

**Єрємкін Михайло Михайлович**

аспірант кафедри безпеки інформаційних технологій

Національного університету «Львівська політехніка», м. Львів, Україна

ORCID: 0009-0008-6067-2382

mykhailo.yeromkin.asp.2025@lpnu.ua

Семенюк Сергій Анатолійович

к.ф.-м.н., доцент кафедри безпеки інформаційних систем

Національного університету «Львівська політехніка», м. Львів, Україна

ORCID: 0000-0002-8143-5887

serhii.a.semenyuk@lpnu.ua

НАСКРІЗНІ МЕТОДИ ВИКОРИСТАННЯ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ ДЛЯ ПРОАКТИВНОЇ КІБЕРРОЗВІДКИ В ІОТ/EDGE-ІНФРАСТРУКТУРАХ

Анотація. Стрімкий розвиток мереж інтернету речей та розподілених периферійних обчислень на тлі паралельного ускладнення кіберзагроз створює умови, за яких традиційні реактивні системи безпеки не забезпечують адекватного захисту від цільових атак і загроз нульового дня. Метою статті є обґрунтування архітектури наскрізного фреймворку та методів використання великих мовних моделей в рамках аналізу кіберзагроз для переходу від традиційного реактивного захисту до проактивної і контекстно-залежної парадигми безпеки в розподілених та корпоративних ІТ-інфраструктурах.. У роботі запропоновано трирівневу архітектуру екосистеми, у межах якої функціонують окремі субкомпоненти-методи: контекстне виявлення аномалій мережевої та системної поведінки IoT/Edge-пристроїв; проактивне прогнозування наступного пакета та типу мережевої активності; виділення й класифікація індикаторів компрометації; атрибуція суб'єкта загрози та тактик, технік і процедур (TTP); пріоритизація вразливостей за метриками CVSS та EPSS; кореляція кібератак і кампаній. Для кожного з компонентів описано теоретичну модель, спосіб інтеграції LLM (навчання в контексті, тонке налаштування, RAG) та наведено емпіричні результати з проаналізованих джерел. Рівні запропонованої архітектури (стратегічний, синхронізаційний, оперативний рівні), об'єднують розглянуті субкомпоненти в єдиний цикл проактивної кіберрозвідки. Також для практичного впровадження визначено ключові обмеження, як-от ресурсомісткість LLM на периферійних пристроях, часову асинхронію між Edge- та Cloud-рівнями, ризик каскадного поширення помилок і відсутність єдиних семантичних стандартів для передачі ймовірнісних висновків моделей.

Ключові слова: інформаційна безпека; великі мовні моделі; кіберрозвідка; проактивна кібербезпека; IoT; Edge Computing; індикатори компрометації; атрибуція загроз; TTP; пріоритизація вразливостей; CVSS/EPSS; генерація з доповненням пошуком

ВСТУП

Постановка проблеми. У зв'язку зі стрімкою еволюцією та впровадженням мереж Інтернету речей (IoT), розподілених периферійних обчислень (Edge Computing) і паралельним розвитком пов'язаних кіберзагроз утворюються умови, за яких традиційні реактивні системи не здатні забезпечувати адекватний рівень захисту. Сучасні засоби безпеки, орієнтовані на фіксацію вже виявлених або добре відомих аномалій, залишають інфраструктуру вразливою до цільових атак і загроз нульового дня. Відповідно, виникає нагальна потреба в переході до проактивної парадигми безпеки, що спирається на випереджувальний аналіз кіберзагроз і прогнозування поведінки потенційних злоумисників.

Основним джерелом для проактивних підходів є неструктуровані звіти про загрози, обговорення на форумах, дописи СТІ-провайдерів. Ці дані формують великий обсяг інформації, що унеможливорює її ефективну обробку людиною. Класичні підходи до автоматизації СТІ на основі простих парсерів і регулярних виразів мають низьку контекстну стійкість, не здатні виявляти складні семантичні зв'язки і призводять до великої кількості хибнопозитивних результатів.



Інтеграція великих мовних моделей (LLM) відкриває нові напрями автоматизації інтелектуальних процесів у сфері інформаційної безпеки завдяки можливості обробки природної мови та логічних міркувань [1]. Однак статичний характер навчальних даних мовних моделей, високий рівень галюцинацій та апаратні обмеження кінцевих пристроїв IoT/Edge накладають суттєві обмеження.

Тому постає необхідність розробки та обґрунтування архітектурних рішень на основі великих мовних моделей, що об'єднують функціональні компоненти інтелектуального аналізу неструктурованих даних, динамічної інтеграції знань у реальному часі та проактивного моделювання поведінки мереж та пристроїв для нейтралізації загроз до настання деструктивної фази.

Аналіз останніх досліджень та публікацій. У дослідженні [2] розроблено розподілену архітектуру безпеки, де на кінцевих пристроях функціонують полегшені моделі машинного навчання, що виконують первинний збір і фільтрацію трафіку та логів. При виявленні нетипової активності локальний Edge-сервер передає дані за допомогою протоколу MQTT на центральний сервер з LLM, який залучається як інтелектуальне ядро для обробки аномалій. Модель використовує технологію навчання в контексті для динамічного оновлення знань про нові загрози та вразливості у реальному часі, що дозволяє оперативно надавати рішення для ізоляції скомпрометованих вузлів.

Автори [3] розробили систему прогнозування вторгнень на рівні IoT. Метод базується на інтеграції двох мовних моделей у цикл зворотного зв'язку. GPT-2 Small виконує функцію авторегресійного генератора, що передбачає ймовірні параметри наступних мережових пакетів на основі поточного контексту. DistilBERT, своєю чергою, виступає в ролі оцінювача і здійснює класифікацію пар мережових пакетів для перевірки семантичної та контекстної послідовності згенерованого трафіку. Прогнозовані пакети класифікуються нейромережею для виявлення ознак шкідливої активності. Такий підхід дозволяє виявляти шкідливий трафік з точністю 98%.

Стаття [4] розвиває ідею прогнозування трафіку шляхом побудови фреймворку BARTPredict на рівні Multi-Access Edge Computing. Модель BART Base у поєднанні з двоспрямованим енкодером та авторегресійним декодером ефективно аналізує мережову активність у певному часовому вимірі. Одна модель BART генерує майбутні мережові пакети, а інша адаптується для бінарної класифікації трафіку. Процес оцінки згенерованих пакетів покладається на спеціально налаштований модуль DistilBERT, який класифікує логічну узгодженість послідовності пакетів.

Дослідження [5] фокусується на розгортанні полегшених мовних моделей безпосередньо на периферійних пристроях. Модель BERT-Small продемонструвала точність виявлення аномалій трафіку на рівні 99,7%. При виявленні аномальних ознак вихідні дані мовної моделі передаються на Edge-компонент на базі дерева рішень, який ініціює контрзаходи (наприклад, ізоляцію пристрою, блокування мережового зв'язку, активацію CAPTCHA). В сукупності це формує автономну систему захисту.

Праця [6] оцінює ефективність використання великих мовних моделей для аналізу кіберзагроз на тіньових хакерських форумах. Система на базі OpenAI GPT-3.5-turbo в автоматизованому zero-shot режимі аналізує неструктуровані щоденні обговорення на веб-ресурсах. LLM аналізує семантичний контекст повідомлень і кодує метрику з 10 CTI-параметрів із середньою точністю 96,2%. Модель успішно вилучає дані про продаж уразливостей, можливість віддаленої експлуатації, торгівлю доступом до мереж та потенційні загрози для критичної інфраструктури чи організацій.

