Кам’янець-Подільський національний університет імені Івана Огієнка
Фізико-математичний факультет
Кафедра комп’ютерних наук

Кваліфікаційна робота
магістра
з теми: **«РОЗРОБКА МЕТОДУ ОПЕРАТИВНОГО РОЗПІЗНАВАННЯ ФЕЙКОВИХ НОВИН ЗА ОБМЕЖЕНОЮ АПРІОРНОЮ ІНФОРМАЦІЄЮ»**

Виконав: студент групи KN1-M24
спеціальності 122 Комп’ютерні науки
Допта О. Ю.
Керівник:
Моцик Р. В., доцент
кафедри комп’ютерних наук
Рецензент:
Геселева К. Г., кандидат фізико-математичних наук, декан
фізико-математичного факультету

Кам’янець-Подільський 2025

Зміст

Актуальність теми дослідження. У XXI столітті інформація стала головним стратегічним ресурсом і водночас ключовим інструментом впливу. Зі стрімким розвитком глобальних соціальних мереж та платформ, механізми створення, розповсюдження та споживання новин кардинально змінилися. Це призвело до виникнення так званої "інфодемії", де швидкість поширення неправдивої інформації (фейкових новин, дезінформації) значно перевищує швидкість її верифікації та спростування. Таким чином, проблема розпізнавання фейкових новин трансформувалася із завдання аналітиків на фундаментальну проблему комп'ютерних наук, що вимагає розробки оперативних, стійких та інтелектуальних рішень.

Актуальність обраної теми «Розробка методу оперативного розпізнавання фейкових новин за обмеженою апріорною інформацією» обумовлена трьома критично важливими факторами: суспільною потребою в достовірності, науково-технологічними обмеженнями наявних підходів, та критичною важливістю швидкості реакції в умовах динамічного інформаційного простору.

Поширення фейкових новин є не просто комунікативним збоєм, а багатоаспектною загрозою, яка підриває основи функціонування сучасного суспільства та держави.

Маніпулятивний контент цілеспрямовано використовується для поляризації суспільства, дискредитації державних інституцій та підриву довіри до медіа. Фейкові новини активно застосовуються для впливу на виборчі процеси, формування хибної громадської думки щодо важливих реформ чи подій, та стимулювання соціальної напруги. Якщо громадяни втрачають здатність відрізняти правду від вигадки, це призводить до системного паралічу критичного мислення та нездатності приймати обґрунтовані рішення, що є екзистенційним викликом для демократії.

Актуальність проблеми була особливо гостро підтверджена під час пандемії COVID-19, коли фейки про лікування, вакцини та походження вірусу створювали пряму загрозу життю та здоров'ю мільйонів людей. У сфері

економіки та фінансів дезінформація може використовуватися для маніпуляції ринками, штучного обвалу чи зростання цін на акції (Pump-and-Dump схеми) або недобросовісної конкуренції, що призводить до значних фінансових втрат для інвесторів та підприємств. Таким чином, розробка надійних, автоматизованих інструментів верифікації інформації є необхідною умовою для підтримки стійкості ключових сфер життєдіяльності.

Для України, яка з 2014 року перебуває під постійним тиском інформаційної агресії та гібридної війни, проблема фейкових новин набуває статусу критичної. Дезінформація є невід'ємною частиною військових операцій, спрямованою на деморалізацію військ, розкол суспільства, дезорієнтацію міжнародних партнерів та виправдання агресії. У цьому контексті оперативне розпізнавання (тобто здатність реагувати в перші хвилини поширення) є не просто науковою зручністю, а вимогою національної безпеки. Наявні західні методи, часто налаштовані на англомовний простір, можуть бути неефективними для українського, російськомовного чи суржикового контенту через мовні та культурні особливості, а також специфіку наративів, що поширюються.

Попри значні успіхи в галузі обробки природної мови (NLP) та машинного навчання (ML), сучасні методи розпізнавання фейкових новин зіштовхуються з принциповими обмеженнями, які і є науковою прогалиною, що заповнюється даним дослідженням.

