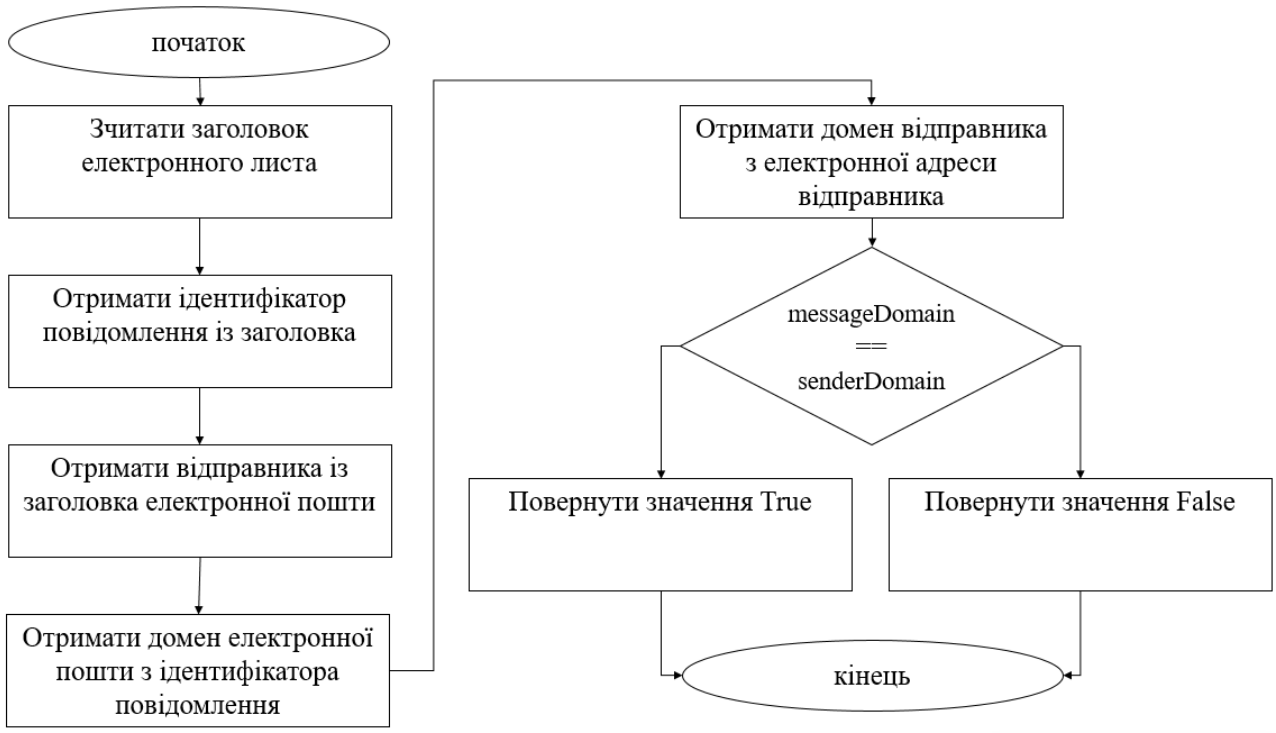

Рисунок 2.3 – Алгоритм роботи функції CompareMsgSenderDomain

На другому етапі проводиться виділення всіх гіперпосилань, наявних у тілі листа. Якщо гіперпосилання відсутні, решта ознак ініціалізується, і програма переходить до обробки наступного листа. Інакше процес одержання даних продовжується на третьому етапі (рис. 2.4). Цей етап передбачає виконання кількох кроків для вилучення характеристик із URL-адрес, отриманих на попередньому, другому етапі. Зібрані характеристики зберігаються у файлі CSV, що дозволяє подальший аналіз. Алгоритм починає з URL-посилань, вилучених із

тіла електронного листа на етапі 2. Це означає, що попередній етап вже виділив всі URL-адреси, які містяться в електронному листі. Далі застосовується набір функцій для кожної отриманої URL, щоб отримати набір характеристик, які можуть допомогти у виявленні фішингових або зловмисних посилань. Після того як всі характеристики для кожного URL обчислені, результати зберігаються у форматі CSV. Це дозволяє структуровано організувати дані для подальшого аналізу. Після збереження даних для одного електронного листа, алгоритм переходить до обробки наступного листа, повертаючись до першого етапу (Phase 1), щоб повторити процес.

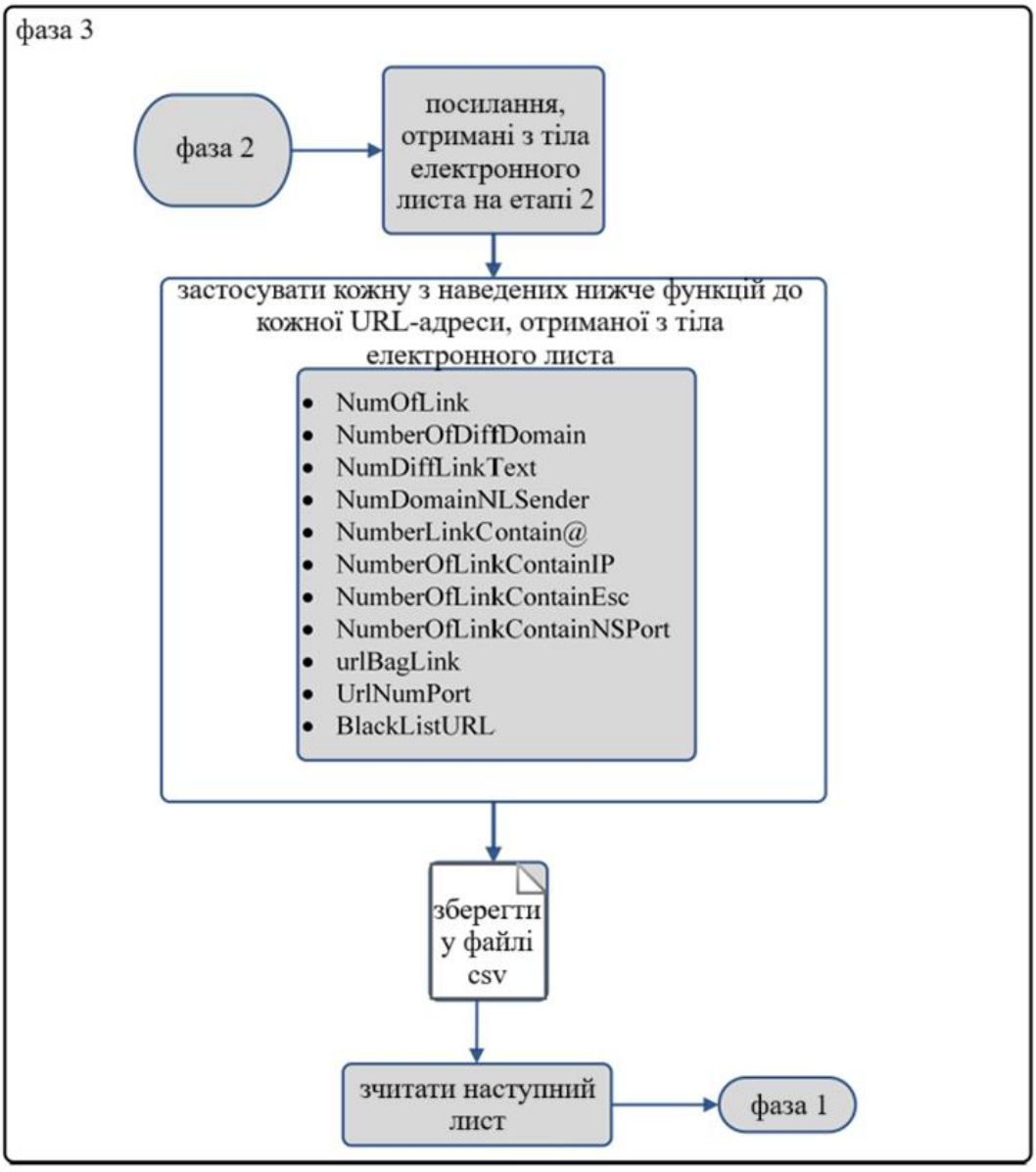

Рисунок 2.4 - Фаза 3 попередньої обробки

Алгоритм попередньої обробки, показаний на рисунках 2.1, 2.2 та 2.4, приймає на вхід список електронних листів із набору даних і створює файл у форматі CSV. Цей файл буде використано на подальших етапах для навчання та тестування запропонованої моделі.

2.3 Оптимізація кількості ознак

Після застосування фази попередньої обробки та одержання ознак, описаних у попередньому параграфі, отримані результати передаються алгоритму Feature Extraction and Ranking (FEaR). Цей алгоритм визначає, скільки саме ознак використовуватиметься для процесу класифікації електронних листів. Вибрані ознаки залежать від навчального набору даних, який допомагає визначити необхідну кількість ознак для класифікації повідомлень у цьому наборі. У разі зміни навчального набору даних зміниться і набір обраних ознак. Алгоритм попередньої обробки вибирає з усіх ознак найбільш значущі для побудови моделі виявлення загроз. Ознаки, які вважаються менш важливими, виключаються з подальшого аналізу, що прискорює процес розробки моделі без зниження точності, яка була б досягнута за використання всіх ознак. Вхідними даними є електронний лист, а результатом є CSV файл із отриманими ознаками. Загалом послідовність кроків роботи алгоритму наступна:

Крок 1. Прочитати один електронний лист за одну ітерацію.

Крок 2. Отримати заголовок листа.

Крок 3. Застосувати код для кожної з наступних характеристик до заголовка листа CompareMsgSenderDomain, HTMLmail, Textmail, MultiPartMail, NumberOfReceivers, NumberOfAttachments, SubjectFwdWord, SubjectReplyWord, SubjectVerifyWord, SubjectNumChars, SubjectNumWords, SubjectRichness, SendNumWords, SendDiffReplayto, NumberOfRecipients, NumberOfCcRecipients та NumberOfBccRecipients.

Крок 4. Зберегти результат у файлі CSV, де для кожного листа буде один

запис, а кожна характеристика займе окрему колонку.

Крок 5. Якщо тип вмісту дорівнює "text", перейти до кроку 6; в іншому випадку перейти до кроку 12.

Крок 6. Отримати тіло листа і застосувати кожну з характеристик SizeOfDocument, BodyDearWord, BodyNumChars, BodyNumWords, BodyNumUniqueWords, BodyRichness, BodyNumFunctionWords, BodySuspensionWord, BodyVerifyYourAccountPhrase. Ініціалізувати значення за замовчуванням для характеристик HTMLform, ContainScript, CountSSLLink, NumOfLinksUsingImage, NumMapLink, NumLinkNonASCII, NumOfDNSrDNS, ScriptOnclick, ScriptPopup, ScriptStatusChange і перейти до кроку 7.

Крок 7. Застосувати вказані характеристики до тіла листа HTMLform, ContainScript, CountSSLLink, NumOfLinksUsingImage, NumMapLink, NumLinkNonASCII, NumOfDNSrDNS, ScriptOnclick, ScriptPopup, ScriptStatusChange. Потім витягти текст, який бачить читач, застосувати характеристики SizeOfDocument, BodyDearWord, BodyNumChars, BodyNumWords, BodyNumUniqueWords, BodyRichness, BodyNumFunctionWords, BodySuspensionWord, BodyVerifyYourAccountPhrase і зберегти результати у вихідному файлі.

Крок 8. Зібрати всі гіперпосилання, що є в тілі листа, і зберегти їх у масив.

Крок 9. Якщо розмір масиву гіперпосилань дорівнює нулю, перейти до кроку 10; в іншому випадку перейти до кроку 11.

Крок 10. Зберегти значення за замовчуванням для всіх решти характеристик, оновити файл CSV і перейти до кроку 12.

Крок 11. Застосувати характеристики NumOfLink, NumberOfDi_Domain, NumDi_LinkText, NumDomainNLSender, NumOfDotInDomain, NumberLinkContain@, NumberOfLink ContainIP, NumberOfLink ContainEsc, NumberOfLink ContainNSPort, urlBagLink, UrlNumPort, BlackListURL до списку гіперпосилань і зберегти результат у вихідному файлі.

Крок 12. Якщо є ще один лист, повернутися до кроку 1; в іншому випадку перейти до кроку 13.

Крок 13. Згенерувати набір даних у форматі CSV.

На першому етапі FEaR використовує алгоритм CART (Classification and Regression Tree). Цей алгоритм оцінює навчальний набір даних, щоб сформувати список важливих ознак і виключити ті, що не є значущими. Алгоритм Cart починає роботу з вибору зі списку функцій тієї, яка має найкращий розподіл з точки зору класу електронної пошти для навчального набору даних. Дані розділяються на основі вибраної функції на першому кроці, той самий процес повторюється, доки навчальний набір даних не буде найкраще класифікований. Загалом алгоритм CART складається з наступних кроків.

Крок 1. Створення дерева класифікації та регресії за допомогою алгоритму CART. Короткий опис роботи алгоритму CART: почати з навчального набору даних; розглянути всі можливі значення всіх характеристик; вибрати характеристику xi із певним значенням t1, (xi = t1), що забезпечує найкращий поділ у класі електронної пошти; розділити дані на дві гілки: якщо (xi < t1), то дані переходять у ліву гілку; інакше — у праву; повторити цей процес для кожного з нових вузлів, доки не буде побудовано дерево. Це дерево, що містить важливі характеристики для поділу навчального набору, стане вхідними даними для наступного кроку.

Крок 2. Оцінка важливості кожної характеристики. Кожен вузол у дереві (побудованому на першому кроці) представляє одну характеристику. Важливість вузла (наприклад, вузла 1, який має дочірні вузли 2 та 3) оцінюється за формулою [51]:

$$V_1 = \frac{(R_1 - R_2 - R_3)}{Num\_node} \tag{2.1}$$

де R1, R2 та R3 — ризики для батьківського і дочірніх вузлів, а Num node — загальна кількість вузлів у дереві.

Ризик вузла визначається за формулою:

$$R_i = P_i * E_i \tag{2.2}$$

де Pi — ймовірність вузла, а Ei — похибка вузла, розрахована алгоритмом CART.

Крок 3. Розрахунок кінцевого значення для кожної характеристики шляхом ділення важливості кожної характеристики з кроку 2 на максимальне значення важливості:

$$\left\{V_i = \frac{V_i}{\max(v)} * 100 | \forall_i = 1,2,3 \dots n\right\} \tag{2.3}$$

де Vi — важливість характеристики i, n — загальна кількість характеристик, а max(v) — максимальне значення у векторі всіх характеристик.

Крок 4. Вибір важливих характеристик із значенням Vi > 0 як значущих. Характеристики з Vi = 0 не мають відповідних вузлів у дереві.

Крок 5. Скорочення кількості характеристик, отриманих на етапі попередньої обробки, з 49 до списку важливих характеристик, визначених на кроці 4.

2.4 Модель ідентифікації фішингових атак в електронних листах

На рис. 2.5 зображені загальні компоненти моделі, розробленої для класифікації електронних листів.

Для навчання моделі було використано десять різних алгоритмів класифікації. Серед них BayesNet, Multilayer Perceptron, Naive Bayes, Random Forest і Логістична регресія. Було проведено порівняння для визначення найкращого алгоритму, який можна використовувати для цього завдання. Вибрані алгоритми класифікації є одними з найпоширеніших у подібних моделях класифікації.

Модель також включає mbx2eml — безкоштовний інструмент, який дозволяє розділити листи, згруповані у форматі mbox, зберігаючи кожен лист у окремому файлі. Далі виконується попередня обробка даних, під час якої з електронних листів вилучаються 48 ознак, що використовуються для навчання класифікаційної моделі.

Для подальшого навчання та тестування було проведено експерименти з використанням методу 10-кратної крос-валідації на основі алгоритмів обробки даних. Цей метод дозволяє виміряти точність та надійність системи. Крос-валідація використовується, оскільки її оцінювачі забезпечують кращі результати в порівнянні з одним набором даних для валідації, що особливо корисно, коли обсяги доступних даних обмежені, як у випадку цієї роботи.