Автори у [7] розробили агентну систему CyLens, що підтримує повний життєвий цикл аналізу кіберзагроз. Для навчання моделей було використано методику Curriculum Training, що включала інтеграцію даних з понад 270 тисяч звітів та баз даних CVE/CWE. CyLens використовує каскадне міркування та шість функціональних NLP-модулів. Завдяки цьому система здатна визначати базову серйозність загроз за стандартом CVSS v3 з точністю 96,86% для відомих і 91,57% для загроз нульового дня з мінімальним показником галюцинацій, порівняно з комерційними загальними LLM.

Дослідження [8] представляє проактивну систему кібербезпеки на основі моделі GPT-4o з динамічними потоками CTI та технологією RAG. GPT-4o обмежена статичною датою зрізу знань і не володіє інформацією про нові загрози, тому фреймворк використовує автоматизацію Patrowl для постійного збору актуальних даних. Зібрані дані кодуються у високорозмірні вектори та зберігаються у векторній базі даних Milvus. За запитом система виконує пошук у базі даних і надає генератору GPT-4o свіжий контекст у реальному часі, що дозволяє ефективно оцінювати поточну ймовірність експлуатації вразливостей і генерувати рекомендації щодо протидії новим загрозам.

У роботі [9] проведено систематичну оцінку використання великих мовних моделей для виявлення індикаторів компрометації (IOC) безпосередньо в неструктурованому веб-контенті. Дослідники побудували автоматизовану систему, що збирає звіти, очищує їх і тестує шість мовних моделей класів Gemini, Qwen, Llama. Завдяки контекстному розумінню моделі класифікують, чи є наявні індикатори реальними джерелами загроз. Найкращі результати продемонструвала модель Gemini 1.5 Pro, яка досягла точності 95,8% з ідеальним показником Recall = 1.0 для реальних шкідливих індикаторів. Такий підхід

дозволяє автоматично вилучати актуальні IOC та превентивно вносити їх до корпоративних систем виявлення вторгнень.

Метою статті є обґрунтування архітектури наскрізного фреймворку та методів використання великих мовних моделей в рамках аналізу кіберзагроз для переходу від традиційного реактивного захисту до проактивної і контекстно-залежної парадигми безпеки в розподілених та корпоративних IT-інфраструктурах.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Архітектура наскрізного трирівневого фреймворку проактивної СТИ-кіберрозвідки.

Для заповнення прогалин між глибоким стратегічним аналізом загроз та оперативним реагуванням на периферійних пристроях запропоновано архітектуру наскрізного фреймворку проактивної кіберрозвідки на основі використання великих мовних моделей.

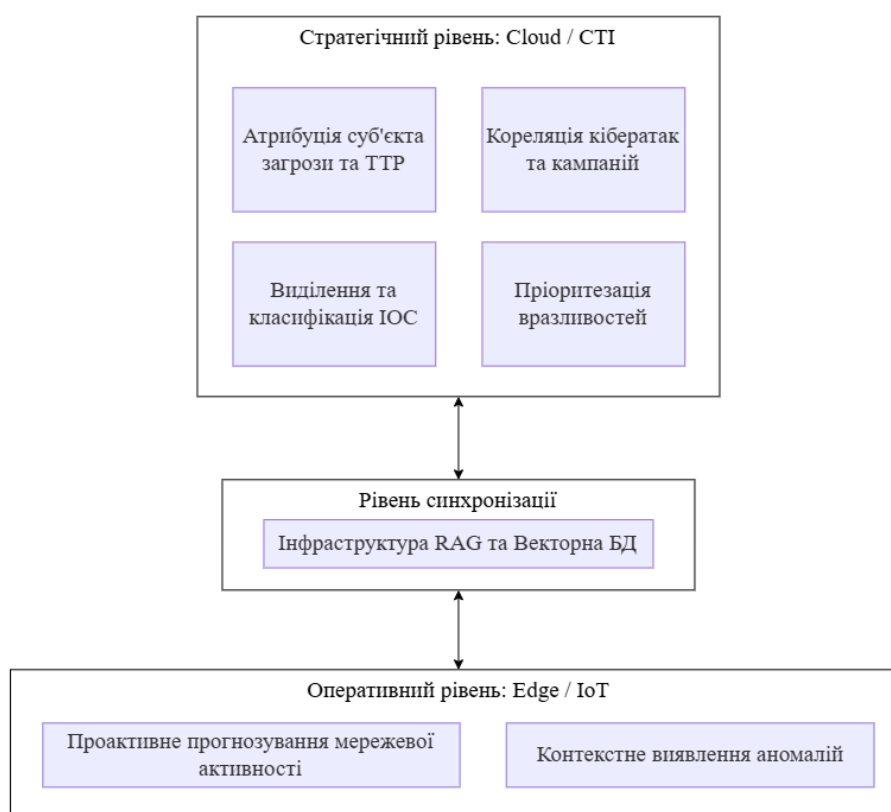


Рис. 1. Архітектурна схема наскрізного фреймворку проактивної кіберрозвідки на основі великих мовних моделей.

Запропонований фреймворк виступає в ролі об'єднувальної екосистеми, у якій його структурні субкомпоненти реалізують різні методи використання LLM для проактивної кіберрозвідки.

Подібна архітектурна модель об'єднує компоненти у три взаємопов'язані рівні. Стратегічний рівень здійснює централізовану ресурсомістку обробку неструктурованої СТИ-інформації з відкритих джерел. На цьому рівні методи виділення та класифікації IOC й семантичної атрибуції суб'єкта загрози та TTP збагачують оцінку критичності вразливостей, а також дозволяють будувати цілісну картину кіберкампаній. Рівень синхронізації виступає інтеграційним містком, що містить у собі векторні бази даних та архітектуру RAG. Він забезпечує передачу стиснутих та актуалізованих профілів загроз із хмарного середовища на периферійні вузли. Водночас, оперативний рівень забезпечує реагування в режимі реального часу безпосередньо поблизу джерел даних. Отримані з хмари профілі дозволяють адаптувати розрахунок інтегрального показника аномальності (1) для контекстного виявлення відхилень і коригувати параметри вихідного вектора (2) для проактивного прогнозування мережевої активності. Додатково, практичні аспекти структуризації СТИ-контексту для автоматизованої генерації офіційних



звітів та обміну даними про загрози через платформи типу MISP із залученням RAG-архітектури підтверджено у роботі [10].

Таким чином, кожен із представлених далі компонентів є не окремим ізольованим рішенням, а функціональним елементом у загальному циклі обробки даних.

Разом з тим, практична побудова такої об'єднаної системи зіштовхується з низкою критичних інтеграційних бар'єрів:

- 1) Часова асинхронія: Необхідність поєднання локального аналізу трафіку на Edge-рівні із ресурсомісткими та тривалими запитами до централізованих або хмарних LLM і RAG-систем.
- 2) Розрив між Edge- та Cloud-рівнями інфраструктури: Складність забезпечення неперервного й, головне, захищеного каналу синхронізації між обмеженими апаратно-периферійними пристроями та центральним «вектором знань».
- 3) Каскадне поширення помилок: Ризик, що хибне вилучення ІОС чи хибна атрибуція призведе до автоматичного блокування критичних вузлів на периферії та дестабілізації системи захисту.
- 4) Відсутність єдиних семантичних стандартів: Обмеженість традиційних СТІ-протоколів у передачі нечітких ймовірнісних висновків і контексту, що постачаються мовними моделями.