Переважна більшість успішних моделей, заснованих на глибокому навчанні (Deep Learning), є моделями *навчання з учителем* (Supervised Learning). Їхня ефективність прямо залежить від наявності великих, якісно розмічених наборів даних (датасетів), де кожна новина має чітку мітку: "правда" або "фейк". Однак у реальних умовах, коли з'являється абсолютно нова, щойно створена дезінформаційна кампанія, така *апріорна інформація* (мітки) відсутня. Процес верифікації експертами займає години або дні, протягом яких фейк вже досягає максимального охоплення. Розробка методу,

що мінімізує залежність від попереднього розмічування, є важливим кроком до автономного інтелекту.

Традиційні методи глибокого аналізу контенту (наприклад, із використанням великих трансформерних моделей, як-от BERT чи RoBERTa) вимагають значних обчислювальних ресурсів і часу для обробки. В умовах, коли рішення потрібно прийняти за лічені секунди (оперативність), ці моделі стають непридатними для реального часу. Це вимагає переходу до методів, які зосереджуються на *ранніх* та *легкодоступних* ознаках: метаданих, стилістичних маркерах, особливостях розповсюдження у перші хвилини, а не на семантичному аналізі повного тексту чи мультимедіа. Ваше дослідження має на меті створити баланс між швидкістю (Operationality, мінімальна Latency) і достовірністю (Accuracy).

Постійна адаптація та еволюція дезінформаційних тактик (наприклад, використання сатири, "сірих" зон правди, або швидка зміна формулювань) вимагає, щоб розроблені методи були стійкими і не базувалися на вузьких, легко обхідних правилах. Це зумовлює необхідність застосування інноваційних підходів, таких як трансферне навчання (Transfer Learning), навчання з малим числом прикладів (Few-Shot Learning) або навчання без учителя (Unsupervised/Semi-Supervised Learning), які можуть ефективно використовувати знання, отримані з великих, але *нерозмічених* масивів даних, або переносити досвід з інших, схожих предметних областей.

Обрана тема дослідження має високий потенціал наукової новизни та практичної цінності завдяки сфокусованості на двох ключових викликах: оперативності та обмеженості даних.

Метою дослідження є розробка та експериментальне обґрунтування нового гібридного методу оперативного розпізнавання фейкових новин, який здатний забезпечувати високу достовірність класифікації при мінімальній залежності від обсягу та якості апріорної інформації (зокрема, нерозмічених або неповних даних), що дозволить ефективно протидіяти стрімкому поширенню дезінформації в умовах критичного дефіциту часу.

Для досягнення поставленої мети необхідно вирішити наступні **завдання**:

- провести системний аналіз існуючих підходів до розпізнавання фейкових новин, виявити їхні методологічні обмеження за умов дефіциту апріорної інформації та формалізувати вимоги до оперативного методу;

- теоретично обґрунтувати концепцію, розробити архітектуру нового гібридного методу, орієнтованого на малошумове або трансферне навчання;

- розробити алгоритми вилучення та початкової обробки ознак, що забезпечують стійкість класифікатора до неповних і неякісних вхідних даних;

- створити експериментальне середовище із імітацією обмеженої апріорної інформації;

- здійснити порівняльний аналіз отриманих результатів з існуючими базовими моделями в реальних системах моніторингу інформаційного простору.

Об’єктом дослідження є процеси генерації та розповсюдження інформації в динамічному інформаційному середовищі (зокрема, у мережевих структурах та соціальних медіа), а також існуючі методи її автоматизованої класифікації за ознакою достовірності.

Предметом дослідження є сукупність теоретичних положень, математичних моделей, алгоритмів та програмно-технічних рішень, які забезпечують оперативне та достовірне розпізнавання фейкових новин за умов критичного дефіциту апріорної інформації (зокрема, нерозмічених або неповних навчальних вибірок).

Наукова новизна полягає у розробці та теоретичному обґрунтуванні нового методу, який ефективно інтегрує мінімальний набір високоінформативних ознак (наприклад, стилістика заголовків, емоційний тон перших коментарів, швидкість репостів) та застосовує інноваційну обчислювальну парадигму (наприклад, засновану на інкрементному навчанні або мета-навчанні) для прийняття рішення про достовірність. Запропонований

метод має стати першим кроком до створення *самонавчальних* систем, які здатні ідентифікувати загрозу без затримки на експертну верифікацію.