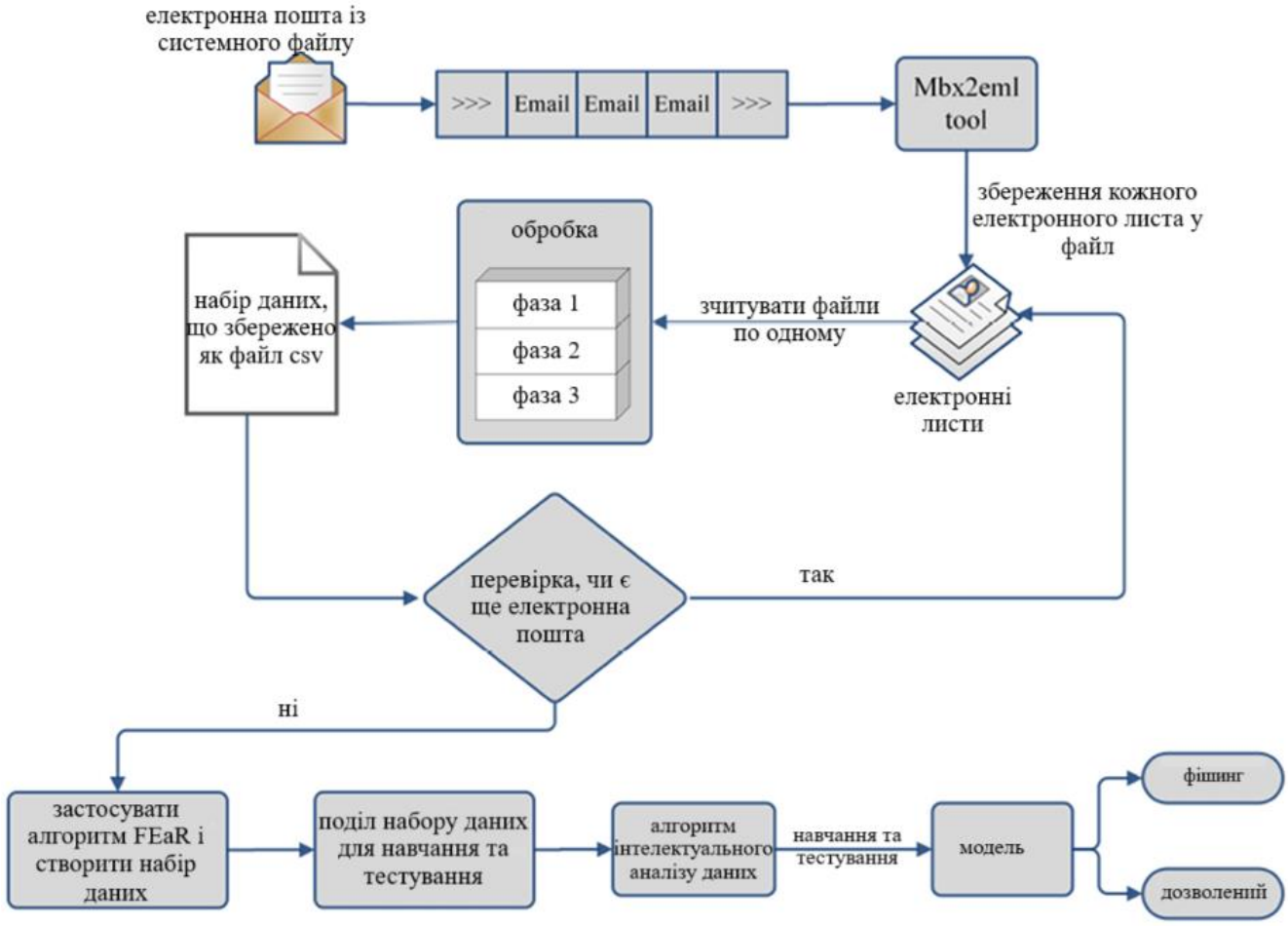

Рисунок 2.5 – Модель виявлення фішингових листів

Оцінка продуктивності може змінюватися при повторенні експерименту з різними групами даних для формування навчальних та тестових наборів. Метод 10-кратної крос-валідації зменшує ці зміни в продуктивності, беручи середнє з десяти різних поділів. Тому оцінка продуктивності менш чутлива до способу розподілу даних у наборі.

Для дослідження були використані три доступні для публічного використання набори даних: два набори, що містять електронні листи, та один набір URL-адрес фішингових сайтів. Перший набір даних, PhishingCorpus [52], був зібраний вручну і представляє собою набір фішингових електронних листів. Цей набір даних активно використовувався дослідниками. Другий набір, SpamAssassin [53], містить 6047 електронних листів, з яких 4951 є легітимними. У навчанні та тестуванні запропонованої моделі використовувалися саме ці легітимні електронні листи з колекції SpamAssassin. Третій набір даних, PhishTank, був зібраний організацією PhishTank, яка є спільним інформаційним центром для даних і інформації про фішинг в Інтернеті. Цей набір даних слугує для оновлення вмісту функції BlackListURL, а список заблокованих URL-адрес автоматично оновлюється кожні 60 хвилин на веб-сайті PhishTank.

Як показано в таблиці 2.5, для набору даних з 500 електронних листів алгоритм FEaR виявив 11 ознак, які можуть бути використані для розрізнення фішингових та легітимних електронних листів. Зміна навчального набору даних дозволила виявити новий фішинговий лист, що свідчить про нову поведінку, яку використовують зловмисники для обману онлайн-клієнтів. Алгоритм FEaR будує різні регресійні дерева, які найкраще класифікують електронні листи в навчальному наборі, при цьому на кожному експерименті змінюються вузли дерева (ознаки) із загального списку. Цей алгоритм продемонстрував здатність виявляти нові поведінки; при обробці наборів даних з різними списками електронних листів обирається різний набір ознак. Для піднабору розміром 1000 листів алгоритм FEaR виявив різний набір ознак. Це сталося через те, що були випадковим чином обрані різні набори електронних листів, в яких фішингові листи могли містити різні поведінки, використовувані зловмисниками для

обману онлайн-клієнтів.

Запропонований алгоритм вибору з великого набору ознак вирішує проблему підбору правильного набору ознак для побудови моделі класифікації. Крім того, кількість важливих ознак динамічно змінюється в залежності від зміни набору даних без жодного втручання з боку користувача. Це відрізняється від інших досліджень, де використовувалися фіксовані кількості ознак для класифікації фішингових електронних листів, а обрані набори ознак змінювалися вручну.

Таблиця 2.5 - Параметри, визначені як важливі за допомогою алгоритму FEaR

Розмір набору даних	Кількість параметрів	Номери параметрів
500	11	3, 8, 10, 12, 14, 18, 23, 40, 41, 45, 46
1000	17	1, 3, 7, 8, 10, 12, 13, 18, 19, 23, 33, 34, 37, 40, 42, 43, 45, 46
1000	18	3, 8, 10, 11, 12, 13, 14 18, 19, 23, 36, 40, 41, 42, 43, 44, 45, 46
1000	18	3, 7, 8, 10, 12, 13, 14, 18, 19, 20, 22, 23, 40, 41, 42, 43, 45, 46
1500	19	3, 5, 6, 8, 10, 11, 12, 13, 18, 21, 22, 23, 33, 40, 41, 43, 44, 45, 46
2000	23	1, 3, 5, 7, 8, 9, 10, 12, 13, 14, 18, 19, 22, 23, 27, 32, 33, 40, 41, 43, 44, 45, 46
2500	27	1, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16 18, 19, 20, 23, 32, 33, 34, 40, 41, 42, 43, 44, 45, 46
3000	27	3, 4, 5, 6, 8, 9, 10, 11, 12, 13, 14, 16, 18, 19, 21, 22, 23, 24, 27, 28, 40, 41, 42, 43, 44, 45, 46
3500	29	1, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19, 20, 22, 23, 27, 28, 32, 34, 37, 40, 41, 42, 43, 44, 45, 46
4000	28	1, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19, 20, 22, 23, 27, 32, 33, 40, 41, 42, 43, 44, 45, 46
9900	31	1, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19, 20, 22, 23, 27, 32, 33, 34, 37, 40, 41, 42, 43, 44, 45, 46, 48

Більш того, запропонований алгоритм може також ранжувати обрані ознаки, як показано в таблиці 2.6, де ранги Vi оцінюються на етапі 3 алгоритму FEaR. Деякі з запропонованих ознак були обрані алгоритмом FEaR. Як показано в таблиці 2.5, ознаки (3, 4, 5, 6 і 16) були визначені як важливі. Вибір цих ознак алгоритмом FEaR залежить від розміру набору даних. Наприклад, якщо розмір набору даних становить 500 електронних листів, то алгоритм FEaR обирає одну ознаку (3). Для набору з 3000 електронних листів були обрані ознаки (3, 4, 5 і 6) як важливі. Ранг нових запропонованих ознак свідчить про те, що ці ознаки є частиною найбільш важливих ознак у порівнянні з раніше запропонованими ознаками, як показано в таблиці 2.6. Наприклад, ознака 3 була обрана як найбільш важлива.

Таблиця 2.6 - Рейтинг параметрів, оцінений алгоритмом FEaR за допомогою 4000 електронних листів

Номер параметра	Ранг	Номер параметра	Ранг	Номер параметра	Ранг
3	100	40	0,91	1	0,34
19	13,97	44	0,88	3	0,3
41	6,5	37	0,83	11	0,29
8	5,18	10	0,70	48	0,28
43	3,77	13	0,68	7	0,21
46	2,27	42	0,66	9	0,14
45	2,03	33	0,55	22	0,07
18	1,81	23	0,5	27	0,05
12	1,28	5	0,39	20	0,04
14	0,93	6	0,35	21	0,04

Алгоритм FEaR оцінюється на наборах даних різних розмірів за допомогою повного списку ознак на вхід та вибору найбільш ефективного

списку ознак, які можуть класифікувати електронні листи в досліджуваному наборі даних. Якщо розмір набору даних змінюється, він міститиме різні групи фішингових і легітимних електронних листів. Алгоритм FEaR успішно обирає різні списки ознак для різних наборів даних.

2.5 Висновки до розділу

У другому розділі детально розглянуто кроки попередньої обробки даних електронних листів з метою їхньої ефективної класифікації як фішингові або легітимні. Висновки з цього розділу підкреслюють важливість кожного етапу, який сприяє вдосконаленню процесу класифікації та підвищенню ефективності методів виявлення загроз.

Попередня обробка є основою для успішної класифікації фішингових листів, оскільки дозволяє підготувати неструктуровані текстові дані для подальшого аналізу. Зокрема, було виявлено, що обробка таких елементів листа, як заголовок, вміст та гіперпосилання, дозволяє виокремити потенційні ознаки фішингу. Ці елементи не тільки містять інформацію, необхідну для визначення легітимності листа, але й часто використовуються зловмисниками для введення користувачів в оману.

На етапі попередньої обробки особливу увагу приділено виділенню ознак, таких як частота використання певних слів, структура посилань, наявність вкладень, специфічні ключові слова в темі листа. Вони були обрані через їхню високу кореляцію з фішинговими схемами, що дозволяє точно диференціювати фішингові листи від безпечних. За допомогою таких характеристик, як кількість одержувачів, наявність ключових слів, структура теми, було досягнуто кращого розуміння поведінки зловмисників у кіберпросторі.

Алгоритм FEaR дозволив оптимізувати кількість ознак, залишивши лише ті, які найбільше впливають на класифікацію. Такий підхід скорочує обсяг даних, які потрібно обробляти, що прискорює роботу моделі, одночасно забезпечуючи

високий рівень точності виявлення фішингових листів. Це сприяє ефективнішій роботі моделі без перевантаження зайвою інформацією, що є важливим аспектом для забезпечення надійної роботи у реальному часі.

Оптимізація на етапі попередньої обробки та вибір найбільш значущих ознак дозволяє зменшити навантаження на обчислювальні ресурси. Це досягається завдяки зосередженню лише на ключових характеристиках, що відповідають за ідентифікацію фішингових листів. Таке рішення дозволяє застосовувати модель у великомасштабних системах з високою пропускною здатністю, зберігаючи її продуктивність і точність.

Запропоновані підходи забезпечують можливість виявлення фішингових атак у реальному часі, що важливо для запобігання втратам даних. Система має змогу швидко обробляти вхідні листи, визначати потенційні загрози та реагувати на нові фішингові техніки зловмисників, навіть коли вони ще не занесені до баз даних загроз.

У підсумку, другий розділ дослідження підтверджує, що ретельна попередня обробка та виділення найважливіших характеристик є критичним етапом у процесі ідентифікації фішингових атак. Обрані параметри забезпечують більш високу точність, швидкість роботи та адаптивність системи виявлення фішингових загроз, що робить її важливим інструментом для забезпечення безпеки користувачів.

3 МЕТОД ІДЕНТИФІКАЦІЇ ФІШИНГОВИХ АТАК В ЕЛЕКТРОННИХ ЛИСТАХ

3.1 Метод ідентифікації фішингових атак в електронних листах

Сучасні методи досить ефективно справляються з ідентифікацією відомих фішингових листів, але існує затримка у додаванні нововиявлених фішингових сайтів до чорних списків. Користувач, який відвідує новий фішинговий сайт, залишається вразливим до моменту, поки цей сайт не буде внесено до чорних списків. Використання евристичних алгоритмів дозволить виявляти нові фішингові листи.

Виявлення нульових фішингових атак має важливе значення для онлайн-транзакцій, але лише обмежена кількість досліджень розробила методи для обробки таких атак. Будь-який метод, що має на меті виявлення нульових фішингових атак, повинен мати можливість динамічно адаптуватися для виявлення, щоб враховувати зміни, виявлені в нових фішингових листах. Крім того, він повинен мати можливість досліджувати нові поведінки в щойно отриманих електронних листах в режимі онлайн. У жодному з попередніх досліджень не надано чіткої ідеї щодо того, як вивчати ці нові поведінки у нульових фішингових електронних листах.

Метод виявлення, призначений для обробки нульових фішингових атак, повинен мати кілька значущих характеристик. По-перше, метод повинен забезпечувати низький рівень хибнопозитивних результатів (FPR), що означає, що легітимний електронний лист не повинен бути ідентифікований як фішинговий. Це є найважливішим показником у сфері виявлення фішингових електронних листів, оскільки легітимний лист, помилково класифікований як фішинг, може містити важливу інформацію для користувача, і його втрата може бути критичною. По-друге, метод повинен забезпечувати високий рівень справжніх позитивних результатів (TPR), щоб виявляти більшість фішингових електронних листів. І нарешті, з високим рівнем справжніх негативних результатів (TNR), метод повинен правильно ідентифікувати легітимні сайти та

вірно перевіряти, що електронний лист є легітимним. Ці характеристики є невід'ємною частиною надійної системи захисту від фішингових атак, що може суттєво підвищити безпеку онлайн-взаємодій користувачів.

Запропонований метод має здатність вивчати нові поведінки в будь-якому новому наборі даних, використовуючи інноваційний алгоритм FEaR. Метод буде поступово вдосконалюватися для обробки нових атак, спираючись на принципи навчання з підкріпленням. В основі методу виявлення лежить нейронна мережа, для створення якої запропоновано алгоритм DENNuRL, що дозволяє знайти оптимальну нейронну мережу для вирішення конкретної проблеми. Додатково, метод виявлення автоматично адаптується, щоб відображати зміни в нульових фішингових атаках.

Метод виявлення фішингових електронних листів в онлайн-режимі базується на техніках машинного навчання з наглядом і без нагляду. Техніка машинного навчання з наглядом використовує навчальний набір даних для побудови моделі виявлення, тоді як без нагляду намагається адаптувати цю модель, використовуючи електронні листи, які щойно надійшли в систему. Запропонований метод об'єднує нейронні мережі, навчання з підкріпленням, методи асоціативної класифікації та набір алгоритмів для виявлення фішингових атак.

Метод, як показано на рис. 3.1, починає роботу з витягнення характеристик з офлайн-даних, після чого застосовується алгоритм FEaR. Застосування алгоритму FEaR має кілька переваг: по-перше, він виявляє фішингові поведінки, що використовуються в офлайн-наборі даних; по-друге, зменшує складність згенерованої моделі, мінімізуючи кількість вхідних нейронів і зв'язків між вхідним і прихованим шарами; по-третє, прискорює процес адаптації для генерації найкращої нейронної мережі, що суттєво вплине на ефективність онлайн-системи; і, нарешті, пришвидшує процес класифікації. На наступному етапі буде використано DENNuRL, яка буде використовуватися в онлайн-режимі для класифікації електронних листів. Далі алгоритм RL-Agent буде безперервно адаптувати алгоритм PEDS, щоб відображати нові вивчені поведінки в онлайн-

режимі.