Незважаючи на вищевказані проблеми, саме створення розгалужених мультиагентних платформ є перспективним вектором розвитку для досягнення максимального синергетичного ефекту у проактивному захисті.

Контекстне виявлення аномалій мережевої та системної поведінки IoT/Edge-пристроїв на основі LLM

На оперативному рівні запропонованого фреймворку ключову роль відіграє субкомпонент контекстного виявлення аномальної поведінки пристроїв. Традиційні підходи, засновані на зіставленні поодиноких подій із відомими сигнатурами атак, демонструють обмежену ефективність проти нових загроз. Натомість цей підхід моделює активність пристрою як безперервну часову послідовність подій, де аномалія визначається як стійке відхилення від сформованого профілю нормальної роботи. Для цього здійснюється синхронний збір і первинне опрацювання мережевого трафіку та логів системних журналів. Системні логи нормалізуються та трансформуються в структуровані записи про події з фіксацією часу, типу події, джерела, ідентифікатора процесу, рівня важливості та її змісту. При цьому ключовим фактором для аналізу моделлю є не стільки семантичний вміст окремого запису, скільки його часова локалізація відносно інших подій. Наприклад, поодинокий збір автентифікації є штатною подією, тоді як ланцюжок із серії помилок входу, за якими слідують запуск нового нетипового процесу, зміна системної конфігурації та встановлення вихідного мережевого з'єднання, свідчить про компрометацію системи.

На етапі попередньої обробки вхідні дані групуються у часові вікна, що відображають узагальнений стан пристрою протягом фіксованого періоду. Для кожного типу пристроїв на основі даних їхнього штатного функціонування створюється динамічний профіль нормальної поведінки, який охоплює типові мережеві протоколи, легітимні IP-адреси, інтенсивність трафіку, штатні процеси та статистичні характеристики подій. Великі мовні моделі залучаються не для спрощеної бінарної класифікації, а для глибокої контекстної оцінки поточного стану пристрою. Оцінювання здійснюється шляхом подання до LLM узагальненого часового вікна з урахуванням попереднього контексту, що дає змогу виявити складні поведінкові аномалії, які неможливо ідентифікувати за окремими ознаками. Наприклад, збільшення обсягів вихідного трафіку для мережевої камери може бути нормальним під час зміни режиму зйомки, проте його поєднання з появою невідомого зовнішнього хоста, нетипового протоколу та помилок у логах однозначно вказує на загрозу.

Для математичного представлення відхилень від нормального профілю формується інтегральний показник аномальності:

$$A = w_p A_p + w_l A_l + w_c A_c \quad (1)$$

де A_p відображає рівень аномальності мережевих пакетів, A_l — системних логів, а A_c є результатом контекстної кореляції обох джерел; вагові коефіцієнти w_p, w_l, w_c визначають чутливість системи до конкретного типу даних. Оскільки нормальна поведінка пристроїв не є статичною і може змінюватися залежно від часових інтервалів чи навантаження, LLM формує контекстно-залежне представлення, мінімізуючи хибнопозитивні спрацювання завдяки оцінці загального експлуатаційного контексту.

Розподіл обчислювального навантаження в IoT/Edge-інфраструктурі ґрунтується на ієрархічному принципі. На кінцевих пристроях виконуються збір, первинна фільтрація та агрегація даних у часові вікна,



що усуває потребу в обробці всього сирого трафіку в реальному часі й знижує навантаження на канали зв'язку. Складніший контекстний аналіз із використанням ресурсомістких LLM переноситься на локальні Edge-сервери або централізовану хмару. Результатом функціонування цього компоненту є структурований звіт про аномалію, що містить ідентифікатор пристрою, часові межі інциденту, перелік відхилених параметрів, записи аномальних подій у логах, розрахований рівень аномальності та детальний опис контексту подій. Це перетворює сирі потоки даних на структурований поведінковий контекст, що передається далі на вищій рівні фреймворку для подальшої атрибуції та пріоритизації загроз.

Ефективність даного субкомпоненту контекстного виявлення аномалій у розподілених IoT/Edge-середовищах підтверджуються результатами емпіричних досліджень.

Перший варіант реалізації децентралізованої архітектури було апробовано у дослідженні [2], де як кінцеві Edge-пристрої використовувалися два мікрокомп'ютери Raspberry Pi та один Android-смартфон. На кінцевих пристроях було розгорнуто полегшені моделі машинного навчання, конвертовані у формат TensorFlow Lite, для проведення первинної фільтрації та локального аналізу мережевої активності. Зв'язок між Edge-пристроями та виділеним локальним Edge-сервером здійснювався за допомогою полегшеного протоколу MQTT. У разі виявлення нетипової поведінки або раніше невідомих аномалій Edge-сервер через REST API звертався до центрального LLM-сервера на базі OpenAI GPT API, використовуючи механізм навчання в контексті без потреби в повному перенавчанні моделей. Така архітектура дозволила не лише звести затримку передачі даних до мінімуму, а й забезпечити автономне реагування, зокрема автоматичний карантин та ізоляцію скомпрометованих вузлів від решти мережі для локалізації кіберзагрози.

Інший варіант практичного втілення контекстного виявлення аномалій базувався на тонкому налаштуванні полегшених мовних моделей на спеціалізованих IoT-датасетах IoT-23 та ToN IoT. У межах дослідження [5] в симульованому за допомогою Docker-контейнерів IoT-середовищі порівнювалися три варіанти моделей: BERT-Small, BERT-Mini та TinyBERT. Оцінювання моделей здійснювалося на тестовій вибірці (розподіл даних: 60% — навчання, 20% — валідація, 20% — тестування) з розрахунком точності класифікації за 21 категорією загроз. Емпіричні результати продемонстрували надвисоку ефективність моделей:

- Модель BERT-Small продемонструвала найкращі показники, досягнувши F1-метрики на рівні 99,75%. Модель показала найнижчі показники втрат під час навчання (0.0089) та валідації (0.0086), забезпечуючи швидкість обробки на рівні 287.82 запитів на секунду.
- Модель BERT-Mini показала аналогічну точність та F1-міру на рівні 99,72% за дещо вищою швидкістю інференсу — 293,56 запитів на секунду.
- Модель TinyBERT забезпечила точність 99,73% зі швидкістю 232,35 запитів на секунду.

Для автоматизації запобігання загрозам, виявленим за допомогою зазначених LLM, аномалії передавалися на Edge-компонент, побудований на базі моделі дерева рішень. Цей компонент у режимі реального часу ініціював контекстно-залежні контрзаходи: блокування скомпрометованих IP-адрес, обмеження швидкості трафіку, ізоляцію пристроїв або активацію CAPTCHA при виявленні масштабних ботнет-атак (наприклад, Okiru чи Mirai) і DDoS-активності, що підтвердило можливість побудови повністю автономних самозахисних IoT-екосистем.