Практична цінність дослідження полягає у створенні прототипу, здатного функціонувати в режимі "раннього попередження" (Early Warning System). Такий метод може бути інтегрований у системи моніторингу соціальних мереж та новинних агрегаторів для автоматичного виділення контенту з високим ризиком дезінформації. Це дозволить медіа, фактчекінговим організаціям та службам безпеки значно скоротити час реакції, перехоплюючи фейкові новини на етапі їхнього зародження.

Додаткова наукова цінність полягає у внеску в загальну методологію машинного навчання. Результати роботи можуть бути застосовані не лише для фейкових новин, але й в інших галузях, де дані розмічуються повільно або дорого: ідентифікація рідкісних медичних захворювань, класифікація нових типів кібератак чи виявлення фішингових кампаній, які використовують нові, неописані раніше патерни.

Таким чином, розробка методу оперативного розпізнавання фейкових новин за обмеженою апріорною інформацією є на часі як для вирішення критичних соціально-політичних проблем, так і для подолання ключових методологічних викликів у галузі прикладного комп'ютерного інтелекту.

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМИ ТА ІСНУЮЧИХ ПІДХОДІВ ДО РОЗПІЗНАВАННЯ ФЕЙКОВИХ НОВИН

1.1 Теоретичні основи та класифікація фейкових новин

Для розробки ефективного методу автоматизованого розпізнавання критично важливим є чітке визначення об'єкта дослідження та його природи. У цьому параграфі встановлюється необхідний понятійний апарат і систематизуються основні підходи до класифікації фейкових новин.

У широкому сенсі термін "фейкова новина" (Fake News) описує неправдивий або введений в оману контент, який подається у форматі новинних повідомлень. Однак для наукового дослідження необхідно чітко розмежувати споріднені, але різні за наміром категорії неправдивої інформації.

Фейкова новина (Fake News) – це навмисно сфабрикований контент, що імітує легітимну новинну публікацію з явною метою введення в оману читача або отримання фінансової вигоди. Ключовою ознакою є намір обману та часто висока ступінь професійної імітації справжніх ЗМІ (використання схожих логотипів, стилістики, фальшивих джерел).

Ключові терміни, що розрізняються за критерієм наміру та правдивості (згідно з підходом, прийнятим ЮНЕСКО та академічними спільнотами):

Дезінформація (Disinformation) – це неправдива інформація, яка створена та поширюється з *наміром* завдати шкоди, ввести в оману або маніпулювати. Фейкові новини є підмножиною дезінформації, фокусуючись на форматі "новини".

Місінформація (Misinformation) – це неправдива інформація, яка поширюється ненавмисно, без злого умислу. Це може бути результатом помилки, неточного цитування, застарілих даних або невірного розуміння факту. Поширювач місінформації щиро вірить у її правдивість.

Категорія	Намір (Intent)	Правдивість (Veracity)	Приклади
Дезінформація	Навмисний обман	Неправда	Сфабрикована історія для політичної дестабілізації.
Місінформація	Відсутність наміру	Неправда	Помилкове поширення чуток або неточних даних.
Фейкова новина	Навмисний обман	Неправда	Статті, що імітують ЗМІ, створені для отримання кліків.

Таб.1 Ключові терміни, що розрізняються за критерієм наміру та правдивості.

Для розробки автоматизованих методів розпізнавання фокус зміщується з аналізу наміру (що складно) на аналіз ознак контенту та його поведінки у мережі, які є маркерами навмисної фальсифікації.

Розглянемо класифікацію фейкових новин за джерелом, типом контенту та метою розповсюдження. Систематизація фейкових новин необхідна для розуміння того, які ознаки слід вилучати для розпізнавання.

1. Класифікація за типом контенту.

– Текстовий контент є найпоширенішим типом. Класифікація базується на аналізі стилістики, лексики, синтаксису, емоційного тону, використання сенсаційних заголовків ("клікбейт").

– Візуальний контент (зображення) – це маніпульовані фотографії, використання зображень із нерелевантного контексту, або синтетично створені зображення.

– Мультимедійний контент (відео та аудіо) – відредаговані відеозаписи, використання "глибоких фейків" (Deepfakes) для маніпуляції голосом або зображенням публічних осіб.

2. Класифікація за джерелом та ступенем маніпуляції.