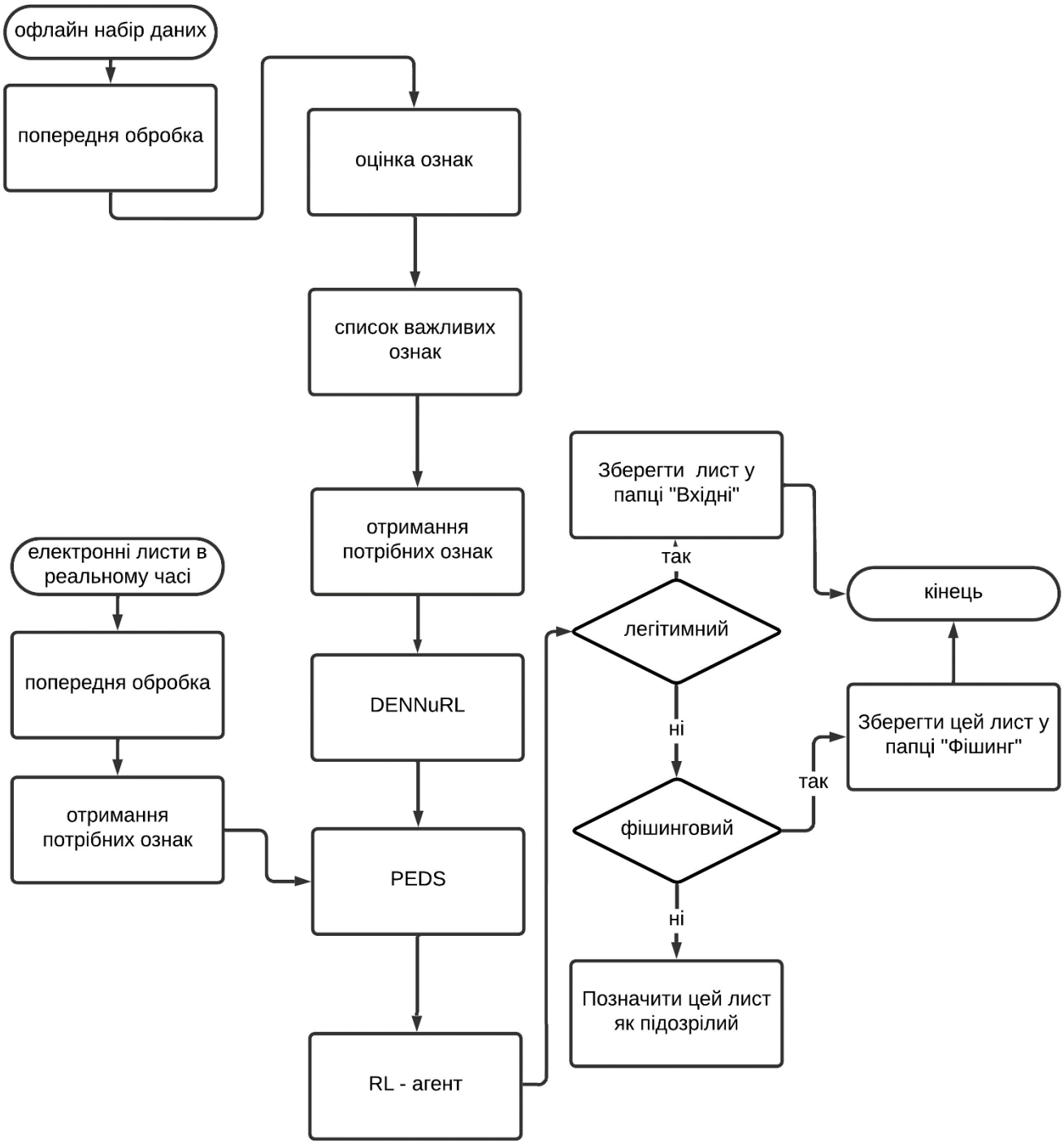

Рисунок 3.1 – Метод ідентифікації фішингових атак в електронних листах

Після створення першої моделі PEDS за допомогою алгоритму DENNuRL, основаного на навчальному наборі даних в офлайн-режимі, система починає процес навчання, читаючи некласифіковані електронні листи в онлайн-режимі. На рис. 3.2 показано алгоритм RL-Agent, за допомогою якого система класифікує

електронні листи на фішингові та легітимні. Алгоритм RL-Agent постійно моніторить вихідні дані PEDS в онлайн-режимі та працює за наступним принципом: електронні листи класифікуються по одному і передаються до системи оцінювання, а потім, якщо електронний лист класифіковано з дуже високою точністю, переходять до наступного етапу, в іншому випадку читається наступний електронний лист. У разі збору певної кількості електронних листів, вони об'єднуються в новий набір даних, і всі характеристики, витягнуті на етапі попередньої обробки, враховуються.

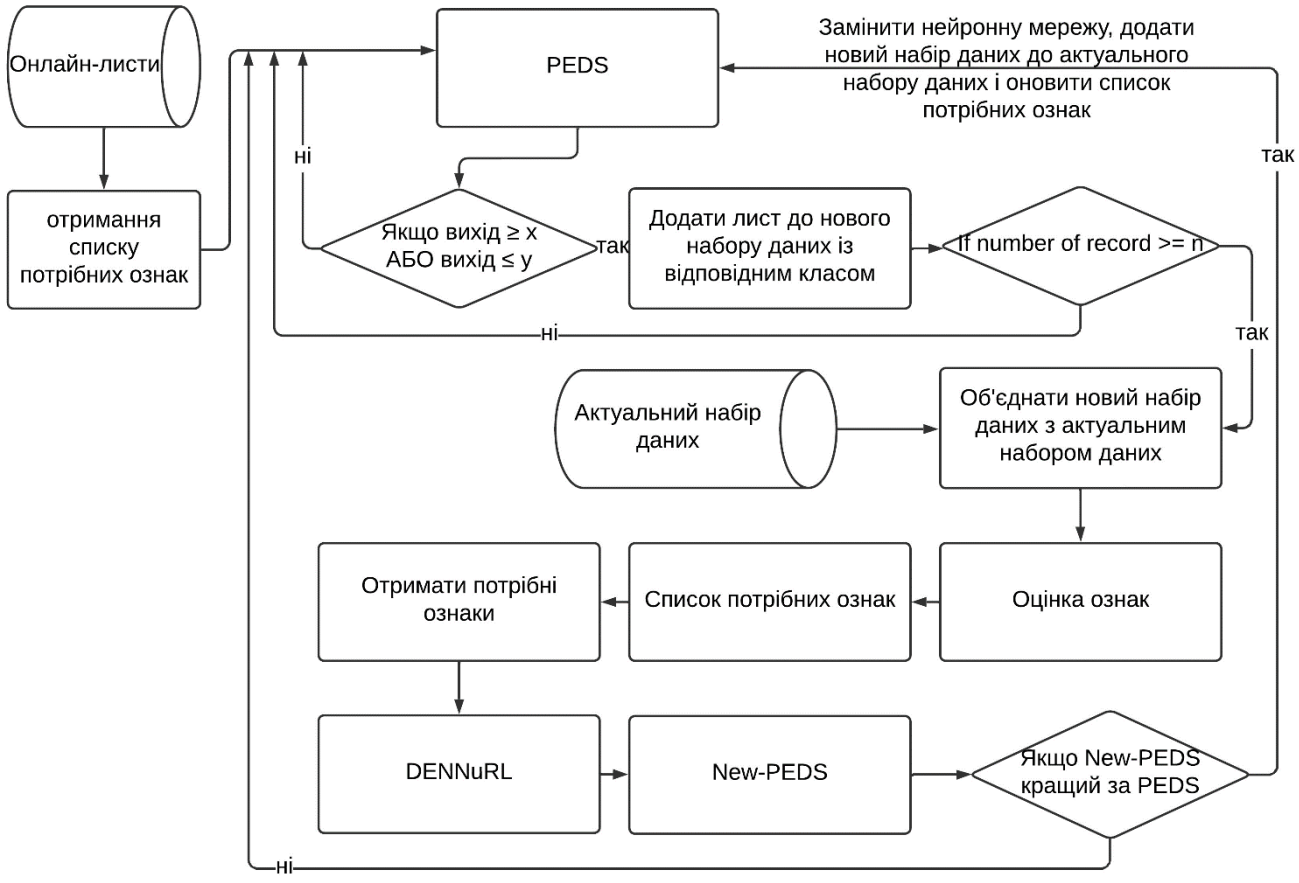

Рисунок 3.2 - Алгоритм RL-Agent

Система об'єднує новий набір даних з актуальним набором, а результати зберігаються в новому наборі даних. Алгоритм FEaR застосовується до нового набору для вивчення нових фішингових поведінок, після чого визначається список важливих характеристик. На основі нових даних генерується нова PEDS

за допомогою алгоритму DENNuRL, що відображає зміни в наборі даних. Нова модель перевіряється на контрольному наборі даних та актуальному наборі, і якщо нова PEDS демонструє кращі результати, вона замінює попередню модель, а актуальний набір даних оновлюється. У разі, якщо нова модель не покращує результати, система ініціалізується заново, щоб уникнути помилкових класифікацій.

Запропонована техніка адаптації PEDS має обмежену кількість параметрів. Значення цих параметрів налаштовуються вручну після попереднього тестування, і вони не є оптимальними. Однак вони не є чутливими до несуттєвих змін у середовищі. Крім алгоритму RL-Agent [54-55], що контролює поведінку DENNuRL.

Алгоритм RL-Agent здійснює дослідження, додаючи групу електронних листів до нового набору даних. Ці електронні листи класифікуються за допомогою поточної PEDS з високою точністю. Новий набір даних міститиме старий набір плюс нову групу електронних листів. Алгоритм RL-Agent генерує нову PEDS та порівнює її з попередньою версією. У випадку успішної класифікації нова PEDS крок за кроком удосконалюється, щоб адаптуватися до нових типів атак. Також експерт може покращити здатність системи до обробки нових атак, вручну класифікуючи підозрілі електронні листи та додаючи нові набори даних до офлайн-даних. Наприкінці, як було обговорено раніше, PEDS завжди тестується на контрольному та актуальному наборах даних, щоб підтвердити, що нова модель не підкріплює неправильні результати.

3.2 Опис алгоритму DENNuRL

Алгоритм DENNuRL (Dynamic Evolving Neural Network using Reinforcement Learning) розроблено для виявлення фішингових електронних листів, особливо в ситуаціях, коли зловмисники використовують нові, невідомі техніки, що утруднює їх виявлення звичайними методами. Він комбінує

нейронну мережу з підходом підкріплюючого навчання, що дозволяє моделі динамічно адаптуватися до змін у поведінці фішингових атак.

DENNuRL (рис. 3.3) починається з попередньої обробки даних, де обираються важливі характеристики листів для аналізу. Після цього алгоритм використовує підкріплююче навчання, щоб у реальному часі навчати мережу на нових даних, адаптуючи її для виявлення раніше невідомих фішингових шаблонів. Цей підхід дозволяє алгоритму з високою точністю виявляти фішингові листи і зберігати низький рівень хибнопозитивних спрацьовувань, що особливо важливо в онлайн-режимі, де небезпека від фішингових атак зростає щодня.

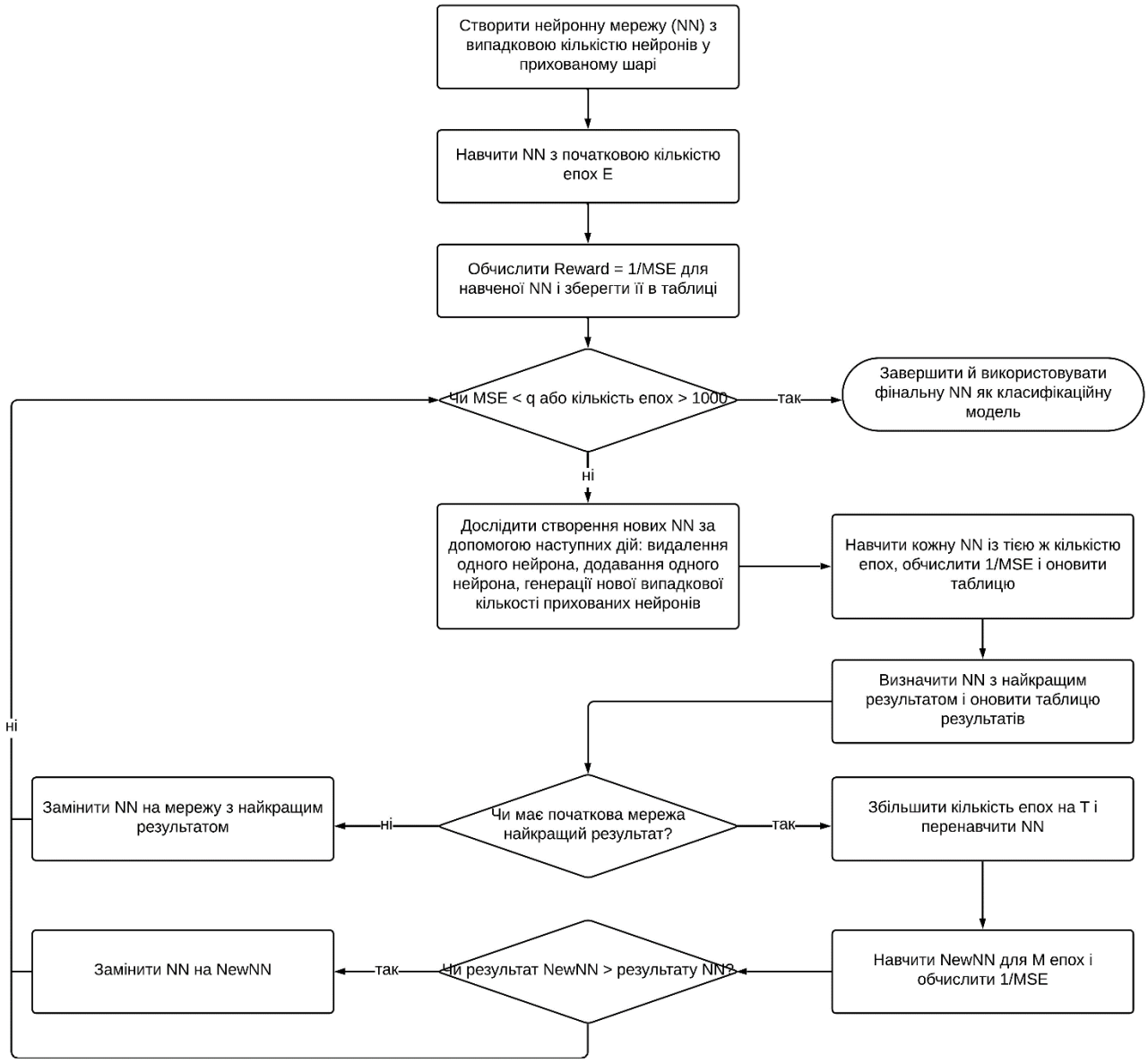

Рисунок 3.3 - Алгоритм DENNuRL

Створюється нейронна мережа, що містить випадкову кількість нейронів у прихованому шарі, тоді як кількість нейронів у вхідному та вихідному шарах визначається залежно від розв'язуваної задачі. Нейронна мережа, побудована на першому етапі, навчається на початковій кількості епох, що дорівнює десяти. Після навчання мережі проводиться тестування, під час якого обчислюється середньоквадратична помилка (СКП), що служить для визначення значення винагороди, адже воно є незалежним від розміру навчального набору. Якщо умова завершення не задовольняється, то перевіряється, чи СКП менша за певний поріг або чи кількість епох перевищує тисячу. Значення цього порогу вибирається вручну й залежить від допустимої похибки для конкретної задачі.

У запропонованому алгоритмі розглядаються три можливі дії, які змінюють стан системи. Дії стосуються прихованого шару нейромережі, оскільки вхідні та вихідні шари залежать від специфіки задачі. По-перше, з прихованого шару видаляється один нейрон шляхом об’єднання його з іншим нейроном. Для вибору нейронів для об’єднання визначається кореляція між ними на основі їх виходу на навчальному наборі. Найменш значущий нейрон об’єднується з найбільш корельованим, що не впливає на процес класифікації. По-друге, до нейромережі додається новий нейрон, після чого мережа перенавчається з новою архітектурою. Це робиться для того, щоб уникнути потрапляння в локальні мінімуми. По-третє, генерується нова випадкова кількість прихованих нейронів, і мережа також перенавчається. Ця дія є важливою для уникнення локальних мінімумів, оскільки нова архітектура досліджується на відстані від попередньої.

Кожна з нейромереж, досліджених на попередньому етапі, навчається, і оновлюється таблиця винагород, яка базується на попередньо визначеній формулі. Використовується метод підкріплення для вивчення можливих дій, що дозволяє обрати ту нейромережу, яка приносить максимальну результативність. Алгоритм Q-навчання визначає, яку дію реалізувати, залежно від отриманого результату. Порівнюється результат нової нейромережі зі старим, і якщо нова нейромережа має вище значення, то стара нейромережа замінюється. Якщо

результат нової нейромережі менший, то кількість епох збільшується на десять, створюється нова нейромережа з оновленою кількістю епох і проводиться навчання. Якщо нова нейромережа забезпечує вищий результат, то відбувається заміна; в іншому випадку, процес повертається до перевірки умов завершення. Врешті-решт, процес адаптації завершується, і фінальна нейромережа використовується як модель для класифікації.

3.3 Висновки до розділу

У третьому розділі представлено та детально описано метод ідентифікації фішингових атак в електронних листах, розроблений для вирішення проблем сучасних кіберзагроз. Основний акцент зроблено на евристичних підходах, які дозволяють виявляти фішингові листи навіть у тих випадках, коли традиційні методи на основі чорних списків не здатні захистити від нових, невідомих форм фішингу.

З метою створення надійної системи захисту, було проаналізовано та обрано кілька ефективних алгоритмів класифікації, таких як Naive Bayes, Random Forest, Логістична регресія та багатошаровий перцептрон. Кожен із них продемонстрував унікальні переваги та обмеження, що вплинули на рішення щодо розробки гібридної моделі, яка поєднує кілька підходів одночасно. Так, алгоритм Naive Bayes виявився ефективним у випадках із типовими фішинговими ознаками, однак його ефективність знижувалася, коли атаки були замасковані під легітимні листи. Random Forest, натомість, показав високу точність завдяки здатності аналізувати складні зв’язки між численними ознаками, що дозволяє знижувати ймовірність хибних спрацювань. Логістична регресія продемонструвала успішні результати в аналізі лексичних особливостей, які відрізняють фішингові повідомлення від легітимних, а багатошаровий перцептрон виявився здатним обробляти великі масиви даних, що дозволило виявляти приховані патерни, притаманні фішинговим листам.

Застосування евристичного підходу стало вирішальним кроком у розробці методу, який може ефективно виявляти фішинг, не покладаючись на попередньо відомі шаблони. Такий підхід дозволяє моделі розпізнавати патерни поведінки, які притаманні фішинговим атакам, а не лише статичні ознаки. Метод також аналізує підозрілі HTML-елементи, що дає змогу виявляти загрози на основі технічних характеристик електронних листів, таких як вбудовані скрипти, фальшиві форми або відсутність захищених протоколів. Це підвищує точність, яка стає менш залежною від оновлення чорних списків і здатною реагувати на нові загрози у режимі реального часу.