Проактивне прогнозування наступного пакета та типу мережевої активності на основі LLM

Для забезпечення проактивного захисту на оперативному рівні розміщено субкомпонент проактивного прогнозування майбутнього стану мережевої активності. Якщо модуль виявлення аномалій орієнтований на фіксацію вже наявних відхилень, то даний субкомпонент здійснює перехід до превентивного передбачення наступних подій. Він моделює мережевий потік даних як безперервну часову послідовність і намагається завчасно передбачити її розвиток. Перед обробкою мовною моделлю сирій мережевий трафік у форматі PCAP-файлів підлягає попередній обробці: з пакетів екстрагуються ключові структуровані ознаки (IP-адреси джерела й призначення, порти, протоколи, тривалість з'єднання, обсяги переданих байтів та стан сесії), що дає змогу сформувати компактні вектори параметрів.

Кожен мережевий пакет представляється як структурований вектор подій, а потік трафіку записується у вигляді часової послідовності $P = \{p_1, p_2, \dots, p_t\}$, де p_i позначає вектор параметрів відповідного пакета. Для подачі до великої мовної моделі числові та категоріальні значення параметрів токенизуються за допомогою спеціальних словників і супроводжуються позиційним кодуванням. Це перетворює трафік на послідовність, що дає змогу вивчати закономірності переходів між мережевими станами.

В основі роботи компоненту лежить формування контекстного вікна поточної активності $C_t =$



$\{p_{t-k+1}, \dots, p_t\}$ яке охоплює фіксовану кількість попередніх мережових подій. Використовуючи цей контекст, авторегресійна модель максимізує ймовірність прогнозування характеристик наступного пакета. Цей підхід дозволяє виявляти довготривалі залежності між віддаленими елементами послідовності (наприклад, початком сесії, службовими пакетами керування та зміною мережевого стану).

Для підвищення надійності процесу загальне аналітичне завдання розділено на два рівні: низькорівневу генерацію параметрів наступного пакета p_{t+1} та його подальшу семантичну інтерпретацію. На другому рівні на основі поєднання контексту та передбаченого пакета здійснюється класифікація типу мережової діяльності $T_{t+1} = F(C_t, p_{t+1})$, де T_{t+1} класифікує майбутню подію як легітимну трансляцію даних або конкретний вид кібератаки.

Оцінювання достовірності згенерованого пакета проводиться за допомогою двоспрямованої моделі, яка аналізує пару «поточний пакет — прогнозований пакет» як єдине ціле для визначення контекстної узгодженості послідовності. Остаточний результат даного субкомпоненту представляється у вигляді структурованого набору:

$$R_{t+1} = p_{t+1}, P_{t+1}, T_{t+1}, C_{t+1}, \quad (2)$$

де p_{t+1} позначає вектор ознак прогнозованого пакета, P_{t+1} — ймовірність прогнозу, T_{t+1} — очікуваний тип активності, а C_{t+1} — оцінка контекстної узгодженості.

Для мінімізації чутливості до поодиноких похибок можна генерувати кілька альтернативних прогнозів $p_{t+1}^1, p_{t+1}^2, \dots$ із відповідними ймовірностями, а також будувати короткострокові траєкторії руху трафіку вперед на кілька кроків $p_t \rightarrow p_{t+1} \rightarrow p_{t+2} \rightarrow p_{t+3}$, глибина яких обмежується для запобігання накопиченню помилок. У межах IoT/Edge-інфраструктур цей компонент реалізується за схемою розподілу обчислень: малопотужні кінцеві пристрої виконують лише збір і базову структурування пакетів, тоді як генерація та аналіз за допомогою ресурсомістких LLM переносяться на виділені Edge/MEC-сервери. Отриманий прогноз виступає випереджальною ознакою для всієї СПІ-екосистеми, уможливаючи упередження загроз до фактичного початку деструктивної фази атаки.

В дослідженні [3] практичну життєздатність авторегресійної схеми прогнозування підтверджує система на базі IoT-дасету CICIoT2023 із відбором 26 ключових параметрів трафіку. Модель LLM GPT-2 Small використовувалася для генерації параметрів наступного пакета, водночас двоспрямований класифікатор DistilBERT оцінював узгодженість послідовності пакетів з точністю 93,40%, а LSTM-нейромережа класифікувала прогнозований трафік. Такий підхід досяг загальної точності виявлення кібератак на рівні 98%, що підтверджує можливість випереджального виявлення загроз до настання деструктивної фази.

У дослідженні [4] авторами розроблено фреймворк BARTPredict для розгортання на платформі MEC. Одна модель BART Base виконувала функцію генератора наступного пакета в черзі, а модуль DistilBERT оцінював логічну узгодженість із точністю 92,8%. Друга модель BART Base виконувала бінарну класифікацію трафіку. Ця конфігурація забезпечує точність класифікації шкідливого трафіку на рівні 92,86%.

Дослідження підтверджують, що використання авторегресійної природи Transformer-моделей дозволяє з високою точністю симулювати мережову активність наперед, а комбінація з двоспрямованим контекстним аналізом BERT мінімізує пропуск цільових мультиетапних атак.

Виділення та класифікація індикаторів компрометації на основі LLM

На стратегічному (хмарному) рівні фреймворку першочерговим аналітичним завданням є автоматизоване вилучення та класифікація індикаторів компрометації з неструктурованих джерел. Основна перешкода для автоматизації полягає в неоднорідності представлення даних у сучасних джерелах: один звіт може містити IP-адреси та домени в таблицях, а інший у звичайному тексті, коді чи командному рядку, а деякі джерела описують індикатори суто словами. Через це стандартизовані формати, як-от STIX/TAXII, не завжди охоплюють повний обсяг оперативної інформації. [11] Для вирішення цієї проблеми вдало підходить гібридний метод, який розділяє обробку даних на два логічні процеси, швидке детерміноване вилучення кандидатів та їхню подальшу контекстну класифікацію за допомогою великих мовних моделей. Строго структуровані індикатори, такі як IPv4, IPv6, URL, домени, хеші тощо, ефективно виявляються регулярними виразами чи спеціалізованими парсерами, що мінімізує обчислювальні витрати. Оскільки формальна відповідність шаблону не гарантує шкідливості об'єкта, наприклад, IP-адреса може належати легітимному ресурсу або автору звіту, мовна модель залучається для оцінки контексту появи кожного кандидата й ухвалення остаточного рішення щодо його приналежності до загроз.



Для контекстуального аналізу формується вектор вхідних даних моделі:

$$X_i = \{I_i, T_i, C_i, M_i\}, \quad (3)$$

де I_i відповідає за значення індикатора, T_i – його первинно визначений синтаксичний тип, C_i – текстовий контекст (фрагмент тексту або весь документ), а M_i – метадані джерела. Якщо дані надходять із кіберзлочинних форумів, відновлення контексту вимагає використання назви теми для нових обговорень або попередньо згенерованого підсумку дискусії для уже існуючих гілок, що дозволяє моделі враховувати передісторію обговорення. На основі цих даних LLM виконує багатокласову класифікацію та формує структуровану оцінку:

$$y_i = \{type, role, confidence, context\}$$

Це дає змогу визначити семантичну роль індикатора, вказати рівень упевненості та надати текстове обґрунтування висновку.