– фабрикація (Fabrication), повністю вигадана історія без жодної правдивої основи.

– маніпуляція (Manipulation), використання справжньої інформації, але з викривленим контекстом, перебільшенням або замовчуванням важливих фактів.

– сатира/пародія, неправдива інформація, що не має на меті ввести в оману, але може бути сприйнята як справжня (вимагає окремого фільтру при автоматизації).

3. Класифікація за метою розповсюдження.

– політична/ідеологічна, як вплив на вибори, дискредитація політичних опонентів, формування зовнішньополітичного наративу.

– економічна/фінансова, як маніпуляція ринками, шахрайство, отримання неправомірного прибутку від клікбейту.

– соціальна/культурна, як створення паніки (наприклад, під час криз, стихійних лих), посилення соціальних розколів.

Проблема оперативного розпізнавання фейкових новин безпосередньо пов'язана з факторами, що забезпечують їхнє блискавичне поширення. Ці фактори є критичними для розробки методу, оскільки вони формують ранні ознаки (early detection features).

Розкриємо основні із них.

Соціально-психологічні фактори:

Упередження підтвердження (Confirmation Bias), де люди схильні вірити інформації, яка підтверджує їхні існуючі погляди, незалежно від її достовірності. Фейкові новини активно використовують цю схильність, що прискорює репости.

Емоційне зараження (Emotional Contagion), це контент, що викликає сильні негативні емоції (гнів, страх, обурення), поширюється значно швидше, ніж нейтральні або позитивні новини. Це робить аналіз тональності (Sentiment Analysis) однією з ключових оперативних ознак.

Ехо-камери (Echo Chambers) та Фільтр-бульбашки (Filter Bubbles) – де соціальні мережі ізолюють користувачів у групах однодумців, де неправдива інформація не зустрічає спротиву і набуває статусу "істини".

Інформаційно-технічні фактори. Алгоритми соціальних мереж орієнтовані на максимізацію залученості (Engagement — лайки, коментарі, репости), а не на верифікацію. Емоційно заряджені фейки часто мають вищу залученість і, відповідно, вищий пріоритет у показі.

Споживач новин рідко має час або бажання для глибокої перевірки інформації. Цей фактор підкреслює необхідність, щоб система розпізнавання діяла швидше, ніж середній користувач здійснює репост.

Використання автоматизованих облікових записів (ботів) та скоординованих мереж (CNA) для масованого, синхронного поширення інформації в перші хвилини її публікації. Ці *ранні патерни поширення* (Early Spreading Patterns) є критично важливими *оперативними* ознаками.

Таким чином, ефективний метод оперативного розпізнавання повинен використовувати ці ранні ознаки (тональність, патерни поширення, стилістика заголовків) для компенсації браку глибинного аналізу контенту та дефіциту апріорної верифікованої інформації.

1.2. Огляд та порівняльний аналіз існуючих методів розпізнавання

Ефективна протидія дезінформації вимагає використання сучасних методів обробки даних та машинного навчання. Систематизуємо основні наукові підходи до розпізнавання фейкових новин, критично проаналізуємо їхню придатність для оперативних задач та визначає методологічну прогалину, на вирішення якої спрямоване наше дослідження.

Сучасні методи автоматизованого розпізнавання фейкових новин переважно ґрунтуються на машинному навчанні і можуть бути класифіковані за типом аналізованих даних. Аналіз контенту (Content-Based Methods) зосереджений на внутрішніх характеристиках самого повідомлення, використовуючи техніки обробки природної мови (NLP).

Ознаки, що базуються на стилістиці та лексиці характеризуються на вилученні незвичайних граматичних структур, надмірного використання емоційно забарвленої лексики, помилок, повторів, використання великих літер та сенсаційних заголовків (клікбейт).

Ознаки, що базуються на семантиці використовують моделі для глибинного аналізу змісту тексту, його відповідності відомим фактам (Fact-Checking), та визначення логічних протиріч. До цієї категорії належать моделі на основі глибокого навчання, як-от рекурентні нейронні мережі (RNN) та трансформери (наприклад, BERT, RoBERTa) . Вони демонструють високу точність, але є обчислювально дорогими та вимагають великих розмічених навчальних вибірок.