Завдяки поєднанню кількох алгоритмів метод забезпечує стабільно високу якість виявлення фішингових листів, дозволяючи обробляти великі обсяги даних швидко та без затримок, що є важливим для корпоративних систем. Використання цього підходу також знижує ризик як хибно позитивних, так і хибно негативних результатів, що підвищує точність та надійність системи.

Розроблений метод ідентифікації фішингових атак може бути інтегрований у захисні системи як самостійний компонент або в рамках більш масштабної системи захисту інформації. Такий підхід є доцільним для застосування у корпоративному, державному та освітньому середовищах, де конфіденційність інформації має критичне значення. Крім того, цей метод може ефективно використовуватися для забезпечення кібербезпеки в умовах дистанційного навчання, що є актуальним у сучасному світі.

Подальші перспективи розвитку методу можуть зосереджуватися на розширенні його функцій для виявлення фішингу не лише в електронних листах, а й в інших комунікаційних каналах, таких як SMS, VoIP та соціальні мережі. Це забезпечить уніфікований захист користувачів у різних середовищах, підвищуючи загальний рівень кібербезпеки.

4 ОЦІНЮВАННЯ МЕТОДУ ІДЕНТИФІКАЦІЇ ФІШИНГОВИХ АТАК В ЕЛЕКТРОННИХ ЛИСТАХ

4.1 Показники оцінювання

Для оцінки пропонованого методу виявлення фішингових листів використовуємо наступні показники. Матриця плутанини є основним інструментом для оцінки результатів роботи класифікатора фішингових листів. Містить такі показники, як:

– істинно позитивні (TP);

– істинно негативні (TN);

– хибно позитивні (FP);

– хибно негативні (FN) випадки.

Ці значення показують, наскільки точно метод може розрізняти фішингові листи від безпечних, допомагаючи аналізувати її ефективність у виявленні кіберзагроз. Визначення P=TP+FN (позитивні випадки) та N=TN+FP (негативні) додатково допомагає розподілити загальну кількість випадків на відповідні категорії [56-57].

Акуратність використовується для оцінки загальної якості моделі, що виявляє фішингові листи, і обчислюється як частка правильно класифікованих випадків серед загальної кількості. Для кібербезпеки цей показник дозволяє визначити, наскільки надійно модель захищає користувачів від фішингових загроз:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{4.1}$$

Точність - показник, що визначає частку листів, правильно визначених як фішингові, серед усіх, позначених як фішингові. Цей параметр є важливим у кібербезпеці, оскільки високий рівень Точність знижує кількість хибних спрацювань (коли безпечні листи помилково визначаються як фішингові):

$$Precision = \frac{TP}{TP + FP} \tag{4.2}$$

Повнота оцінює здатність класифікатора виявляти всі випадки фішингу серед усіх справжніх фішингових листів. У сфері безпеки важливо, щоб повнота була високою, адже вона показує, скільки загроз фішингу не залишилося непоміченими:

$$Recall = \frac{TP}{TP + FN} \tag{4.3}$$

F-міра - гармонійне середнє точності та повноти. У випадку фішингу цей показник дозволяє досягти збалансованої оцінки, уникаючи крайнощів. Наприклад, якщо модель матиме високу прецизію, але низьку повноту, фішингові атаки можуть залишитися непоміченими. Формула:

$$F\ score = \frac{Recall + Precision}{2} \tag{4.4}$$

Геометричне середнє (G-Mean) використовується для балансування виявлення фішингових листів (показник позитивних) та уникнення хибних спрацьовувань (показник негативних). Цей показник допомагає зрозуміти, наскільки добре модель зберігає рівновагу між ефективністю виявлення загроз і відсутністю зайвих спрацювань:

$$G - Mean = \sqrt{\frac{TP}{TP + FN} * \frac{TN}{TN + FP}} \tag{4.5}$$

У кібербезпеці площа під кривою (AUC) допомагає оцінити загальну здатність моделі вірно відрізняти фішингові листи від безпечних. Високе значення AUC свідчить про високу ймовірність того, що класифікатор коректно розподілить випадкові зразки між позитивним і негативним класами.

Коефіцієнт кореляції Меттьюса (MCC) враховує всі можливі категорії (TP, TN, FP, FN) і є універсальним для ситуацій з нерівними класами. MCC показує загальну збалансованість моделі при класифікації фішингових листів, навіть якщо їх кількість значно менша за безпечні листи:

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TN + FN)(P)(N)}} \quad (4.6)$$

Ці показники є основою для точного та збалансованого виявлення фішингових загроз і підвищення рівня безпеки в інформаційному середовищі.

4.2 Аналіз ефективності

Експерименти проводилися з використанням набору даних, сформованого з трьох загальнодоступних наборів, описаних раніше в параграфі 2.4. Щоб навчити та протестувати метод, набір даних було розділено на офлайн- і онлайн-частини. Онлайн-набір вважався некласифікованим, і загальна кількість електронних листів становила 4951 для звичайних листів та 7315 для фішингових листів. У кожному експерименті випадково обиралися 4951 фішинговий лист. Загалом було 9902 електронні листи, з яких 4000 обирали випадковим чином як офлайн-набір. Решта листів використовувалася як онлайн-набір для оцінки ефективності системи у відповідь на фішингові атаки нульового дня. Експеримент проводився та повторювався п'ятдесят разів, і в кожній ітерації офлайн-набір вибирався у випадковому порядку та поділявся на 70% для навчання, 15% для валідації та 15% для тестування. Результати фіксували для кожної ітерації, після чого обчислювали середнє значення для всіх ітерацій, щоб забезпечити надійну роботу системи.

Як обговорювалося раніше, алгоритм DENNuRL, використовується для створення першої моделі PEDS, яка працює в онлайн-режимі. Пізніше, в онлайн-режимі, цей самий алгоритм застосовується як частина агента з

підкріплювальним навчанням (RL-Agent). В обох випадках алгоритм автоматично обиратиме найкращу нейронну мережу для вирішення поставленої задачі.

Перша модель PEDS створюється за допомогою алгоритму DENNuRL на основі офлайн-набору даних із 4000 електронних листів. Як приклад застосування алгоритму DENNuRL, у таблиці 4.1 показано покращення нейронної мережі від випадково згенерованої на першому етапі до фінальної версії, яка використовується як PEDS. Процес удосконалення керується рівнем помилки MSE, причому мережа з найнижчим значенням MSE вважається кращою. Переваги таких невеликих змін у помилці MSE стануть очевидними, коли буде враховано інші метрики оцінювання.

Таблиця 4.1 - Приклад адаптації NN за допомогою алгоритму DENNuRL

№	Кількість нейронів у прихованому шарі	Номер епохи	Помилка MSE
1	27	10	0,0243
2	37	10	0,0232
3	38	10	0,0231
4	39	10	0,0230
5	6	10	0,0227
6	7	10	0,0222
7	7	60	0,0128
8	8	60	0,0112
9	8	360	0,0107
10	20	610	0,0106
11	19	610	0,0106
12	19	1010	0,0099

Таблиця 4.1 та рисунок 4.1 демонструють покращення, виконане алгоритмом DENNuRL, в контексті зменшення помилки MSE, відповідно до розвитку системи.

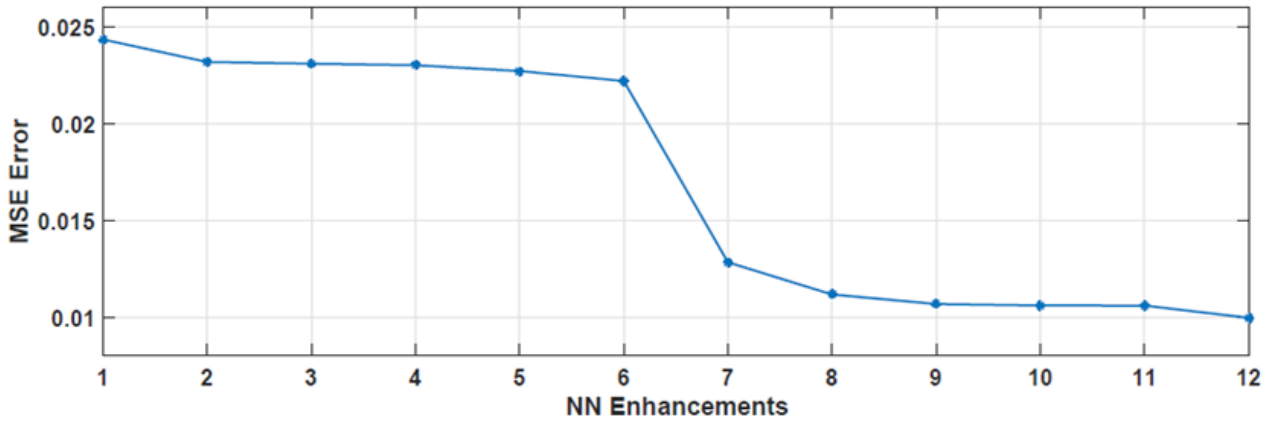

Рисунок 4.1 - Приклад покращення NN за допомогою DENNuRL з точки зору помилки MSE

У першій нейронній мережі, показаній у таблиці 4.1, алгоритм випадковим чином обрав 27 нейронів, після чого система пробує різні дії та обирає мережу, яка забезпечує максимальну винагороду (мінімальну помилку MSE). У другому покращенні обирається нова випадкова нейронна мережа, а в третьому додається ще один нейрон у прихованому шарі. На 11-му етапі два нейрони об’єднуються в прихованому шарі, а в останньому покращенні збільшується кількість епох навчання для зниження помилки MSE.

Після створення першої системи PEDS за допомогою алгоритму DENNuRL, вона розпочинає класифікацію онлайн-даних. Під час класифікації електронної пошти в режимі онлайн агент з підкріпленням (RL-агент) стежить за результатами системи. Для порівняння здатності системи PEDS виявляти фішингові листи до і після адаптації використовується крива компромісу помилок виявлення (Detection Error Trade-off, або DET). Крива DET є графічним зображенням співвідношення двох типів помилок: пропущених виявлень (по осі y) і хибних спрацювань (по осі x). Пропущене виявлення (помилково-негативний результат) виникає, коли фішинговий лист класифікується як легітимний, а

хибне спрацювання (помилково-позитивний результат) – коли легітимний лист помилково визначається як фішинговий.

Обидва типи помилок були проаналізовані для двох версій PEDS на випадковій вибірці з початкового набору даних. Перша версія — це початкова PEDS до адаптації, а друга — адаптована система після впровадження RL-агента. Крива DET надає цінну інформацію про ефективність системи на різних операційних точках. Як показано на графіку (рис. 4.2), адаптована система демонструє меншу ймовірність як пропущених виявлень, так і хибних спрацювань у всіх точках.

Щоб створити криву DET, поріг прийняття рішень поступово змінюється у межах значень, згенерованих системою виявлення, а ефективність оцінюється на малих інтервалах. Потім графічно відображаються нормальні відхилення між рівнями пропущених виявлень та хибних спрацювань на кожному інтервалі. Функція компромісу помилок виявлення, що приймає список прогнозованих значень і список реальних значень, видає рівень хибних тривог (FPR) і пропущених виявлень (FNR) для всіх порогових значень. Лінія, яка відображає компроміс між FPR і FNR, зазвичай подається в логарифмічній системі координат. Крива DET демонструє зв’язок між рівнями хибних спрацювань (по осі x) та пропущених виявлень (по осі y), який математично визначається рівнянням 4.7:

$$y = -(\frac{1}{\sigma_T}x + \frac{\mu_T}{\sigma_T}), \tag{4.7}$$

де μ_T та σ_T – параметри цільового розподілу, який вважається Гаусовим.

Точка на кривій DET, де ймовірність помилкових спрацювань дорівнює ймовірності пропущеного виявлення (цільової електронної пошти), називається EqualError Rate (EER). EER можна оцінити за допомогою формули [58]:

$$x = y = -\frac{\mu_T}{1+\mu_T} \; and \; P_{EER} = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-t^2/2}dt, \tag{4.8}$$

Аналізуючи криву DET, можна підтвердити, що адаптована система PEDS ефективніша з точки зору обох типів помилок. Для визначення якості роботи системи використовують зважене середнє частоти пропущених виявлень та хибних спрацювань. Точка на кривій DET, у якій таке середнє мінімізується, позначається як рівень рівнозначної похибки (EER), і на рис. 4.2 вона показана стрілками.

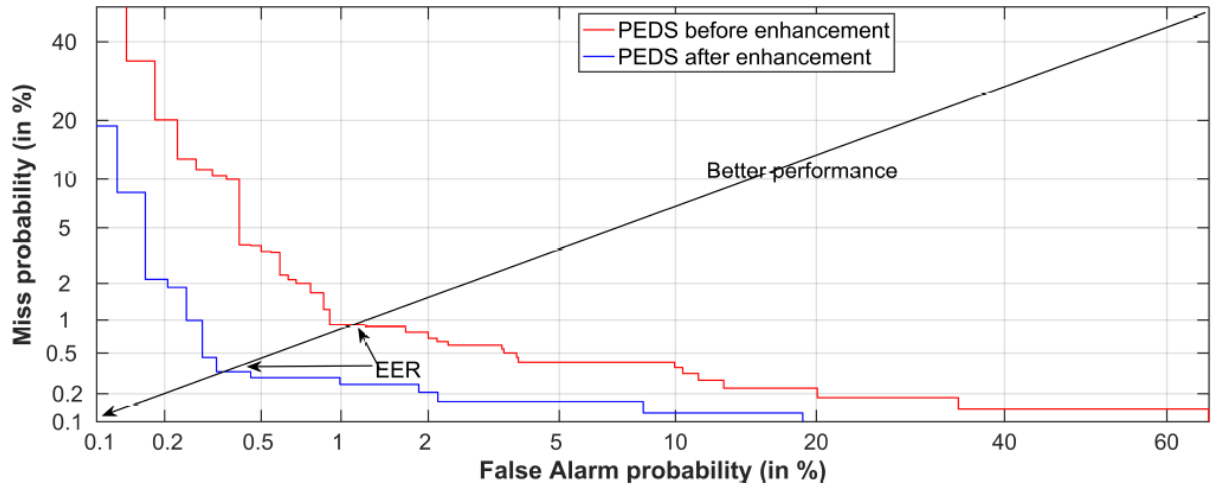

Рисунок 4.2 - Компроміс помилки виявлення (DET) для PEDS до і після адаптації

У таблицях 4.2 та 4.3 наведено узагальнення покращень у PEDS. Щоб продемонструвати еволюцію системи в онлайн-режимі, після кожної адаптації обчислювали вісім показників, зокрема точність, прецизію, повноту, F-міру, а також TPR, FPR, TNR і FNR.

Рисунок 4.3 ілюструє покращення точності, яке відбулося внаслідок п’ятнадцяти послідовних оновлень під час обробки онлайн-набору з 5902 електронних листів, і результати чітко демонструють поступове покращення всіх показників. Для підтвердження стабільної ефективності моделі експеримент повторили п’ятдесят разів, починаючи кожен з випадкової точки, після чого було створено кінцеву версію PEDS з урахуванням результатів тестування покращень.

У таблицях 4.4 та 4.5 узагальнено показники, що використовувались для оцінки продуктивності PEDS; для отримання середнього значення результати

експерименту також усереднені за всіма п’ятдесятьма повтореннями.