Застосування LLM розширює традиційне поняття IOC, дозволяючи класифікувати семантично неоднозначні сутності без строгої структури (назви шкідливого ПЗ, файлів, інструментів атаки чи ключів реєстру), які виглядають як звичайні слова й потребують контекстного розпізнавання. Модель успішно пов'язує індикатор із його функціональним призначенням у тексті, відповідаючи на питання про його роль у конкретному інциденті, а не просто констатує факт його наявності. Завдяки здатності до zero-shot та few-shot класифікації, такий підхід не потребує збору й ручної розмітки великих навчальних вибірок. У складніших сценаріях архітектура інтегрує RAG: після первинного виявлення індикатора LLM формує запит до зовнішніх баз знань, таких як MITRE ATT&CK чи VirusTotal, для отримання актуальних даних. Зрештою, нормалізовані результати разом із доказовими фрагментами зберігаються в СТІ-базі та пошукових індексах на рівні синхронізації та транслюються іншим субкомпонентам фреймворку для подальшого проактивного реагування.

Результати емпіричної апробації двоетапного гібридного вилучення індикаторів компрометації в дослідженні [9] підтверджують життєздатність цього субкомпоненту. Дослідження було проведено на корпусі з 479 вебсторінок, що містили загалом 2658 IOC, із репрезентативною верифікацією через антивірусну базу VirusTotal. Аналіз шести мовних моделей показав, що модель Gemini 1.5 Pro забезпечила абсолютний показник повноти виявлення на рівні 1,0 за точності 95,8%, гарантуючи повну відсутність пропущених шкідливих індикаторів, тоді як менші чи універсальні моделі, як-от Llama 8B, Qwen3 32B, мали суттєві пропуски або високий рівень хибних спрацювань.

Варто зазначити, що аналіз помилок у роботі мовних моделей вказує, що основні труднощі полягають у класифікації доменних імен і розрізненні специфічного синтаксичного контексту. Наприклад, моделі Gemini схильні помилково класифікувати доброякісні шаблони адрес як активні індикатори загрози. Проте для завдань проактивного виявлення пріоритетним є саме високий показник повноти, який дозволяє мінімізувати ризик пропуску реальних загроз за умови збереження прийнятного рівня точності, що запобігає перевантаженню аналітиків.

Атрибуція суб'єкта загрози та ТТР на основі LLM

Стратегічний рівень фреймворку доповнює субкомпонент атрибуції, що призначений для семантичного визначення ймовірного суб'єкта кібератаки та характерних тактик, технік і процедур, що відповідають спостережуваній активності.

Вхідними даними для цього субкомпоненту є неструктуровані звіти про загрози, описи інцидентів, IOC, відомості про шкідливе ПЗ та інфраструктуру, а також, за необхідності, інформація з CVE, MITRE ATT&CK та інших СТІ-джерел.

Основна складність процесу атрибуції полягає в тому, що інформація про нападника не представлена у вигляді однозначного поля. Звіти можуть окремо описувати ПЗ, С2-інфраструктуру, використані техніки, цілі атаки, часові характеристики, попередню активність, а зв'язок цих ознак із конкретним суб'єктом необхідно встановити семантично. Тому завданням субкомпоненту є встановлення зв'язку між сукупністю спостережуваних ознак і профілем відомих суб'єктів загрози.

Першим етапом є попередня структуризація наявних даних. З документів виокремлюються сутності, що мають значення для атрибуції. Для цього може використовуватися LLM-based Named Entity Recognition, яка безпосередньо виокремлює та класифікує сутності в тексті. Модель отримує текстові вхідні дані та повертає структурований перелік сутностей із зазначенням їхніх типів.

Надалі LLM виконує пошук і класифікацію зв'язків між сутностями. На цьому етапі визначаються



семантичні зв'язки між об'єктами. Такі відносини можна подати у вигляді структурованих трійок: суб'єкт, зв'язок, об'єкт. Модель повинна використовувати лише зв'язки, підтверджені вхідним контекстом.

Після цього формується профіль спостережуваної атаки:

$$E = \{IOC, ПЗ, ТТР, Інфраструктура, Ціль, Час\}$$

Для кожного відомого суб'єкта формується аналогічний профіль, що містить характерні ТТР, цілі, попередні кампанії тощо. Завдання LLM полягає у визначенні того, наскільки поточний набір доказів відповідає одному з таких профілів.

Важливо, що LLM повинна враховувати, комбінацію ознак. Наприклад, окремий домен може використовуватися різними злочинними групами, тоді як поєднання ПЗ, методів персистенції, C2-комунікації, цільової галузі та специфічної техніки експлуатації може характеризувати конкретного суб'єкта.

Саме тут проявляється основна перевага LLM. Замість жорстких правил модель здатна аналізувати зв'язки між великою кількістю неоднорідних ознак. Модуль логічного виведення використовується саме для встановлення таких зв'язків і оцінювання найбільш імовірної атрибуції на основі контекстних ознак.

Паралельно з визначенням суб'єкта виконується вилучення ТТР. LLM аналізує опис дій зловмисника та зіставляє їх із поняттями тактики, техніки й процедур. Тобто модель повинна перейти від природномовного опису, наприклад: «зловмисник отримав доступ до хоста через вразливість у процесі автентифікації в BMC Supermicro, скинув root-пароль хоста та встановив persistence за допомогою *gspnetcat*», до структурованого набору ТТР.

Це відрізняється від простого зіставлення ключових слів, оскільки одна й та сама техніка може описуватися різними формулюваннями. LLM використовує семантичний контекст для нормалізації таких описів у спільну репрезентацію СТІ.

Для процесу атрибуції ТТР результат є множиною, тому його можна оцінювати за показниками Precision, Recall, F1 та IoU, що вкаже, наскільки передбачений LLM набір технік збігається з еталонним.

Атрибуція залежить від актуальності інформації. Суб'єкт може змінювати інфраструктуру, використовувати нове ПЗ або модифікувати ТТР. Тому доцільно не покладатися виключно на параметри самої LLM. Для цього субкомпонент може використовувати RAG: після виділення ключових сутностей LLM формує запит до зовнішніх СТІ-джерел, отримує актуальні дані та використовує їх як додаткові докази.

Зовнішній пошук може звертатися до MITRE ATT&CK, VirusTotal, Shodan, агрегаторів threat feeds та інших джерел. Отримані дані додаються до контексту LLM, після чого виконується повторний аналіз. Це дозволяє відокремити здатність моделі до міркування від статичності її внутрішніх знань.

Важливо, що результатом має бути ймовірнісна атрибуція з набором доказів.

Результати дослідження [7] вказують на суттєву перевагу спеціалізованої СТІ-моделі над універсальними LLM. Наприклад, точність атрибуції суб'єкта для ChatGPT-4o становила лише 40,31%, тоді як для CyLens-1B — 79,52%, а для CyLens-8B — 83,74%. ТТР CyLens-8B продемонструвала результат 93,74%, перевищивши CyLens-70B із 88,26%. Це показує, що для атрибуції важливішими є спеціалізація моделі та інтеграція СТІ-відомостей, а не лише збільшення кількості параметрів.

Додатковий аналіз показав, що видалення NER з CyLens-8B знижувало точність атрибуції суб'єкта з 84,23% до 71,12% для відомих загроз і з 81,17% до 70,19% для загроз нульового дня. Якісне виділення сутностей є критичним компонентом атрибуції.

Атрибуцію суб'єкта загрози та ТТР на основі LLM доцільно розглядати як семантичне зіставлення спостережуваної сукупності СТІ-доказів із профілями відомих суб'єктів загроз. Ключова перевага LLM полягає в можливості одночасно інтерпретувати численні ознаки в межах необхідного контексту та встановлювати зв'язки між ними, а використання технологій NER, вилучення зв'язків та RAG дозволяє перетворити таке міркування на структурований і відтворюваний процес.