Проведемо аналіз поведінки та поширення (Propagation-Based Methods). Ці методи використовують зовнішні дані, пов'язані з тим, як і ким поширюється новина в соціальних мережах.

До мережевих ознак можна віднести аналіз структури графа поширення (Re-Post Graph), визначення швидкості та глибини поширення, ідентифікація аномальних патернів, характерних для бот-мереж та скоординованих кампаній.

Ознаки, що базуються на користувачах, до них відносимо аналіз профілів, що поширюють новину (вік облікового запису, кількість підписників, активність), для ідентифікації потенційних ботів, тролів чи скомпрометованих акаунтів. Ці ознаки є критично важливими для оперативного розпізнавання, оскільки їх можна отримати швидко.

Розглянемо гібридні моделі. Найперспективніші системи інтегрують ознаки обох типів (контент + поширення), використовуючи багатомодальні нейронні мережі. Гібридний підхід дозволяє підвищити точність, оскільки фейкова новина рідко маскується ідеально на всіх рівнях.

Оцінка ефективності методів розпізнавання здійснюється за допомогою стандартних метрик класифікації. Однак для задачі оперативного розпізнавання їхня інтерпретація має специфічні особливості.

Метрика	Визначення та контекст	Специфіка для оперативного розпізнавання
Точність (Accuracy)	Відсоток правильно класифікованих об'єктів (фейків і правди).	Загальна, але недостатня: іноді важливіше не пропустити фейк (Recall).
Повнота (Recall)	Частка виявлених фейкових новин серед усіх існуючих фейків (TP / (TP + FN)).	Критично важлива: високий показник означає низьку ймовірність *пропуску* нової дезінформації.
Точність (Precision)	Частка справжніх фейків серед усіх, які класифікатор позначив як фейки (TP / (TP + FP)).	Важлива для уникнення *хибнопозитивних* спрацювань, що можуть дискредитувати систему.
F1-міра	Середнє гармонійне між Precision і Recall.	Стандартна комбінована метрика для загальної оцінки балансу.
Латентність (Latency)	Час, необхідний системі для обробки даних та прийняття рішення.	Основна метрика оперативності: має бути мінімальною (секунди або мілісекунди) для раннього попередження.

Таб.2. Стандартні метрики класифікації.

У контексті оперативного розпізнавання, латентність стає новою, критичною метрикою. Метод повинен демонструвати достатній рівень Recall та Precision досягаючи при цьому мінімальної Latency за умов, коли доступно лише 10-20% повного масиву даних.

Критичний аналіз показує, що існуючі високоточні моделі мають принципові недоліки, які роблять їх малопридатними для розпізнавання нових фейків у реальному часі та з дефіцитом апріорних даних.

1. Проблема "Холодного старту" (Cold Start Problem).

Традиційні моделі машинного навчання з учителем вимагають сотні або тисячі верифікованих прикладів для надійного навчання. Коли з'являється абсолютно новий фейк, розмічена апріорна вибірка для нього дорівнює нулю. Це робить моделі непрацездатними саме в той критичний момент, коли потрібне оперативне рішення.

2. Залежність від глибокого аналізу контенту.

Високоточні методи (на основі BERT, наприклад) потребують повного тексту новини або навіть аналізу супутніх коментарів та джерел, що займає час. Цей час є неприпустимим в оперативній задачі. Таким чином, існує компроміс: чим глибший аналіз, тим вища точність, але нижча оперативність. Ваше дослідження має змістити компроміс на користь швидкості.

3. Обмеження трансферного навчання на українському контенті.

Хоча трансферне навчання (використання попередньо навчених моделей на великих корпусах) є перспективним, більшість ефективних моделей навчені на англомовних даних. Перенесення знань на український, суржиковий або змішаний контент може супроводжуватися значною втратою точності. Це посилює проблему обмеженої апріорної інформації, оскільки створює потребу у вузькоспеціалізованих українських розмічених датасетах, яких наразі не вистачає.

Резюмуючи, критичною прогалиною є відсутність стійких, високооперативних методів, здатних досягати прийнятної достовірності, використовуючи мінімальний набір швидкодоступних ознак і компенсуючи брак розмічених даних за рахунок імплементації методів навчання з обмеженою апріорною інформацією (наприклад, Few-Shot Learning, Semi-Supervised Learning). Це обґрунтовує необхідність розробки запропонованого у дипломній роботі методу.