Таблиця 4.2 - Покращення PEDS з точки зору FNR, FPR, TPR, Accuracy, Precision, Recall, and F_Measure

№	FNR, %	FPR, %	TPR, %	TNR, %
1	1.89	5.15	98.11	94.85
2	1.89	4.82	98.11	95.18
3	1.52	5.15	98.48	94.85
4	1.52	5.15	98.48	94.85
5	1.52	5.11	98.48	94.89
6	1.56	5.11	98.44	94.89
7	1.60	4.36	98.40	95.64
8	1.60	4.48	98.40	95.52
9	1.48	3.57	98.52	96.43
10	1.07	2.82	98.93	97.18
11	1.07	2.82	98.93	97.18
12	1.11	2.82	98.89	97.18
13	0.99	2.62	99.01	97.38
14	1.03	2.03	98.97	97.97
15	1.15	1.74	98.85	98.26

Таблиця 4.3 - Покращення PEDS з точки зору FNR, FPR, TPR, Accuracy, Precision, Recall, and F_Measure

№	ACC, %	Precision, %	Recall, %	F_Measure, %
1	2	3	4	5
1	96.49	95.06	98.11	96.56
2	96.65	95.36	98.11	96.72
3	96.67	95.08	98.48	96.75
4	96.67	95.08	98.48	96.75

Кінець таблиці 4.3

1	2	3	4	5
5	96.69	95.11	98.48	96.77
6	96.67	95.11	98.44	96.75
7	97.02	95.79	98.40	97.08
8	96.96	95.68	98.40	97.02
9	97.48	96.53	98.52	97.52
10	98.06	97.25	98.93	98.08
11	98.06	97.25	98.93	98.08
12	98.04	97.25	98.89	98.06
13	98.2	97.45	99.01	98.22
14	98.47	98.00	98.97	98.49
15	98.55	98.28	98.85	98.56

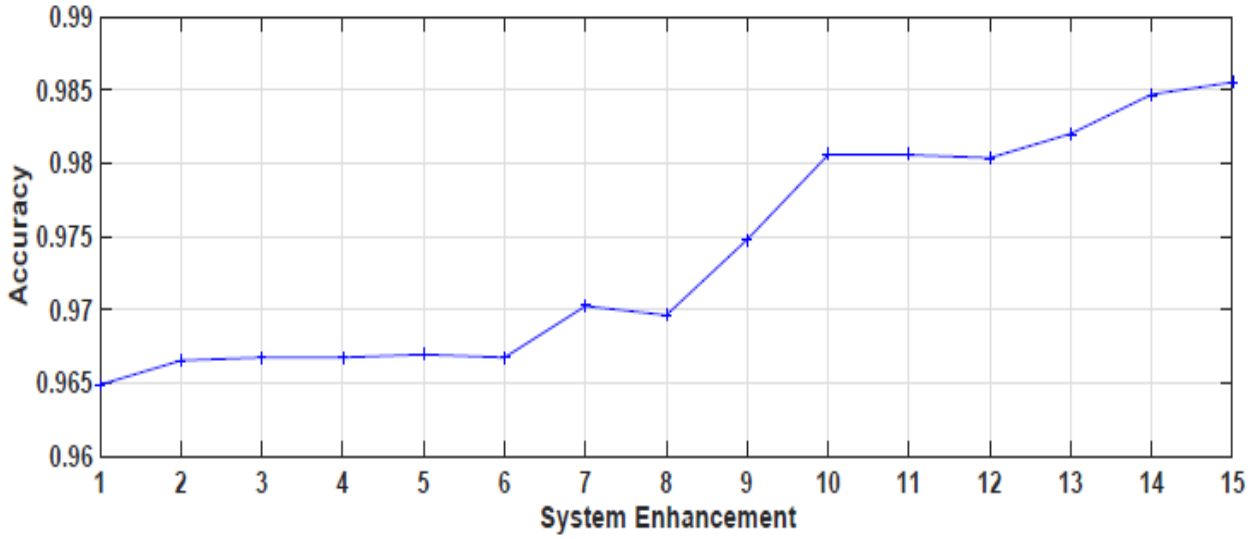

Рисунок 4.3 - Покращення точності системи

Таблиця 4.4 - Результати онлайн-тестування PEDS для параметрів FNR, FPR, TPR, TNR та ACC

FNR	FPR	TPR	TNR	ACC
0.93%	1.81%	99.07%	98.19%	98.63%

Таблиця 4.5 - Результати онлайн-тестування PEDS для параметрів Precision, Recall, F_Measure та AUC

Precision	Recall	F_Measure	AUC
98.21%	99.07%	98.64%	99.43%

4.3 Висновок до розділу

У четвертому розділі основна увага зосереджується на експериментальному тестуванні та аналізі результатів, що підтверджують доцільність використання розробленого методу для боротьби з фішингом. Для оцінки продуктивності методу застосовувався метод 10-кратної крос-валідації. Завдяки цьому вдалось уникнути залежності моделі від конкретних вибірок даних та гарантувати надійність її роботи в умовах змінної інформаційної бази. Отримані результати підтверджують, що запропонована модель здатна з високою точністю ідентифікувати фішингові листи, при цьому ефективно розрізняючи їх від звичайних повідомлень, що особливо важливо для забезпечення повсякденного захисту електронної пошти користувачів.

Одним із важливих аспектів роботи методу є її здатність знижувати кількість хибнопозитивних та хибнонегативних результатів. Запропонований метод показав здатність зменшити кількість таких помилок завдяки поєднанню кількох алгоритмів, що працюють паралельно та забезпечують більш точне визначення небезпечного листа. Це дає змогу не лише знизити ризик пропуску загроз, але й зменшити вплив фішингових атак на кінцевого користувача та захистити його конфіденційну інформацію.

Для оцінки ефективності методу використовувались також різні набори даних, що включають легітимні та фішингові листи. Такі дані давали можливість протестувати модель в умовах, максимально наближених до реальних, адже лише практичні тести на змішаних базах даних дозволяють повною мірою зрозуміти її продуктивність.

Експериментальні результати свідчать, що метод має високу точність, незалежно від набору даних, на якому він тестувався, що підтверджує його адаптивність до нових і незнайомих фішингових атак. Особливо важливо, що метод успішно виявляє фішингові атаки нульового дня, які є новими загрозами, що не можуть бути ідентифіковані за допомогою традиційних підходів, заснованих на чорних списках. Це забезпечується за рахунок використання комплексного підходу, який аналізує поведінкові ознаки листів, їх структуру, зміст та метадані, що дає змогу виявляти фішингові загрози, навіть якщо вони раніше не зустрічались у відомих наборах даних. Крім того, значною перевагою методу є його здатність до швидкої обробки даних, що дозволяє оперативно реагувати на потенційні загрози в умовах реального часу.

Висока швидкість роботи є критичною для сучасних систем кіберзахисту, особливо у сфері електронної пошти, де затримка у виявленні фішингового листа може призвести до серйозних наслідків для користувачів або навіть організацій. Запропонований метод демонструє здатність до обробки великої кількості даних за короткий час, що робить його перспективним інструментом для масштабних систем кібербезпеки, які повинні забезпечувати захист не лише індивідуальних користувачів, а й корпоративного середовища, де електронна пошта є одним із головних каналів комунікації. Ще однією перевагою розробленого методу є його можливість до адаптації та самонавчання на основі нових даних. Це дозволяє методу зберігати свою актуальність і ефективність у довготривалій перспективі, навіть якщо методи фішингу еволюціонуватимуть та набуватимуть нових форм.

ВИСНОВКИ

Кваліфікаційна робота присвячена розробці нового методу для виявлення фішингових атак в електронних листах. Основна загроза фішингу полягає у здатності зловмисників використовувати електронну пошту для викрадення конфіденційної інформації користувачів. Це може призводити до серйозних фінансових втрат, порушення конфіденційності даних та зниження довіри до цифрових комунікаційних каналів. У контексті стрімкого розвитку технологій та збільшення обсягів інтернет-транзакцій потреба у створенні надійної системи захисту від фішингових атак є надзвичайно актуальною.

Важливим аспектом роботи є розробка методу попередньої обробки електронних листів, що забезпечує виділення ознак фішингових повідомлень. На початкових етапах дослідження було визначено найбільш поширені ознаки фішингових листів, які включають підозрілий зміст, відсутність захищеного протоколу, а також спроби маніпуляції користувачами за допомогою шкідливих посилань або неправдивих повідомлень. На основі цих спостережень була розроблена система, яка виділяє найважливіші характеристики електронних листів, що сприяє покращенню точності класифікації повідомлень. Етап виділення ознак є ключовим, оскільки він закладає основу для наступного аналізу та забезпечує ефективне функціонування моделі.

Особливу увагу було приділено оптимізації кількості ознак, оскільки великий обсяг даних може ускладнювати аналіз і знижувати швидкість роботи методу. Запропонований метод оптимізації зменшує кількість параметрів, зберігаючи при цьому їх інформативну цінність для класифікації фішингових листів. Це дозволяє суттєво прискорити обробку даних, що особливо важливо у реальному часі. Завдяки оптимізації вдалося скоротити обсяг аналізованих характеристик до найсуттєвіших параметрів, зберігаючи високу точність і надійність моделі.

У дослідженні проведено оцінку різних алгоритмів класифікації, що могли б забезпечити найбільш ефективне виявлення фішингових атак. На основі

порівняння низки популярних алгоритмів, таких як Naive Bayes, Random Forest і Логістична регресія, було обрано оптимальний варіант для реалізації моделі. Гібридний підхід до класифікації забезпечує баланс між точністю та швидкістю ідентифікації, що дозволяє методу працювати в режимі реального часу та виявляти нові типи фішингових атак, які можуть обходити стандартні засоби захисту. Такий підхід забезпечує адаптивність методу до нових загроз та знижує ризик пропуску невідомих фішингових атак, особливо атак нульового дня, які не можуть бути виявлені традиційними методами на основі чорних списків.

Експериментальні результати роботи підтвердили, що розроблена модель досягла високих показників точності та надійності у виявленні фішингових листів. Для перевірки було використано кілька наборів даних, які містять як легітимні, так і фішингові електронні листи. Це дозволило забезпечити реалістичні умови тестування та оцінити здатність методу до правильного розпізнавання фішингових повідомлень. Запропонований метод показав значне зниження кількості хибно позитивних і хибно негативних результатів, що підвищує його надійність і знижує ймовірність помилкових блокувань легітимних листів. Така ефективність важлива для забезпечення захисту користувачів у корпоративному середовищі, де некоректне блокування важливих листів може спричинити збої у комунікації.

Наукова новизна дослідження полягає у запропонованому підході, що поєднує методи попередньої обробки, евристичні методи і сучасні алгоритми класифікації. Це дозволяє створити метод, який здатний адаптуватися до нових загроз і забезпечувати ефективний захист без необхідності постійного оновлення чорних списків. Такий підхід є вагомим кроком вперед у боротьбі з фішинговими атаками та може стати основою для подальших досліджень у сфері кібербезпеки.

Практична значущість отриманих результатів є очевидною, оскільки вони можуть бути впроваджені для підвищення рівня безпеки в електронній пошті, яка продовжує залишатися важливим каналом комунікації для бізнесу та приватних користувачів. Результати цієї роботи можна застосовувати для створення нових програмних продуктів або вдосконалення існуючих рішень для

захисту від фішингових атак. Впровадження розробленого методу сприятиме не лише зниженню кількості успішних фішингових атак, а й підвищенню довіри до електронної пошти як засобу комунікації, що є важливим у цифрову епоху.

Результати дослідження підтвердили, що використання алгоритмів машинного навчання для виявлення фішингових атак є перспективним напрямом розвитку кібербезпеки. Гнучкість та адаптивність методу забезпечують можливість для її подальшого вдосконалення, зокрема розширення функціоналу для роботи з іншими каналами комунікації, такими як SMS або месенджери.

У підсумку, робота є внеском у галузь кібербезпеки, зокрема у захист від фішингових атак. Запропонована модель демонструє високу ефективність, здатність до адаптації та можливість забезпечення захисту користувачів у реальних умовах. Розроблений метод до виявлення фішингових листів на основі виділення ознак, оптимізації кількості параметрів та використання сучасних алгоритмів класифікації дозволяє створити універсальну систему захисту, що не лише підвищує рівень безпеки, а й відкриває нові перспективи для подальшого розвитку у сфері захисту інформації.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Definition of phishing noun from the Oxford Advanced American Dictionary. URL: https://www.oxfordlearnersdictionaries.com/definition/american_english/phishing (дата звернення 5.06.2024)

2. PhishTank.Join the fight against phishing URL: https://phishtank.org/ (дата звернення 5.06.2024)

3. The New Face of Phishing. URL: https://apwg.org/the-new-face-of-phishing/ (дата звернення 5.06.2024)

4. Закон України "Про електронні комунікації" від 16 грудня 2020 року № 1089-IX. Останні зміни № 3994-IX від 08.10.2024. URL: https://zakon.rada.gov.ua/laws/show/1089-20#Text

5. The Definition of Phishing. URL: https://www.phishlabs.com/blog/the-definition-of-phishing (дата звернення 11.06.2024)

6. What is phishing? URL: https://www.ibm.com/topics/phishing (дата звернення 12.06.2024)

7. How to spot and protect yourself from a phishing attack. URL: https://cybersecurityguide.org/resources/phishing/ (дата звернення 14.06.2024)

8. Які типові ознаки фішингових електронних листів. URL: https://nadiyno.org/yaki-typovi-oznaky-fishyngovyh-elektronnyh-lystiv/ (дата звернення 23.06.2024)

9. Ознаки листів та повідомлень з фішинговими посиланнями. URL: https://harazd.bank.gov.ua/article/sahrajstvo/platizna-bezpeka/oznaki-listiv-ta-povidomlen-z-fisingovimi-posilannami (дата звернення 27.06.2024)

10. How to Spot a Phishing Email. URL: https://www.crowdstrike.com/en-us/cybersecurity-101/social-engineering/how-to-spot-a-phishing-email/ (дата звернення 28.06.2024)

11. S. Laptiev. Удосконалений метод захисту персональних даних від атак за допомогою алгоритмів соціальної інженерії. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*. 2022. Вип. 4, ст. 45–62. DOI:

10.28925/2663-4023.2022.16.4562

12. The Three Stages Of a Phishing Attack - Bait, Hook And Catch URL: https://blog.usecure.io/three-steps-of-phishing (дата звернення 28.06.2024)

13. The Lifecycle of a Phishing Attack: How Cybercriminals Bait, Hook, and Exploit. URL: https://medium.com/@kisalnelaka6/the-lifecycle-of-a-phishing-attack-how-cybercriminals-bait-hook-and-exploit-e05cce7e4f5f (дата звернення 29.06.2024)

14. Основні типи фішингових атак. URL: https://www.issp.ua/post/phishing (дата звернення 2.09.2024)

15. The most common types of phishing attacks and their signs. URL: https://datalabsua.com/en/the-most-common-types-of-phishing-attacks-and-their-signs/ (дата звернення 3.09.2024)

16. 19 Types Of Phishing Attacks. URL: https://www.fortinet.com/resources/cyberglossary/types-of-phishing-attacks (дата звернення 3.09.2024)

17. Phishing attack timeline: 21 hours from target to detection. URL: https://www.infosecinstitute.com/resources/phishing/phishing-attack-timeline-21-hours-from-target-to-detection/ (дата звернення 4.09.2024)

18. 84% of Phishing Sites Last for Less Than 24 Hours. URL: https://www.infosecurity-magazine.com/news/84-of-phishing-sites-last-for-less/ (дата звернення 5.09.2024)

19. Alkhalil Z, Hewage C, Nawaf L and Khan I. Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. *Front. Comput. Sci*. 2021. Vol. 3. DOI: 10.3389/fcomp.2021.563060

20. Pawankumar Sharma, Bibhu Dash, Meraj Farheen Ansari. Anti-Phishing Techniques. A Review of Cyber Defense Mechanisms. *International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified Impact Factor 7.39.* 2022. Vol. 11, Issue 7

21. Bilal Naqvi, Kseniia Perova, Ali Farooq, Imran Makhdoom, Shola Oyedeji, Jari Porras. Mitigation strategies against the phishing attacks: A systematic literature review. *Computers & Security*, 2023. Vol. 132. DOI: 10.1016/j.cose.2023.103387.

22. D. Li, Q. Chen, and L. Wang. Phishing Attacks: Detection and Prevention Techniques. *Journal of Industrial Engineering & Applied Science*. 2024. Vol. 2, No. 4, PP. 48–53.

23. S. Kumar Birthriya, A. K. Jain. A Comprehensive Survey of Phishing Email Detection and Protection Techniques. *Information Security Journal: A Global Perspective*. 2021, Vol. 31, PP. 411–440. DOI: 10.1080/19393555.2021.1959678

24. R. Sahay, W. Meng, W. Li. A Comparative Analysis of Phishing Tools: Features and Countermeasures. *Information Security Practice and Experience. ISPEC 2024. Lecture Notes in Computer Science*. Vol. 15053. DOI: 10.1007/978-981-97-9053-1_21

25. 15 найкращих засобів перевірки фішингу веб-сайтів. URL: https://www.morningdough.com/uk/ai-tools/best-website-phishing-checker/ (дата звернення 10.09.2024)

26. T.O. Ojewumi, G.O. Ogunleye, B.O. Oguntunde, O. Folorunsho, S.G. Fashoto, N. Ogbu. Performance evaluation of machine learning tools for detection of phishing attacks on web pages. *Scientific African*. 2022. Vol. 16. DOI: 10.1016/j.sciaf.2022.e01165.