Пріоритизація вразливостей за допомогою LLM

Одну з ключових ролей на стратегічному рівні відіграє субкомпонент пріоритизації вразливостей. Традиційні підходи до управління вразливостями постійно стикаються з проблемою масштабованості через стрімке щорічне зростання кількості зафіксованих CVE та ускладнень, що виникають у сучасних ІТ-системах. Використання LLM дозволяє автоматизувати та суттєво підвищити точність ранжування загроз за ступенем їхньої небезпеки й імовірністю практичної експлуатації, що оптимізує розподіл обмежених ресурсів безпекових команд для усунення найбільш критичних ризиків.



У межах наукових досліджень оцінювання та пріоритизація загроз за допомогою LLM здійснюється за двома ключовими напрямками:

- 1) Оцінювання серйозності на основі метрик стандарту CVSS 3.x. Компонент передбачає налаштування та інструктування мовної моделі для класифікації текстових описів загроз навіть за умов відсутності офіційного ідентифікатора CVE або для класифікації вразливостей нульового дня за вісьма метриками CVSS. Моделі аналізують неструктуровані звіти для визначення параметрів: вектор атаки (AV), складність атаки (AC), необхідні привілеї (PR), взаємодія з користувачем (UI), область дії (S), а також вплив на конфіденційність (C), цілісність (I) та доступність (A). Параметри агрегуються в показник базового рівня серйозності (Base Severity), який поділяється на чотири категорії: Low, Medium, High та Critical. Кожна метрика CVSS спирається на специфічні контекстуальні докази, тому LLM є доволі ефективним інструментом для інтеграції та інтерпретації цих різномірних даних.
- 2) Оцінювання ймовірності експлуатації через інтеграцію метрик EPSS (Exploit Prediction Scoring System). Цей напрям орієнтований на аналіз динамічної поведінки загроз. Поєднуючи контекстуальний аналіз і регресійне моделювання, LLM виконує два завдання: інтерполяцію значень EPSS на попередніх часових проміжках і прогнозування майбутніх тенденцій. Важливо, що деякі кіберзагрози демонструють різкі стрибки показників EPSS внаслідок раптової появи раніше невідомих експлоїтів, тому моделі навчаються інтегрувати знання про реальні події у світі для коригування прогнозів ймовірності експлуатації вразливостей.

Для досягнення якісних результатів мовна модель піддається спеціалізованій кастомізації. Базова модель проходить масштабне доповнення знань шляхом донавчання на великому корпусі звітів CVE за допомогою прогресивної методології курикулярного навчання. Після цього застосовується інструкційне налаштування на основі фреймворку каскадного міркування, яке відображає природну залежність завдань у життєвому циклі СТІ. Починаючи з первинної атрибуції зловмисника, ідентифікації пов'язаних CVE, аналізу CVSS-метрик і визначення пріоритету реагування. Під час безпосереднього аналізу процес генерації структурується через спеціалізовані NLP-модулі, зокрема тематичне моделювання, вилучення іменований сутностей та відношень, а також RAG для залучення найсвіжіших веб-даних.

Пріоритизація вразливостей також успішно реалізується через zero-shot аналіз неструктурованих даних на хакерських форумах та динамічну інтеграцію СТІ-потоків у реальному часі.

Для аналізу неструктурованих даних з мережі впроваджуються системи, які щодня збирають повідомлення з тінювих майданчиків. LLM використовує розроблені інструкції з роллю СТІ-аналітика для автоматичного вилучення та кодування бінарних змінних, безпосередньо пов'язаних із процесом пріоритизації ризиків. Зокрема, мовні моделі визначають, чи є згадана вразливість віддалено експлуатованою та чи експлуатується вона активно на момент обговорення. Поєднання цих маркерів дозволяє аналітикам ухвалювати зважені рішення та спрямувати зусилля на найбільш небезпечні події.

У випадку динамічної інтеграції СТІ-потоків даних для подолання проблеми обмеження знань статичних LLM датою зрізу навчальної вибірки, впроваджується архітектура RAG. Автоматизовані фреймворки безперервно збирають свіжі оновлення з баз даних CVE, CWE, каталогів KEV CISA та щоденних оновлень EPSS. Кодують отримані дані у високорозмірні векторні представлення за допомогою ембедінг-моделей і зберігають їх у векторних базах даних. Під час аналітичного запиту система витягує найбільш релевантний семантичний контекст і передає його моделі генерації, що дозволяє отримати точний і актуальний аналіз пріоритизації в реальному часі.

Практична спроможність даного субкомпоненту підтверджується дослідженнями на основі реальних даних. Спеціалізоване донавчання має перевагу над універсальними хмарними моделями. У завданнях класифікації CVSS Base Severity модель CyLens-8B досягла точності 96,86% на історичних загрозах та 91,57% на загрозах нульового дня, що суттєво випереджає результати комерційної моделі GPT-o1: 69,25% і 64,71% відповідно. Версія CyLens-70B продемонструвала точність 97,57% для історичних загроз і 93,16% для загроз нульового дня. У цьому підході важливий етап попереднього контекстного навчання, оскільки відмова від нього призводить до обвалу точності моделі CyLens-8B до 53,62% [7].

Під час автоматизованого моніторингу діалогів на формах Exploit.in і RAMP середня точність вилучення СТІ-параметрів за допомогою моделі GPT-3.5-Turbo склала 96,23%. Для ознаки можливості віддаленої експлуатації точність досягла 97,2% (Precision = 94,9%, Recall = 82,2%), а для ознаки активної експлуатації — 97,3% (Precision = 75,0%, Recall = 94,1%) [6].

Інтеграція векторної бази даних Milvus допомагає розв'язати проблему статичності знань у генеративних моделях. У той час як базова GPT-4o не мала інформації про нові CVE через обмеження часового зрізу, RAG-система безпомилково вилучала з векторного сховища поточні показники EPSS, наприклад, EPSS 0.29708 для CVE-2019-16758, ідентифікувала наявність вразливості в каталозі CISA KEV та генерувала актуальні рекомендації щодо протидії [7].



Кореляція кібератак та кампаній за допомогою LLM

На стратегічному рівні підсумковим аналітичним етапом обробки є субкомпонент кореляції кібератак, спрямованим на виявлення взаємозв'язків між розрізненими індикаторами загроз, ідентифікаторами вразливостей та діями злоумисників для реконструкції цілісної картини злоумисної кампанії. Застосування великих мовних моделей дозволяє автоматизувати та масштабувати цей процес, зв'язуючи відомі ідентифікатори, такі як CVE чи CWE, і розкриваючи пов'язані загрози в межах однієї або кількох кампаній.

У межах життєвого циклу СТИ-аналізу завдання кореляції кампаній позиціонується як комплексний крок, що має чіткі залежності від попередніх етапів. Успішна кореляція вимагає наявності не лише первинних доказів інциденту, а й інтеграції результатів атрибуції загрози (ідентифікації хакерського угруповання та його тактик, технік і процедур — TTPs) і її контекстуалізації (визначення уражених систем та мережевої інфраструктури). Під час безпосередньої роботи моделі цей процес структуровано послідовно задіюється за допомогою спеціалізованих NLP-модулів. Спочатку модуль тематичного моделювання розпізнає намір користувача провести кореляцію, NER та модуль вилучення відношень ідентифікують ключові сутності та зв'язки між ними, а фреймворк RAG залучає свіжі зовнішні СТИ-дані із веб-ресурсів. На основі цього об'єднаного контексту модулі логічного висновку та сумаризації послідовно генерують структурований аналітичний звіт.