27. Alghenaim F., Abu Bakar M. A., Abdul Rahim F. Anti-Phishing Tools: State of the Art and Detection Efficiencies. *Applied Mathematics & Information Sciences*. 2022. Vol. 16, Iss. 6, Article 9. DOI: http://dx.doi.org/10.18576/amis/160609

28. The Challenges of Phishing Detection. URL: https://www.vadesecure.com/en/blog/the-challenges-of-phishing-detection-part-1 (дата звернення 12.09.2024)

29. Білий список (список дозволених): важлива інформація для звичайних користувачів. URL: https://plisio.net/uk/blog/whitelist-allowlist (дата звернення 12.09.2024)

30. Bayer J., Maroofi S., Hureau O., Duda A., Korczynski M. Building a Resilient Domain Whitelist to Enhance Phishing Blacklist Accuracy. *2023 APWG Symposium on Electronic Crime Research (eCrime), Barcelona, Spain.* 2023, PP. 1–14. DOI: 10.1109/eCrime61234.2023.10485549

31. Aljofey A., Jiang Q., Rasool A., et al. An effective detection approach for phishing websites using URL and HTML features. *Sci Rep*. 2022. Vol. 12. DOI: 10.1038/s41598-022-10841-5

32. Mankar N. P., Sakunde P. E., Zurange S., Date A., Borate V., Mali Y. K. Comparative Evaluation of Machine Learning Models for Malicious URL Detection. *2024 MIT Art, Design and Technology School of Computing International Conference (MITADTSoCiCon), Pune, India*. 2024. PP. 1–7. DOI: 10.1109/MITADTSoCiCon60330.2024.10575452

33. Leka C., Ntantogian C., Karagiannis S., Magkos E., Verykios V. S. A Comparative Analysis of VirusTotal and Desktop Antivirus Detection Capabilities. *2022 13th International Conference on Information, Intelligence, Systems & Applications (IISA), Corfu, Greece*. 2022, PP. 1–6. DOI: 10.1109/IISA56318.2022.9904382

34. Čermák M. Why Human Firewall Fails in the Battle with Sophisticated Spear Phishing Campaigns. *Trends and Future Directions in Security and Emergency Management. Lecture Notes in Networks and Systems. Cham: Springer*. 2022. Vol. 257. DOI: 10.1007/978-3-030-88907-4_16

35. Tudosi A.-D., Graur A., Balan D. G., Potorac A. D. An Email Classification Framework for Phishing Detection in Virtualized Network Environments. *2023 22nd RoEduNet Conference: Networking in Education and Research (RoEduNet), Craiova, Romania.* 2023. PP. 1–5. DOI: 10.1109/RoEduNet60162.2023.10274915

36. Jayaprakash R., Natarajan K., Daniel J. A., et al. Heuristic machine learning approaches for identifying phishing threats across web and email platforms. *Front. Artif. Intell*. 2024. Vol. 7, Article 1414122. DOI: 10.3389/frai.2024.1414122

37. Elgharbi S. E., Ait Yahia M., Ouchani S. Online Phishing Detection: A Heuristic-Based Machine Learning Framework. *2024 13th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro*. 2024. PP. 1–4. DOI: 10.1109/MECO62516.2024.10577848

38. Shin K., Ishikawa T., Liu Y.-L., Shepard D. L. Learning DOM Trees of Web Pages by Subpath Kernel and Detecting Fake e-Commerce Sites. *Machine Learning and Knowledge Extraction*. 2021. Vol. 3, Iss. 1, PP. 95–122. DOI: 10.3390/make3010006

39. Bu S.-J., Cho S.-B. Deep Character-Level Anomaly Detection Based on a Convolutional Autoencoder for Zero-Day Phishing URL Detection. *Electronics*. 2021. Vol. 10, Iss. 12, Article 1492. DOI: 10.3390/electronics10121492

40. Yang R., Zheng K., Wu B., Wu C., Wang X. Phishing Website Detection Based on Deep Convolutional Neural Network and Random Forest Ensemble Learning. *Sensors*. 2021. Vol. 21, Iss. 24, Article 8281. DOI: 10.3390/s21248281

41. Almseidin M., Alkasassbeh M., Alzubi M., Al-Sawwa J. Cyber-Phishing Website Detection Using Fuzzy Rule Interpolation. *Cryptography*. 2022. Vol. 6, Iss. 2, Article 24. DOI: 10.3390/cryptography6020024

42. Kocyigit E., Korkmaz M., Sahingoz O. K., Diri B. Enhanced Feature Selection Using Genetic Algorithm for Machine-Learning-Based Phishing URL Detection. *Applied Sciences*. 2024. Vol. 14, Iss. 14, Article 6081. DOI: 10.3390/app14146081

43. Yoon J.-H., Buu S.-J., Kim H.-J. Phishing Webpage Detection via Multi-Modal Integration of HTML DOM Graphs and URL Features Based on Graph Convolutional and Transformer Networks. *Electronics*. 2024. Vol. 13, Iss. 16, Article 3344. DOI: 10.3390/electronics13163344

44. Chen Z., Liu S.-Z., Huang J., Xiu Y.-H., Zhang H., Long H.-X. Ethereum Phishing Scam Detection Based on Data Augmentation Method and Hybrid Graph Neural Network Model. *Sensors*. 2024. Vol. 24, Iss. 12, Article 4022. DOI: 10.3390/s24124022

45. Su M.-Y., Su K.-L. BERT-Based Approaches to Identifying Malicious URLs. *Sensors*. 2023. Vol. 23, Iss. 20, Article 8499. DOI: 10.3390/s23208499

46. Alnemari S., Alshammari M. Detecting Phishing Domains Using Machine Learning. *Applied Sciences*. 2023. Vol. 13, Iss. 8, Article 4649. DOI: 10.3390/app13084649

47. Samad S. R. A., Balasubaramanian S., Al-Kaabi A. S., et al. Analysis of the Performance Impact of Fine-Tuned Machine Learning Model for Phishing URL Detection. *Electronics*. 2023. Vol. 12, Iss. 7, Article 1642. DOI: 10.3390/electronics12071642

48. Zhou J., Cui H., Li X., Yang W., Wu X. A Novel Phishing Website Detection

Model Based on LightGBM and Domain Name Features. *Symmetry*. 2023. Vol. 15, Iss. 1, Article 180. DOI: 10.3390/sym15010180

49. Elsadig M., Ibrahim A. O., Basheer S., et al. Intelligent Deep Machine Learning Cyber Phishing URL Detection Based on BERT Features Extraction. *Electronics*. 2022. Vol. 11, Iss. 22, Article 3647. DOI: 10.3390/electronics11223647

50. Simple Mail Transfer Protocol (SMTP). URL: https://www.geeksforgeeks.org/simple-mail-transfer-protocol-smtp/ (дата звернення 14.09.2024)

51. Креневич А.П. Алгоритми і структури даних. Підручник. – К.: ВПЦ "Київський Університет", 2021. – 200 с.

52. Phishing corpus. URL: https://academictorrents.com/details/a77cda9a9d89a60dbdfbe581adf6e2df9197995a (дата звернення 17.09.2024)

53. Apache SpamAssassin. URL: https://spamassassin.apache.org/ (дата звернення 17.09.2024)

54. Shakya A. K., Pillai G., Chakrabarty S. Reinforcement learning algorithms: A brief survey. *Expert Systems with Applications*. 2023. Vol. 231. DOI: 10.1016/j.eswa.2021.116468

55. Padakandla S., J P. K., Ganguly S., Bhatnagar S. Data Efficient Safe Reinforcement Learning. *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic*. 2022. PP. 1167–1172. DOI: 10.1109/SMC53654.2022.9945313

56. Liang J. Confusion Matrix: Machine Learning. *POGIL Activity Clearinghouse*. 2022. Vol. 3, Iss. 4.

57. Heydarian M., Doyle T. E., Samavi R. MLCM: Multi-Label Confusion Matrix. *IEEE Access*. 2022. Vol. 10, PP. 19083–19095. DOI: 10.1109/ACCESS.2022.3151048

58. ROC (Receiver Operating Characteristic) and EER (Equal Error Rate). URL: https://jimmy-shen.medium.com/roc-receiver-operating-characteristic-and-eer-equal-error-rate-ac5a576fae38 (дата звернення 24.09.2024)

59. СОУ 207.01:2017. Текстові документи. Загальні вимоги. Хмельницький: ХНУ, 2017. 46 с. URL: https://msn.khnu.km.ua/pluginfile.php/466522/mod_resource/content/1/132_C%20T%20A%20H%20Д%20A%20P%20T%20чист%20.pdf

60. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання. [Чинний від 2016-07-1]. Київ, 2016. 20 с. (Державна наукова установа ―Книжкова палата України імені Івана Федорова).

ДОДАТОК А. СПИСОК ПРАЦЬ

Technical sciences *ISSN 2307-5732*

DOI 10.31891/2307-5732-2024-341-5-73
УДК 004.056

PETLIAK NATALIIA
Khmelnytskyi National University
https://orcid.org/0000-0001-5971-4428
e-mail: npetlyak@khmnu.edu.ua
BEZKOROVALNYI YAROSLAV
Khmelnytskyi National University
e-mail: bezkorovalnyiiao@khmnu.edu.ua
KUPCHYK NATALIIA
Khmelnytskyi National University
e-mail: nataliiakupchyk@khmnu.edu.ua

ANALYSIS OF MODERN METHODS OF DETECTION OF PHISHING E-MAILS

Phishing attacks are one of the common threats to modern cyber security. The most common method fraudsters use to send fake messages to collect data is phishing emails. However, with the development of technology and artificial intelligence, the number and complexity of phishing attacks are increasing, making detecting them difficult. The article discusses traditional and modern methods of combating phishing, particularly blocklists and signature methods, and the latest machine and deep learning approaches. The analysis of the latest research made it possible to develop a generalised algorithm (fig. 2) for the implementation of the phishing email detection system, which consists of the following steps: data collection, data pre-processing, feature selection, modelling, email classification, model updating, blocking and notification/ Machine learning makes it possible to analyse large volumes of data and detect hidden patterns, which makes these methods effective for automatically blocking phishing emails. Convolutional and recurrent neural networks are also used to analyse the text of phishing messages at the level of words and phrases. Special attention is paid to developing natural language processing methods that help better understand the context of letters and detect anomalies. Deep models allow for extracting valuable features without pre-processing the data, making them practical for detecting new attacks. The implementation of machine and deep learning methods significantly increases the effectiveness of detecting phishing emails. However, further research is needed to improve and realise the models' full potential. It is necessary to create models that can independently adapt to new threats without manual intervention, analysing new patterns and strategies of attackers. This will ensure a more effective fight against phishing threats in the rapidly changing digital environment.

Keywords: phishing, social engineering, Large Language Model, machine learning.

ПЕТЛЯК НАТАЛІЯ
Хмельницький національний університет
БЕЗКОРОВАЛЬНИЙ ЯРОСЛАВ
Хмельницький національний університет
КУПЧИК НАТАЛІЯ
Хмельницький національний університет

АНАЛІЗ СУЧАСНИХ МЕТОДІВ ВИЯВЛЕННЯ ФІШИНГОВИХ ЕЛЕКТРОННИХ ЛИСТІВ

Фішингові атаки є однією з поширених загроз сучасній кібербезпеці. Найпоширеніший метод, який використовують шахраї для надсилання фальшивих повідомлень для збору даних, – це фішингові електронні листи. Однак із розвитком технологій і штучного інтелекту кількість і складність фішингових атак зростає, що ускладнює їх виявлення. У роботі наведено аналіз фішингових атак, етапи їх еволюції та основні методи захисту. Також розглянуто традиційні та сучасні методи боротьби з фішингом, зокрема списки блоків і сигнатурні методиу, алгоритми машинного та глибокого навчання. Машинне навчання дає змогу аналізувати великі обсяги даних і виявляти приховані шаблони, що робить метод ефективними для автоматичного блокування фішингових електронних листів. Згорткові та рекурентні нейронні мережі також використовуються для аналізу тексту фішингових повідомлень на рівні слів і фраз. Особлива увага приділяється розробці методів обробки природної мови, які допомагають краще розуміти контекст літер і виявляти аномалії. Глибокі моделі дозволяють отримувати цінні функції без попередньої обробки даних, що робить їх практичними для виявлення нових атак. Проаналізовано основні недоліки та перспективу подальших досліджень у контексті кіберзахисту.

Ключові слова: фішинг, соціальна інженерія, Large Language Model, машинне навчання.

Introduction

Phishing attacks are one of the common cybersecurity threats in today's digital world. They attack individual users, large organisations, and governments, making them a universal problem. The primary goal of such attacks is to obtain sensitive information such as passwords, credit card numbers, or personal information by impersonating the attacker as a trusted source. The nature of phishing attacks is constantly evolving, and their complexity is growing, which creates additional challenges for security systems and prompts the development of new, more sophisticated protection methods. Among the widespread forms, it is possible to single out phishing attacks via e-mail [1-2]. This method involves scammers sending fake emails that look legitimate to trick the victim into providing sensitive information or clicking on a malicious link. Along with this, other forms are common, such as spear phishing (targeted attacks on specific individuals or organisations), vishing (phishing via phone calls) and smishing (phishing via SMS).

With the development of digital technologies and Internet services, the number and complexity of phishing attacks have increased dramatically [3-4]. And the remote working mode has increased the dependence on e-mail and

other online services. This has created additional opportunities for attackers who have adapted their methods to attack less secure networks and users who need more cybersecurity training.

Phishing is a form of social engineering in which criminals manipulate users to force them to reveal confidential information [5-6]. According to the Anti-Phishing Working Group (APWG) report, phishing attacks increased significantly in 2023, reaching more than 1.2 million in the second quarter. The main target of phishing attacks remains the financial sector, which accounts for more than 23% of all cases [7]. Other vital industries such as healthcare, e-commerce and education face this threat. The problem is exacerbated by social engineering, which is often the first step in more sophisticated cybercrimes, such as infrastructure attacks or malware distribution.

Especially dangerous is the emergence of so-called "black" models of large language models that automate the creation of phishing messages. These models generate high-quality, personalised messages that are difficult to distinguish from legitimate emails. Criminals use them to create campaigns of mass phishing attacks, which significantly complicates the process of their detection and blocking.

Detecting phishing attacks is essential for keeping users and organisations safe. Classic anti-phishing methods include blocklists and signature methods. Blocklists allow you to capture suspicious domains or IP addresses used in previous attacks. However, this method has limitations due to the dynamic nature of phishing attacks. Fraudsters are constantly creating new URLs or changing minor details to bypass blocklists. The signature-based method also has drawbacks, as successful detection requires prior knowledge of attack patterns, and phishing often has new variations.

Traditional approaches are replaced by machine learning and deep learning methods, which demonstrate significant effectiveness in detecting phishing attacks. Machine learning algorithms can analyse large volumes of data, identify hidden patterns, and make predictions based on this data. Such methods allow the modelling of complex defence systems that can automatically adapt to new attacks, minimising human involvement. For example, machine learning algorithms can be trained on large datasets of phishing emails, identifying essential characteristics for their identification. This allows you to create effective systems for automatically blocking phishing messages before they are delivered to the user.

Unlike machine learning, deep learning techniques such as convolutional neural networks or recurrent neural networks can independently extract useful features from raw data without needing to pre-select features. This makes them practical for phishing detection and text classification tasks. Deep neural networks have demonstrated performance in various natural language processing tasks, including sentiment analysis, text categorisation, and message classification. However, these methods have several disadvantages, including the requirements for significant computing resources and difficulty explaining the decisions made.

Classification of methods for detecting phishing emails

Different methods of detecting phishing e-mails (fig. 1) have been developed to combat these threats, and each of them is based on different theoretical aspects and approaches that allow not only the identification of the danger but also the protection of the system from potential attacks.

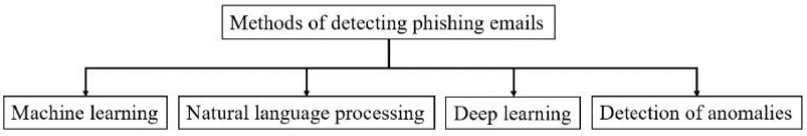

Fig. 1. Methods of detecting phishing emails

Machine learning is one of the main techniques widely used to detect phishing emails. The theoretical basis of machine learning is that the system can learn from previously collected data and model the behaviour of phishing attacks, using this data to predict new threats. Algorithms such as logistic regression, support vector machines, random forests and neural networks are used. Logistic regression is a classic two-class classification approach that predicts the probability that a given email is phishing. Support vector machines determine the optimal boundaries between classes, ensuring high accuracy in classifying phishing emails. Random Forest is an ensemble learning method that creates multiple tree-based solutions that increase robustness to variations in the data. Neural networks make it possible to effectively work with sequential text data, revealing hidden dependencies between words in the text of phishing emails. The main advantage of machine learning is the ability to automatically improve its results with new data, which allows the creation of dynamic detection systems that adapt to new types of attacks.

Natural language processing (NLP) is one of the techniques for detecting phishing emails because the primary information is in the text. The theoretical basis of NLP is to transform textual data into formats suitable for analysis and to detect potentially harmful patterns in the content of emails. NLP steps include tokenisation, stemming, and vectorisation. During tokenisation, the text is split into separate words or phrases. Stemming allows you to simplify words to their base form to improve text analysis. Text vectorisation using Bag of Words or Term Frequency-Inverse Document Frequency methods will enable you to convert text into numerical vectors for machine analysis. Modern NLP models, such as Bidirectional Encoder Representations from Transformers and Generative Pre-trained Transformer, provide contextual embedding of words, which allows the detection of more complex semantic relationships and phrases characteristic of phishing attacks. Detection of anomalies in the text can also be used to identify elements that are not characteristic of routine correspondence, which may indicate phishing.

Deep learning allows you to create robust models to detect complex phishing patterns using large amounts of

data. Deep learning models will enable the recognition of text and understanding of its context by studying interactions between words and their positions in the text. Fundamental in this context are models that allow embedding words' semantic and contextual meanings into vectors, making it possible to analyse individual words and their relationships in the text. With the ability to process large volumes of data and discover hidden connections, deep learning can help identify new and previously unknown phishing schemes.

Approaches based on detecting anomalies assume that phishing emails differ from ordinary emails by specific characteristics. Using statistical methods or machine learning algorithms allows you to analyse the typical behaviour of postal communications and determine deviations from this behaviour. For example, distance-based methods such as K-nearest neighbours help measure the similarity between a new email and already-known phishing patterns. Density-based methods, in turn, allow the detection of groups of anomalous emails that may indicate a massive phishing attack. Methods based on reconstruction (for example, autoencoders) study standard email patterns and detect those that differ from them.

One of the key theoretical aspects is the development of functions used to train models. Detecting phishing emails requires picking suitable email characteristics, such as text content, metadata, sender information, URLs, and behavioural data. Analysing textual content helps identify suspicious patterns or words typical of phishing messages. Analysis of metadata and email headers allows you to check the time of sending/receiving the email and the history of interactions with the sender, which can help detect suspicious activity. Analysing URLs in the text and studying the reputation of domains allows you to detect fake or malicious links.

Various theoretical approaches to detecting phishing e-mails allow the creation of sophisticated and effective systems to combat this threat. Each of the approaches has its advantages and limitations. Still, their combination will enable you to achieve the best results in detecting phishing and ensuring cyber security at the appropriate level.

Analysis of the latest research on methods of detecting phishing emails

The authors [8] suggest using machine learning models to classify phishing e-mails, dividing them into those generated by humans and by the Large Language Model (LLM). The research collected a dataset of thousands of human-generated phishing emails and thousands more emails created using WormGPT. Several machine learning models, such as neural networks, SVM, logistic regression, and others, have been used. However, if attackers begin to actively change the styles of creating phishing emails or use other LLMs to generate attacks, the effectiveness of the models may decrease. Since phishing attack tactics constantly change, the model will need regular updates and retraining to remain effective. The models have yet to be tested in open real-world conditions or conflict situations where attackers can actively counter defence mechanisms.

The study [9] presents an approach to detecting phishing e-mails using a deep learning model using an advanced convolutional neural network. The proposed model analyses phishing emails at multiple levels, focusing on the email header, body, character, and word levels. The model uses an attention mechanism that allows you to prioritise more critical information in the letter's structure. The study highlights the limitations of traditional phishing detection methods, such as blocklist mechanisms and classical machine learning algorithms, which require manual development of features and cannot adapt to the specifics of phishing emails. The model was tested on an unbalanced data set representing real-world proportions of phishing and legitimate emails. The model's results demonstrate its ability to recognise phishing emails more effectively than previous methods. A system architecture, including a data flow diagram and an entity-relationship model, is developed to represent the e-mail discovery and management process. The authors acknowledge the need for further improvement, particularly in cases where phishing emails do not contain critical headers.

The research in [10] involves developing and implementing a new model for detecting phishing e-mails based on recurrent convolutional neural networks using multi-level vectors and an attention mechanism. The study's uniqueness is that it analyses the structure of e-mails and considers both the header and the body of the letter at the level of characters and words. This allows for more accurate detection of phishing attacks, especially in the part of the email that most often contains fraudulent elements. An unbalanced database containing realistic proportions of phishing and legitimate emails was used to evaluate the proposed model's effectiveness. In addition, the study analyses existing technologies such as sender and link blocklists but notes that their effectiveness depends on the relevance and completeness of such lists. The authors highlight the growing threat of phishing attacks and emphasise the importance of automated tools based on machine learning and neural networks to counter these threats. The proposed model with multilevel vectors and an attention mechanism requires significant computing power for learning and working in real-time.

The work [11] systematically analyses NLP and machine learning methods. The study compares machine learning methods, including the Naive Bayesian classifier and logistic regression, focusing on their accuracy, reliability, and performance. Considerable attention is paid to the importance of data preprocessing, mainly cleaning, tokenisation, and removal of stop words, improving text analysis quality. A unique approach is also the analysis of URL characteristics to detect phishing sites, such as link length, presence of memorable characters and anomalies in the domain structure. This study offers an in-depth comparative analysis of existing methods and points to ways to improve the accuracy and effectiveness of phishing protection. Pre-processing of the data, including tokenisation and removal of stop words, can lead to the loss of important information, which can reduce the accuracy of the models.

Article [12] describes developing a system that detects and blocks phishing e-mails in real time using machine learning. The system recognises the difference between legitimate and malicious emails by analysing various characteristics such as email content, headers and attachments. This approach protects users from phishing attacks and

prevents the spread of malware and other threats. For example, the system can detect phishing emails imitating bank communications by analysing suspicious elements such as the sender's email address or requests for personal information. In addition, it protects against malicious attachments such as viruses or ransomware, thereby protecting users' devices from compromise and the spread of threats. Thanks to the model's ability to adapt and constantly improve through analysing new data, the system remains effective against the latest phishing tactics.

The authors of [13] suggest improving the detection system of phishing e-mails using combined machine learning algorithms. The study presents an approach combining supervised and unsupervised machine learning algorithms to analyse email properties and user behaviour to detect subtle signs of phishing. The system analyses email content, sender reputation, and behaviour patterns. It uses advanced natural language processing and pattern recognition algorithms to evaluate email content, URLs, and information from QR codes. The integration of these methods allows to improve the accuracy of detection of phishing attacks and reduce the number of false positive results compared to existing detection techniques. The study also highlights the importance of adapting to new phishing tactics, such as dynamic content loading and URL obfuscation. It suggests a comprehensive approach that includes monitoring login activity, scanning QR codes and URLs, and in-depth email content analysis. This extensive method increases email security and provides a proactive approach to cybersecurity, making it a valuable tool for protecting against phishing attacks.

The authors of [14] created an interpreted AI-based platform for real-time detection of phishing emails, using the most extensive available public data sets to train models, significantly improving their generalisation ability. The paper proposes a methodology for combining several data sets to increase the effectiveness of detecting different phishing emails.

Research [15] aims to solve phishing attacks by developing a hybrid approach combining machine and deep learning methods. Using a genetic algorithm to optimise feature selection allows the model to identify better features, which increases its performance. The proposed approach was evaluated on a dataset of 1,173 records emphasising writing letters with Arabic content.

Research [16] focuses on increasing the effectiveness of detecting phishing e-mails using ensemble learning, which combines different machine learning methods. The proposed model demonstrates an increase in accuracy compared to traditional approaches to classification, thanks to the combination of several algorithms, which allows for better detection of details and patterns characteristic of phishing attacks. A stack method combines SVM, XGBoost and logistic regression to optimise the classification.

Algorithm for detecting phishing emails

The analysis of the latest research made it possible to develop a generalised algorithm (fig. 2) for the implementation of the phishing email detection system, which consists of the following steps: data collection, data pre-processing, feature selection, modelling, email classification, model updating, blocking and notification.

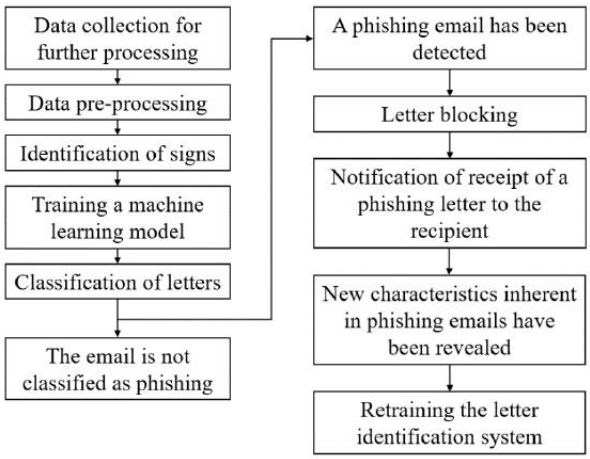

Figure 2. Algorithm for the implementation of the phishing email detection system

During the collection, the data necessary for identifying the phishing letter is selected; in particular, such data includes letter headers, attachments, the sender's address, the body of the letter, and other parameters. The next stage is data collection, which involves providing legitimate and phishing emails. Data pre-processing consists of removing unnecessary characters, HTML tags, and special characters that do not carry significant information, breaking the text into separate words or symbols, removing words that do not have meaningful information, and reducing words to their primary form to reduce data dimensionality. Identifying the signs of phishing emails begins with content analysis, which examines keywords and specific phrases frequently used by attackers. These can be urgent requests for action or threats to block the account. Next, the URLs are checked for anomalies such as length, suspicious characters, or obfuscation. The metadata of the letter is also analysed, particularly the headers, where the sender and the domain's reputation are evaluated. A critical component is the analysis of attachments for malware or suspicious file types. At the simulation

stage, appropriate machine learning models such as naive Bayesian classifier, logistic regression, recurrent neural networks, deep neural networks with an attention mechanism, or convolutional neural networks are selected, which allow text analysis at the character and word level. The model is trained on labelled data containing both phishing and legitimate emails. After training, its performance is evaluated by metrics such as accuracy, completeness, specificity, and F1-measure. When classifying new mail, the incoming mail undergoes the same preprocessing and feature extraction procedure. The model classifies the email as phishing or legitimate based on the features it obtains. As phishing attack tactics change, the model is tested for effectiveness and needs to be periodically retrained on new data to stay relevant, considering new writing styles or text generation methods. If the email is recognised as phishing, it is automatically blocked or moved to the "Spam" folder. The user receives a notification about a potential threat with recommendations for caution or further actions. For example, the algorithm can integrate with other systems to check senders and URLs using blocklists or reputation databases. In addition, the user's behaviour during interaction with the letter is analysed to detect potentially dangerous actions.

Conclusions

The implementation of machine and deep learning methods significantly increases the effectiveness of detecting phishing emails. However, further research is needed to improve and realise the models' full potential. It is necessary to create models that can independently adapt to new threats without manual intervention, analysing new patterns and strategies of attackers. This will ensure a more effective fight against phishing threats in the rapidly changing digital environment.

Література

1. Most Common Types of Phishing Attacks in 2024. URL: https://www.upguard.com/blog/types-of-phishing-attacks (дата звернення 2.09.2024)

2. Phishing Attacks: Statistics and Examples. URL: https://www.checkpoint.com/cyber-hub/threat-prevention/what-is-phishing (дата звернення 2.09.2024)

3. Тенденції у розвитку фішингу та протидія йому. URL: https://my-itspecialist.com/trends-in-phishing-development-and-countermeasures (дата звернення 3.09.2024)

4. Trends in Cyber Challenges and Solutions 2024. URL: https://www.h-x.technology/blog/trends-cyber-challenges-solutions-2024 (дата звернення 4.09.2024)

5. Avoiding Social Engineering and Phishing Attacks. URL: https://www.penncommunitybank.com/wp-content/uploads/2021/06/Avoiding-Social-Engineering-and-Phishing-Attacks.pdf (дата звернення 4.09.2024)

6. Understanding Social Engineering Tactics: 8 Attacks to Watch Out For. URL: https://www.tripwire.com/state-of-security/5-social-engineering-attacks-to-watch-out-for (дата звернення 9.09.2024)

7. Phishing activity trends report. URL: https://docs.apwg.org/reports/apwg_trends_report_q4_2023.pdf (дата звернення 9.09.2024)

8. Francesco Greco, Giuseppe Desolda, Andrea Esposito, Alessandro Carelli. David versus Goliath: Can Machine Learning Detect LLM-Generated Text? A Case Study in the Detection of Phishing Emails, ITASEC: The Italian Conference on CyberSecurity, Italy, Vol. 3731, 2024.

9. P. R. Uyyala. Phishing email detection using CNN, Journal of Engineering and Technology Management, Vol. 72, 2024, pp. 1046-1051.

10. S. A.Nabi, G. Srija, G. Madhuri, M. R. Reddy, C. Jashuva, Phishing email detection using improved RCNN with multilevel. International Journal For advanced research in science & technology, Vol. 13, No. 7, 2023, pp. 63–69.

11. J.Keerthika, A. Adisvara, S. Akash, B. Jayanesh, T. Arul Prakash, E-mail spam detection and phishing link detection using machine learning, Advances in Computational Intelligence in Materials Science, 2023, pp. 47-53, doi: 10.53759/acims/978-9914-9946-9-8_9.

12. Jude Osamor et al, Real-Time Detection of Phishing Emails Using XG Boost Machine Learning Technique, International Conference on Information Technologies and Smart Systems, India, 2024.

13. G. B. Sambare, S. B. Galande, S. Kale, P. Nehete, V. Jadhav & et al, Towards enhanced security: An improved approach to phishing email detection, Journal of Electrical Systems, Vol. 20, No. 2, 2024, pp. 2763-2772.

14. Abdulla Al-Subaiey, Mohammed Al-Thani, Naser Abdullah Alam, Kaniz Fatema Antora, Amith Khandakar, SM Ashfaq Uz Zaman, Novel interpretable and robust web-based AI platform for phishing email detection, Computers and Electrical Engineering, Vol. 120, Part A, 2024, doi: 10.1016/j.compeleceng.2024.109625.

15. A. Enhancing, Arabic Phishing Email Detection: A Hybrid Machine Learning Based on Genetic Algorithm Feature Selection, International Journal of Advanced Computer Science & Applications, Vol. 15, No. 8, 2024, p. 312, doi: 10.14569/ijacsa.2024.0150832

16. Anirudh S, P Radha Nishant, Sanjay Baitha, K Dinesh Kumar, An Ensemble Classification Model for Phishing Mail Detection, Procedia Computer Science, Vol. 233, 2024, pp. 970-978, doi: doi.org/10.1016/j.procs.2024.03.286.

References

1. Most Common Types of Phishing Attacks in 2024. URL: https://www.upguard.com/blog/types-of-phishing-attacks (application date 2.09.2024)

2. Phishing Attacks: Statistics and Examples. URL: https://www.checkpoint.com/cyber-hub/threat-prevention/what-is-phishing

(application date 2.09.2024)

3. Trends in the development of phishing and countering it. URL: https://my-itspecialist.com/trends-in-phishing-development-and-countermeasures (application date 3.09.2024)

4. Trends in Cyber Challenges and Solutions 2024. URL: https://www.h-x.technology/blog/trends-cyber-challenges-solutions-2024 (application date 4.09.2024)

5. Avoiding Social Engineering and Phishing Attacks. URL: https://www.penncommunitybank.com/wp-content/uploads/2021/06/Avoiding-Social-Engineering-and-Phishing-Attacks.pdf (application date 4.09.2024)

6. Understanding Social Engineering Tactics: 8 Attacks to Watch Out For. URL: https://www.tripwire.com/state-of-security/5-social-engineering-attacks-to-watch-out-for (application date 9.09.2024)

7. Phishing activity trends report. URL: https://docs.apwg.org/reports/apwg_trends_report_q4_2023.pdf (application date 9.09.2024)

8. Francesco Greco, Giuseppe Desolda, Andrea Esposito, Alessandro Carelli. David versus Goliath: Can Machine Learning Detect LLM-Generated Text? A Case Study in the Detection of Phishing Emails, ITASEC: The Italian Conference on CyberSecurity, Italy, Vol. 3731, 2024.

9. P. R. Uyyala. Phishing email detection using CNN, Journal of Engineering and Technology Management, Vol. 72, 2024, pp. 1046-1051.

10. S. A.Nabi, G. Srija, G. Madhuri, M. R. Reddy, C. Jashuva, Phishing email detection using improved RCNN with multilevel. International Journal For advanced research in science & technology, Vol. 13, No. 7, 2023, pp. 63–69.

11. J.Keerthika, A. Adisvara, S. Akash, B. Jayanesh, T. Arul Prakash, E-mail spam detection and phishing link detection using machine learning, Advances in Computational Intelligence in Materials Science, 2023, pp. 47-53, doi: 10.53759/acims/978-9914-9946-9-8_9.

12. Jude Osamor et al, Real-Time Detection of Phishing Emails Using XG Boost Machine Learning Technique, International Conference on Information Technologies and Smart Systems, India, 2024.

13. G. B. Sambare, S. B. Galande, S. Kale, P. Nehete, V. Jadhav & et al, Towards enhanced security: An improved approach to phishing email detection, Journal of Electrical Systems, Vol. 20, No. 2, 2024, pp. 2763-2772.

14. Abdulla Al-Subaiey, Mohammed Al-Thani, Naser Abdullah Alam, Kaniz Fatema Antora, Amith Khandakar, SM Ashfaq Uz Zaman, Novel interpretable and robust web-based AI platform for phishing email detection, Computers and Electrical Engineering, Vol. 120, Part A, 2024, doi: 10.1016/j.compeleceng.2024.109625.

15. A. Enhancing, Arabic Phishing Email Detection: A Hybrid Machine Learning Based on Genetic Algorithm Feature Selection, International Journal of Advanced Computer Science & Applications, Vol. 15, No. 8, 2024, p. 312, doi: 10.14569/ijacsa.2024.0150832

16. Anirudh S, P Radha Nishant, Sanjay Baitha, K Dinesh Kumar, An Ensemble Classification Model for Phishing Mail Detection, Procedia Computer Science, Vol. 233, 2024, pp. 970-978, doi: doi.org/10.1016/j.procs.2024.03.286.

Міністерство освіти і науки України
Хмельницький національний університет

ЗБІРНИК НАУКОВИХ ПРАЦЬ

за матеріалами XVI Всеукраїнської науково-практичної конференції
«Актуальні проблеми комп'ютерних наук АПКН-2024»

15-16 листопада 2024

Хмельницький 2024

УДК 004:37:001:62

Збірник наукових праць за матеріалами XVI Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2024». Хмельницький. 2024. 582с.

У збірнику наукових праць подані перспективні практичні розробки аспірантів, студентів та здобувачів в області сучасних інформаційних технологій. Розглянуто актуальні проблеми комп'ютерних наук, комп'ютерної інженерії, прикладної математики й інженерії програмного забезпечення, приведено ряд робіт по впровадженню інформаційних технологій у виробництво та управління. Висвітлено перспективні розробки сучасних систем пошуку, обробки й захисту інформації, медійних та комунікаційних системи.

УДК 004:37:001:62

Матеріали конференції відтворені з авторських оригіналів, друкуються в авторській редакції та наведені в алфавітному порядку прізвищ авторів. При макетуванні можливі незначні зміни компоновки контенту авторських оригіналів. Відповідальність за якість та зміст публікацій несе автор.

Участь у конференції та складові всіх її етапів (розгляд праць, перевірка на плагіат, макетування, публікація збірника наукових праць та видача сертифікатів) є безкоштовними для всіх учасників. Оргкомітет конференції висловлює подяку учасникам конференції та сподівається на подальшу співпрацю.

З питань проведення конференції та подальшого обміну інформацією звертатись на e-mail конференції: *apkt.khnu@gmail.com*

АКТУАЛЬНІ ПРОБЛЕМИ КОМП'ЮТЕРНИХ НАУК - 2024

XVI Всеукраїнська науково-практична конференція

Метою конференції є висвітлення актуальних проблем комп'ютерних наук, інформатики та інформаційних технологій.