Водночас виникає проблема так званої розширеної кореляції, що полягає у визначенні супутніх або одночасних загроз, які не згадуються безпосередньо у первинних даних, проте є частиною тієї ж кампанії. Для розв'язання цього завдання можна використати різні архітектурні стратегії:

- 1) Пряме розширення, за якого модель намагається знайти пов'язані загрози безпосередньо в тексті первинного звіту.
- 2) Використання CVE як проміжних вузлів, де модель спочатку вилучає явні вразливості зі звіту, а потім використовує їх як смислові опорні точки для пошуку супутніх загроз у базах даних.
- 3) Використання CWE як проміжних вузлів, коли первинним кроком є класифікація типу слабкості системи (CWE), після чого здійснюється пошук пов'язаних CVE.

Стратегія з використанням CVE суттєво покращує точність кореляції, оскільки конкретні технічні параметри вразливостей звужують простір пошуку. З іншого боку, залучення абстрактних категорій CWE призводить до погіршення результатів. Одна категорія CWE може описувати тисячі різних вразливостей, тому це вносить надмірну семантичну невизначеність і шум, заважаючи моделі зосередитися на специфіці поточної атаки.

Практичну ефективність даного субкомпонента підтверджують результати дослідження [7], у якому практичні випробування продемонстрували перевагу спеціалізованих агентних моделей над універсальними LLM.

У завданнях прямого розширення спеціалізована модель CyLens-70B продемонструвала точність 77,5% за метрикою Hit@10, що полягає у визначенні наявності хоча б одного правильного корельованого ідентифікатора у першій десятці згенерованих варіантів, а CyLens-8B — 74,3%. У той самий час провідні універсальні моделі та СТИ-агенти зазнали невдачі: показник ChatGPT-o1 становив лише 8,1%, а Gemini-Pro — 2,2%. Це свідчить про те, що загальні моделі не здатні самостійно орієнтуватися у колосальному масиві взаємозв'язків вразливостей, що налічує понад 200 тисяч записів CVE, без цільового донавання.

Результати CyLens-70B з використанням CVE зросли до 81,2%, а CyLens-8B — до 79,0%. Застосування цієї стратегії базовими комерційними моделями призвело до накопичення помилок на першому кроці, що викривило подальший аналіз: точність ChatGPT-o1 досягла лише 25,4%, спеціалізованого СТИ-агента SevenLLM — 7,3%.

Через згадану проблему концептуальної розмитості точність кореляції впала в усіх моделях із використанням CWE. Показники CyLens-70B знизилися до 60,7%, а CyLens-8B — до 55,4%. Показники ж інших моделей наблизилися до нуля: ChatGPT-o1 — 0,8%, SevenLLM — 0,0%.

Обмеження, недоліки та ризики підходу.

Незважаючи на високу потенційну ефективність подібного фреймворку та його субкомпонентів, практичне впровадження супроводжується низкою суттєвих обмежень та обчислювальних бар'єрів, що напряму впливають на фінальну ефективність:

- Великі мовні моделі мають критично високі вимоги до обсягів пам'яті та обчислювальних потужностей. На рівнях Edge та MEC без застосування дистиляції чи квантування запуск таких моделей є ускладненим, а за умов дефіциту розмічених даних традиційні класичні архітектури, такі як Bi-LSTM, можуть перевершувати системи з інтеграцією LLM за швидкістю та



стабільністю.

- Універсальні LLM або моделі без безперервного доменного донавчання демонструють низьку повноту та схильні до критичних помилок під час виявлення рідкісних або нестандартних категорій кіберзагроз. Відповідно, потребують окремого доменного донавчання.
- Моделі схильні плутати часовий контекст: наприклад, не розрізняють актуальні оголошення про продаж експлойтів від історичних описів атак, припускаються помилок через мовні бар'єри та семантичну неоднозначність певних термінів, а також викривлення й галюцинації під час сегментації довгих текстів. Для запобігання накопиченню помилок та захисту системи від маніпулятивного або хибного самонавчання доцільно застосовувати механізми верифікації людиною [12].
- Використання мовних моделей з «коробки» без залучення RAG із динамічним оновленням векторних баз знань призводить до швидкої втрати актуальності щодо нових вразливостей і до виникнення семантичних помилок під час видалення ІОС.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У роботі запропоновано та обґрунтовано архітектуру наскрізного 3-рівневого фреймворку проактивної кіберрозвідки на основі використання великих мовних моделей, у якому висвітлені субкомпоненти реалізують різні методи проактивної кіберрозвідки. Наведені в літературі емпіричні результати слугують дієвим підтвердженням життєздатності запропонованих рішень та модулів. Запропонована система відповідає сучасним тенденціям розвитку галузі та охоплює широкий цикл захисту від стратегічної аналітики й оцінки ризиків до блокування поодиноких аномалій на периферії.

Водночас практична побудова такої монолітної екосистеми стримується низькою обчислювальних і архітектурних обмежень. Основними бар'єрами є висока ресурсна місткість моделей для периферійних пристроїв, часова асинхронія між швидким аналізом трафіку та тривалими запитами до хмарних LLM, ризик каскадного поширення помилок під час автоматичного реагування, а також відсутність семантичних протоколів для передачі ймовірного контексту мовних моделей.

З огляду на це, перспективними напрямками подальших досліджень є розробка прототипу та практична апробація запропонованого фреймворку в умовах реальної інфраструктури з урахуванням виявлених обмежень. Загалом, подальші напрямки в даній сфері передбачають автоматизацію побудови конвеєрів обробки СТІ-даних шляхом модульної контейнеризації, впровадження пояснюваної багатокрокової аргументації, розробку механізмів безперервного оновлення векторних баз знань для RAG-систем, а також інтеграцію поведінкових шаблонів на основі LLM у робочі процеси сучасних SOC-центрів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вавіленкова, А. (2026). Великі мовні моделі для оптимізації задач у сфері пошуку та захисту інформації. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 1(33), 833–841. <https://doi.org/10.28925/2663-4023.2026.33.1276>
2. Hasan, S. M., Alotaibi, A. M., Talukder, S., & Shahid, A. R. (2024). Distributed threat intelligence at the edge devices: A large language model-driven approach. *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*, 1496–1497. <https://doi.org/10.48550/arXiv.2405.08755>
3. Diaf, A., Korba, A. A., Karabadi, N. E., & Ghamri-Doudane, Y. (2024). Beyond detection: Leveraging large language models for cyber attack prediction in IoT networks. *2024 20th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, 117–123. <https://doi.org/10.48550/arXiv.2408.14045>
4. Diaf, A., Korba, A. A., Karabadi, N. E., & Ghamri-Doudane, Y. (2024). BARTPredict: Empowering IoT security with LLM-driven cyber threat prediction. *Research Paper*. <https://doi.org/10.48550/arXiv.2501.01664>
5. Otoum, Y., Asad, A., & Nayak, A. (2024). LLM-based threat detection and prevention framework for IoT ecosystems. *IEEE*. <https://doi.org/10.48550/arXiv.2505.00240>
6. Clairoux-Trépanier, V., Beauchamp, I.-M., Ruellan, E., Paquet-Clouston, M., Paquette, S.-O., & Clay, E. (2024). The use of large language models (LLM) for cyber threat intelligence (CTI) in cybercrime forums. *White Paper*, Université de Montréal & Flare Systems. <https://doi.org/10.48550/arXiv.2408.03354>
7. Liu, X., Liang, J., Yan, Q., Jang, J., Mao, S., Ye, M., Jia, J., & Xi, Z. (2025). CyLens: Towards reinventing cyber threat intelligence in the paradigm of agentic large language models. <https://doi.org/10.48550/arXiv.2502.20791>