Робочі мови конференції:

українська, англійська

СЕКЦІЇ КОНФЕРЕНЦІЇ:

1. Комп'ютерні науки та прикладні інформаційні технології.
2. Комп'ютерна інженерія та системи захисту інформації.
3. Математичне моделювання та інженерія програмного забезпечення
4. Телерадіокомунікації, медійні та комунікаційні системи.
5. Проблеми впровадження інформаційних технологій у виробництво та управління.

СПИСОК ОРГАНІЗАЦІЙ, ПРЕДСТАВНИКИ ЯКИХ БРАЛИ УЧАСТЬ У РОБОТІ КОНФЕРЕНЦІЇ:

Донбаська державна машинобудівна академія
Західноукраїнський національний університет
Національний технічний університет «Харківський політехнічний інститут»
Національний університет «Львівська політехніка»
Приватний заклад вищої освіти «ІТ СТЕП Університет»
Сумський державний університет
Харківський національний університет радіоелектроніки
Хмельницький національний університет
Хмельницький фаховий економіко-технологічний коледж УЕП

ОРГКОМІТЕТ КОНФЕРЕНЦІЇ:

Олег СИНЮК – голова оргкомітету, проректор Хмельницького національного університету з наукової роботи, доктор технічних наук, професор.

Тетяна ГОВОРУЩЕНКО – заступник голови оргкомітету, декан факультету інформаційних технологій Хмельницького національного університету, доктор технічних наук, професор.

Олександр БАРМАК – заступник голови оргкомітету, завідувач кафедри комп'ютерних наук Хмельницького національного університету, доктор технічних наук, професор.

Олег САВЕНКО – професор кафедри комп'ютерної інженерії та інформаційних систем Хмельницького національного університету, доктор технічних наук, професор

Олена ВИСОЦЬКА – доктор технічних наук, завідувач кафедри радіоелектронних та біомедичних комп'ютеризованих засобів і технологій Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут», професор

Євгеній ЛАВРОВ – доктор технічних наук, професор (Сумський державний університет)

Людмила ТІМОФЄЄВА – відповідальна за студентську науково-дослідну роботу ХНУ

Олександр МАЗУРЕЦЬ – секретар конференції, доцент кафедри комп'ютерних наук Хмельницького національного університету, к.т.н., доцент кафедри комп'ютерних наук ХНУ

Марина МОЛЧАНОВА – секретар конференції, викладач кафедри комп'ютерних наук Хмельницького національного університету

КОНТАКТНА ІНФОРМАЦІЯ:

e-mail для листування: *apkt.khnu@gmail.com*

ЗМІСТ

УДК 004.77

Безкоровальний Я.О., Навроцька К.В., Петляк Н.С.

Хмельницький національний університет

АНАЛІЗ СУЧАСНИХ МЕТОДІВ ВИЯВЛЕННЯ ФІШИНГОВИХ ЕЛЕКТРОННИХ ЛИСТІВ

У тезі проведено аналіз сучасних методів виявлення фішингових електронних листів, включно з традиційними методами на основі чорних списків та новітніми технологіями машинного навчання. Особлива увага приділяється глибокому навчанню та обробці природної мови для покращення виявлення прихованих шаблонів у текстах фішингових повідомлень.

The thesis analyzes modern methods of detecting phishing e-mails, including traditional methods based on blacklists and the latest machine learning technologies. Special attention is paid to deep learning and natural language processing to improve the detection of hidden patterns in the texts of phishing messages.

Фішингові атаки з кожним роком стають дедалі складнішими, використовуючи нові технології та методи для обману користувачів і проникнення в їхні системи [1]. Основна мета сучасних дослідників — це розробка нових методів виявлення фішингових електронних листів, які можуть забезпечити ефективний захист у динамічних умовах [2-3].

Метою роботи є аналіз сучасних підходів до виявлення фішингових листів, включаючи традиційні методи, машинне навчання та глибоке навчання.

Традиційні методи виявлення фішингових електронних листів включають чорні списки (blocklists) і сигнатурні методи. Чорні списки працюють на основі заборони доступу до відомих шкідливих доменів і IP-адрес. Основною перевагою цього підходу є його простота та швидкість впровадження. Якщо адреса або домен раніше використовувалася для фішингової атаки, їх можна заблокувати, не допускаючи повторних атак з цього джерела. Однак цей метод має суттєві недоліки. Фішери швидко адаптуються, змінюючи домени або використовуючи нові IP-адреси, що дозволяє їм обійти блокування. До того ж, метод чорних списків не може ефективно реагувати на нові фішингові кампанії, що використовують ще не заблоковані ресурси.

Сигнатурні методи, які спираються на розпізнавання певних шаблонів або «сигнатур» у тексті повідомлень, також довели свою ефективність у виявленні фішингових атак. Наприклад, сигнатури можуть включати певні фрази, URL-адреси

або структури листів, які часто використовуються у фішингових атаках. Проте цей підхід вимагає постійного оновлення бази даних сигнатур для врахування нових варіантів фішингових атак. Це робить сигнатурні методи менш адаптивними до динамічно змінюваних загроз.

Машинне навчання стало революційним підходом до виявлення фішингових атак. Завдяки здатності аналізувати великі обсяги даних і виявляти приховані патерни, цей метод дозволяє ефективно класифікувати електронні листи на фішингові та легітимні. Основна перевага машинного навчання полягає в тому, що система може навчатися на історичних даних і поліпшувати свою точність у міру отримання нової інформації. Серед поширених алгоритмів машинного навчання, які використовуються для виявлення фішингових електронних листів, можна виділити: логістичну регресію для бінарної класифікації, яка дозволяє оцінити ймовірність того, що конкретний електронний лист є фішинговим; метод опорних векторів знаходить оптимальні межі між класами (фішингові та легітимні листи), забезпечуючи високу точність класифікації; випадковий ліс використовує кілька дерев рішень для підвищення надійності й стійкості до змін у даних; нейронні мережі використовуються для обробки послідовних текстових даних і виявлення прихованих залежностей між словами в тексті фішингових електронних листів.

Алгоритми машинного навчання можуть навчатися на великих наборах даних, що містять приклади фішингових листів. Це дозволяє створювати автоматизовані системи, які можуть виявляти нові загрози, навіть якщо вони дещо відрізняються від раніше відомих атак. Велика кількість характеристик, таких як текст листа, URL-адреси та метадані, можуть бути використані для навчання моделей і підвищення їхньої ефективності.

Глибоке навчання є більш просунутим підходом порівняно з класичним машинним навчанням. Завдяки використанню багатошарових нейронних мереж, глибоке навчання дозволяє моделювати складні залежності між різними характеристиками електронного листа. Зокрема, згорткові нейронні мережі (CNN) та рекурентні нейронні мережі (RNN) дозволяють аналізувати текст фішингових повідомлень на рівні слів і фраз, що робить їх ефективними для завдань класифікації тексту.

CNN зазвичай використовуються для аналізу зображень, однак їх можна адаптувати для роботи з текстами. CNN можуть виявляти приховані шаблони у фішингових повідомленнях, обробляючи текстові фрагменти подібно до зображень.

RNN особливо корисні для обробки послідовних даних, таких як текст. RNN можуть враховувати попередні слова в реченні при аналізі наступних, що дозволяє їм краще зрозуміти контекст повідомлення. Наприклад, Long Short-Term

Memory (LSTM) є варіантом RNN, що використовується для довготривалої пам'яті та виявлення довготривалих залежностей у тексті.

Однією з переваг глибокого навчання є те, що ці моделі можуть самостійно виділяти ознаки з «сирих» даних, без необхідності попередньої обробки або вибору ознак вручну. Це робить їх особливо ефективними для виявлення нових, ще невідомих типів фішингових атак, які раніше не зустрічалися в навчальних наборах даних. Також глибокі моделі можуть одночасно обробляти кілька рівнів текстової інформації, аналізуючи як загальні характеристики, так і специфічні фрази або структури листа.

Обробка природної мови (NLP) є важливою складовою сучасних методів виявлення фішингових атак, оскільки більшість інформації у фішингових листах представлена у вигляді тексту. NLP дозволяє перетворювати текстові дані у формат, придатний для аналізу машинами, а також виявляти потенційно шкідливі шаблони в контенті електронних листів.

Основні кроки обробки природної мови включають: розбиття тексту на окремі слова або фрази для полегшення аналізу; зведення слів до їх базової форми для спрощення обробки тексту; перетворення тексту у числові вектори, які можна використовувати для подальшого машинного аналізу.

Сучасні моделі обробки природної мови, такі як BERT (Bidirectional Encoder Representations from Transformers) та GPT (Generative Pre-trained Transformer), дозволяють створювати контекстні представлення слів і виявляти складні семантичні зв'язки між фразами у фішингових листах. Це значно підвищує точність виявлення фішингу, особливо у випадках, коли повідомлення виглядають як легітимні, але містять підозрілі або нехарактерні для звичайної кореспонденції фрази.

Методи виявлення аномалій передбачають, що фішингові електронні листи відрізняються від звичайних за певними характеристиками. Статистичні методи або алгоритми машинного навчання дозволяють аналізувати звичну поведінку електронної пошти та визначати відхилення від неї. Наприклад, методи на основі відстаней, такі як метод k-ближчих сусідів, допомагають вимірювати схожість між новим листом і вже відомими шаблонами фішингу.

Для досягнення максимальних результатів у виявленні фішингових листів часто використовують комбіновані підходи, що поєднують кілька методів машинного навчання, глибокого навчання та обробки природної мови. Це дозволяє створювати гнучкі системи, здатні адаптуватися до нових загроз та мінімізувати кількість хибнопозитивних результатів.

Проте, навіть новітні методи мають свої обмеження. Використання глибоких нейронних мереж потребує значних обчислювальних ресурсів, а моделі

машинного навчання повинні постійно оновлюватися для підтримки актуальності у зв'язку з появою нових видів атак. Це вимагає безперервного вдосконалення систем та їхньої адаптивності. Тому необхідно створювати більш гнучкі системи захисту, що здатні самостійно адаптуватися до нових типів фішингових атак без необхідності постійного втручання людини. Це може бути досягнуто через використання самонавчальних моделей глибокого навчання, які можуть автоматично покращувати свої алгоритми на основі нових даних. Використання кількох методів одночасно, таких як машинне навчання, глибоке навчання та аналіз аномалій, може покращити ефективність виявлення фішингових листів та зменшити кількість хибнопозитивних результатів. Такі системи можуть бути більш гнучкими й стійкими до нових типів атак. Подальший розвиток BERT та GPT дозволить покращити виявлення фішингових атак, особливо у випадках, коли шахраї використовують складні семантичні структури для обману користувачів.

Отже, використання сучасних методів машинного навчання та глибокого навчання є перспективним напрямком у боротьбі з фішинговими атаками. Проте для максимального ефекту необхідно продовжувати дослідження та впровадження новітніх технологій, спрямованих на автоматизацію та підвищення адаптивності систем кібербезпеки.

Перелік посилань

1. Most Common Types of Phishing Attacks in 2024. URL: https://www.upguard.com/blog/types-of-phishing-attacks
2. Phishing Attacks: Statistics and Examples. URL: https://www.checkpoint.com/cyber-hub/threat-prevention/what-is-phishing
3. Trends in Cyber Challenges and Solutions 2024. URL: https://www.h-x.technology/blog/trends-cyber-challenges-solutions-2024

АКТУАЛЬНІ ПРОБЛЕМИ КОМП'ЮТЕРНИХ НАУК 2024

ЗБІРНИК НАУКОВИХ ПРАЦЬ

Комп'ютерна верстка: **Олександр МАЗУРЕЦЬ**

Підписано до друку 21.11.2024.
Версія друку «APKN2024_CorpusPaper v3mod5 Finita».

E-mail: *apkt.khnu@gmail.com*
ХНУ. м. Хмельницький, вул. Інститутська,11.

Завідувачу кафедри кібербезпеки
к.т.н., доц. Кльоцу Ю.П.
Безкоровального Ярослава Олеговича
ПІБ здобувача вищої освіти

Студента ФІТ, 2 курсу, групи КБЗІм-23-1

ЗАЯВА

З правилами чинного Положення про систему забезпечення академічної доброчесності у Хмельницькому національному університеті, згідно з яким виявлення академічного плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту і застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність академічного плагіату оповіщений (а) та надаю свою згоду на обробку й збереження університетом моєї роботи в інституційному репозитарії Хмельницького національного університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-обчислювального комплексу StrikePlagiarism та/або програмнотехнічного засобу Anti-Plagiarism) і використання роботи для виявлення академічного плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення текстових збігів в роботах.

Робота надається для перевірки в електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

29.11.2024
дата

підпис

Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Безкоровальний Ярослав

Співавтор:

Назва: Метод ідентифікації фішингових атак в електронних листах

Науковий керівник: Касянчук М.М.

Підрозділ: Кафедра кібербезпеки

Коефіцієнт подібності 1:1.9%

Коефіцієнт подібності 2:0.2%

Мікропробіли: 0

Заміна букв: 0

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2024-12-01 13:15:16.0

Після аналізу Звіту подібності констатую наступне:

☐ **Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.**

☐ **Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.**

☐ **Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.**

Обгрунтування:

Дата експерт

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 1.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 10%**

ID: 152729 Назва: Метод ідентифікації фішингових атак в електронних листах Додано в БД: 2024-12-01 Автора: Безкоровальний Ярослав Керівники: Касянчук М.М. Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	90891	677	594 (1%)	7 (1%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КІБЕРБЕЗПЕКИ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод ідентифікації фішингових атак в електронних листах

Автор: Безкоровальний Ярослав Олегович

Спеціальність: 125 – Кібербезпека та захист інформації

Освітня програма:Кібербезпека та захист інформації

Науковий керівник:Михайло КАСЯНЧУК, докт. техн. наук, професор

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних). Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи.	
3	Виявлені запозичення не є плагіатом; але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
5	Інше:	

Підтвердження:

Оригінальність тексту роботи за результатами перевірки системою StrikePlagiarism складає 98,1%, оригінальність тексту роботи за результатами перевірки системою Anti-Plagiarism складає 99%.

Згідно з правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 24.09.2024, авторська робота, обсяг оригінального тексту у відсотках до загального обсягу матеріалу в якій складає 90-100%, визначається роботою з високою унікальністю тексту і допускається до захисту.

Виявлені модифікації стосуються математичних формул і не є порушенням академічної доброчесності.

Керівник роботи _____ Михайло КАСЯНЧУК

Гарант ОП _____ Віра ТІТОВА

Завідувач кафедри кібербезпеки _____ Юрій КЛЬОЦ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

освітнього ступеня «бакалавр»

Студент Безкоровальний Ярослав Олегович

Тема Метод ідентифікації фішингових атак в електронних листах

Спеціальність 125 – Кібербезпека та захист інформації

Обсяг кваліфікаційної роботи освітньо-кваліфікаційного рівня «бакалавр»:

кількість листів креслень – ; кількість сторінок записки 93 .

1. Короткий зміст роботи та прийнятих рішень У кваліфікаційній роботі була розроблено метод ідентифікації фішингових атак в електронних листах. Проаналізовано поняття фішингу та існуючі методи його виявлення. Проаналізовано параметри електронних листів, що використовуються для ідентифікації фішингу та виконано попередню обробку набору даних для тестування. Описано модель ідентифікації фішингових листів. Розроблено метод ідентифікації фішингових атак в електронних листах. Робота також включає оцінку ефективності запропонованого методу.

2. Висновок про відповідність кваліфікаційної роботи завданню У кваліфікаційній роботі було виконано поставлене завдання як у теоретичній, так і в практичній частині.

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі роботи наведено загальну характеристику поставленої задачі, визначено об'єкт, предмет та методи дослідження, а також сформульовано мету. У першому розділі систематизовано наявні підходи до визначення фішингу, охарактеризовано його життєвий цикл та особливості атак. У цьому розділі розглянуто основні методи виявлення фішингових атак, надано порівняльний аналіз їхніх переваг і недоліків. У другому розділі описано процес обробки вхідних даних, виділення ознак з електронних листів і формування наборів даних для машинного навчання, пояснено вибір характеристик, які найбільш інформативні для ідентифікації фішингових атак. Третій розділ присвячено розробці алгоритму та методу ідентифікації фішингових атак в електронних листах. Четвертий розділ зосереджується н оцінці ефективності прототипу системи.

4. Позитивні сторони роботи Кваліфікаційна робота має як наукову, так і практичну цінність. Запропонований метод ідентифікації фішингових атак в електронних листах здатен знизити рівень витоку даних шляхом інтеграції розробленого методу у системи кіберзахисту.

5. Негативні сторони роботи Бракує детального аналізу впливу різних параметрів на продуктивність алгоритму.

6. Оцінка графічного оформлення та пояснювальної записки роботи Пояснювальна записка відповідає нормам оформлення.

7. Відгук про роботу в цілому Кваліфікаційна робота заслуговує позитивної оцінки, оскільки весь матеріал роботи є структурованим, чітким та послідовним. Усі розділи роботи мають логічну послідовність, що сприяє зрозумінню викладеного матеріалу в рамках теми роботи.

8. Інші зауваження

9. Оцінка кваліфікаційної роботи Враховуючи всі позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що вона заслуговує оцінки «відмінно».

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи)
Мартинюк Валерій Володимирович,
завідувач кафедри автоматизації, комп'ютерно-інтегрованих технологій та робототехніки, доктор технічних наук, професор

«13» грудня 2024. (підпис)