8. Paul, S., Alemi, F., & Macwan, R. (2024). LLM-assisted proactive threat intelligence for automated reasoning. *IEEE Transactions*. <https://doi.org/10.48550/arXiv.2504.00428>
9. Chawla, A., & Prasad, U. (2025). Proactively detecting threats: A novel approach using LLMs. <https://doi.org/10.48550/arXiv.2601.09029>
10. Андреев, Д., Чорний, А., Стьопочкіна, І., & Ільїн, М. (2026). Метод автоматизації звітів про кіберінциденти з використанням LLM . Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 1(33), 274–285. <https://doi.org/10.28925/2663-4023.2026.33.1156>
11. OASIS Open. (n.d.). STIX introductory walkthrough. <https://oasis-open.github.io/ctidocumentation/stix/walkthrough>
12. Y. Momryk, Y., & Sabodashko, D. (2026). Model and methodology for the formation of intelligent human factor error management with adaptive self-learning in large language models. CEUR Workshop Proceedings, 4223, 113–126. <https://ceur-ws.org/Vol-4223/paper9.pdf>

**Mykhailo M. Yeromkin**

Ph.D. student of the Department of Information Technology Security

Lviv Polytechnic National University, Lviv, Ukraine

ORCID: 0009-0008-6067-2382

mykhailo.yeromkin.asp.2025@lpnu.ua

Serhiy Semenyuk

PhD, Associate Professor of the Department of Information Technology Security

Lviv Polytechnic National University, Lviv, Ukraine

ORCID: 0000-0002-8143-5887

serhii.a.semenyuk@lpnu.ua

**END-TO-END METHODS OF LARGE LANGUAGE MODELS USAGE FOR
PROACTIVE CYBER INTELLIGENCE IN IOT/EDGE INFRASTRUCTURES**

Abstract. The rapid expansion of Internet of Things (IoT) networks and distributed edge computing, coupled with the increasing sophistication of cyber threats, has rendered traditional reactive security systems inadequate against targeted attacks and zero-day threats. This article aims to substantiate the architecture of an end-to-end framework and methods for utilizing Large Language Models (LLMs) in cyber threat intelligence to transition from traditional reactive defence to a proactive, context-aware security paradigm across distributed and enterprise IT infrastructures. The study proposes a three-tier ecosystem architecture within which the covered methods function as its specialized subcomponents: contextual anomaly detection in the network and system behavior of IoT/Edge devices; proactive prediction of the next packet and network activity type; extraction and classification of Indicators of Compromise (IoCs); attribution of threat actors and Tactics, Techniques, and Procedures (TTPs); vulnerability prioritization based on CVSS and EPSS metrics; and the correlation of cyberattacks and campaigns. For each method, the paper describes the theoretical model and the LLM integration approach (in-context learning, fine-tuning, RAG) and presents empirical results from analyzed sources. The tiers of the proposed architecture (strategic, synchronization, and operational levels) unite the considered methods into a single proactive cyber intelligence cycle. Additionally, the study identifies key constraints for practical implementation, such as the high resource demands of LLMs on edge devices, temporal asynchrony between edge and cloud layers, the risk of cascading error propagation, and the lack of unified semantic standards for communicating the models' probabilistic inferences.

Keywords: information security; large language models; cyber intelligence; proactive cybersecurity; IoT; edge computing; indicators of compromise; threat attribution; TTP; vulnerability prioritization; CVSS/EPSS; retrieval-augmented generation (RAG).

REFERENCES

1. Vavilenkova, A. (2026). Large language models for optimizing tasks in information retrieval and protection. *Cybersecurity: Education, Science, Technique*, 1(33), 833–841. <https://doi.org/10.28925/2663-4023.2026.33.1276>
2. Hasan, S. M., Alotaibi, A. M., Talukder, S., & Shahid, A. R. (2024). Distributed threat intelligence at the edge devices: A large language model-driven approach. *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*, 1496–1497. <https://doi.org/10.48550/arXiv.2405.08755>
3. Diaf, A., Korba, A. A., Karabadi, N. E., & Ghamri-Doudane, Y. (2024). Beyond detection: Leveraging large language models for cyber attack prediction in IoT networks. *2024 20th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, 117–123. <https://doi.org/10.48550/arXiv.2408.14045>
4. Diaf, A., Korba, A. A., Karabadi, N. E., & Ghamri-Doudane, Y. (2024). BARTPredict: Empowering IoT security with LLM-driven cyber threat prediction. *Research Paper*. <https://doi.org/10.48550/arXiv.2501.01664>
5. Otoum, Y., Asad, A., & Nayak, A. (2024). LLM-based threat detection and prevention framework for IoT ecosystems. *IEEE*. <https://doi.org/10.48550/arXiv.2505.00240>.



6. Clairoux-Trépanier, V., Beauchamp, I.-M., Ruellan, E., Paquet-Clouston, M., Paquette, S.-O., & Clay, E. (2024). The use of large language models (LLM) for cyber threat intelligence (CTI) in cybercrime forums. *White Paper*, Université de Montréal & Flare Systems. <https://doi.org/10.48550/arXiv.2408.03354>
7. Liu, X., Liang, J., Yan, Q., Jang, J., Mao, S., Ye, M., Jia, J., & Xi, Z. (2025). CyLens: Towards reinventing cyber threat intelligence in the paradigm of agentic large language models. <https://doi.org/10.48550/arXiv.2502.20791>
8. Paul, S., Alemi, F., & Macwan, R. (2024). LLM-assisted proactive threat intelligence for automated reasoning. *IEEE Transactions*. <https://doi.org/10.48550/arXiv.2504.00428>
9. Chawla, A., & Prasad, U. (2025). Proactively detecting threats: A novel approach using LLMs. <https://doi.org/10.48550/arXiv.2601.09029>
10. Andreiev, D., Chornyi, A., Stopochkina, I., & Ilin, M. (2026). Methodology for automating cyber incident reports using LLM. *Cybersecurity: Education, Science, Technique*, 1(33), 274–285. <https://doi.org/10.28925/2663-4023.2026.33.1156>
11. OASIS Open. (n.d.). STIX introductory walkthrough. <https://oasis-open.github.io/ctidocumentation/stix/walkthrough>
12. Y. Momryk, Y., & Sabodashko, D. (2026). Model and methodology for the formation of intelligent human factor error management with adaptive self-learning in large language models. *CEUR Workshop Proceedings*, 4223, 113–126. <https://ceur-ws.org/Vol-4223/paper9.pdf>

Отримано редакцією журналу / Received: 05.09.26

Прорецензовано / Revised: 18.09.26

Схвалено до друку / Accepted: 26.10.26



This work is licensed under Creative Commons Attribution-noncommercial-sharealike 4.0 International License.