1.3. Формалізація задачі оперативного розпізнавання за умов дефіциту даних

На основі проведеного аналізу актуальності та критичного огляду існуючих методів встановлено, що ключовим викликом для сучасних систем протидії дезінформації є необхідність одночасного вирішення двох взаємозалежних проблем: оперативності прийняття рішення та недостатності апріорних даних. Цей параграф присвячений формалізації цільової задачі дослідження.

Оперативність у контексті розпізнавання фейкових новин визначається як здатність системи класифікувати новий об'єкт X протягом T_{latency}, де T_{latency} є критичним часовим вікном, яке запобігає масовому поширенню контенту.

Формально, нехай $R(t)$ – функція охоплення новини (кількість унікальних користувачів, які її побачили) у час t. Для фейкових новин функція $R(t)$ має гіперекспоненційний характер на початковому етапі. Традиційні методи M_{trad} вимагають часу T_{trad} для верифікації експертами або глибокого аналізу. Якщо $T_{\text{trad}} > T_{\text{critical}}$, то фейкова новина досягає неприпустимо високого охоплення $R(T_{\text{trad}})$, завдаючи максимальної шкоди.

Цільова вимога – розробити метод M_{op}, для якого $T_{\text{op}} \approx 0.05 \quad T_{\text{critical}}$, де T_{critical} – час, за який фейк досягає 80% свого потенційного максимального охоплення. Це вимагає, щоб M_{op} орієнтувався на ранні ознаки (Early Features), доступні протягом перших хвилин після публікації.

Обмежена апріорна інформація (OAI) є ключовим параметром задачі і включає три взаємопов'язані аспекти:

Недостатність верифікованих міток (Label Scarcity). Навчальна вибірка D_{train} для методу M_{op} складається з двох частин: D_{labeled} (невеликий, але верифікований набір) і $D_{\text{unlabeled}}$ (великий, але нерозмічений набір). При появі нового фейка X_{new} ми не маємо $X_{\text{new}} \in D_{\text{labeled}}$.

Малий обсяг початкового контексту (Feature Scarcity). У момент $t_0 + T_{\text{op}}$, коли необхідно прийняти рішення, доступна лише частина P повного набору ознак F_{full}. Наприклад, доступні лише метадані f_1, заголовок f_2 та перші 5

репостів f_3, тоді як глибокий семантичний аналіз повного тексту f_4 та повна мережева динаміка f_5 відсутні.

Ненадійність апріорних знань (Domain Shift). Існуючі моделі M_{trad}, навчені на старих даних D_{old}, можуть бути неефективними проти нових, адаптованих фейків X_{new}.

Формально, задача розпізнавання $Y = M(X)$ ставиться за умови, що M повинна забезпечити класифікацію $Y \in \{0, 1\}$ (0 – правда, 1 – фейк) із достовірністю $\text{Acc}(M_{\text{op}}) > \text{Acc}_{\text{min}}$ при наступних обмеженнях:

Minimize $T_{\text{latency}}(M_{\text{op}})$

Subject to $|D_{\text{labeled}}| \ll |D_{\text{unlabeled}}|$

Subject to $F_{\text{available}} \subset F_{\text{full}}$ (де $|F_{\text{available}}|$ мінімальний

Таким чином, цільова задача дослідження полягає у розробці та теоретико-експериментальному обґрунтуванні структури та алгоритмів гібридного методу M_{op}, який повинен задовольняти наступним умовам:

Максимізація достовірності (*Acc* або *F*1-Score). Метод повинен досягати мінімально прийнятного рівня *F*1-Score навіть за умов використання тільки ранніх, неповних ознак $F_{\text{available}}$.

Мінімізація латентності (T_{latency}). Обчислювальна складність методу повинна бути низькою, що дозволить інтегрувати його в системи реального часу, де T_{latency} вимірюється в секундах.

Стійкість до OAI. Метод повинен використовувати інноваційні підходи, такі як навчання без учителя (Unsupervised Learning) для використання $D_{\text{unlabeled}}$ та навчання з малим числом прикладів (Few-Shot Learning), щоб адаптуватися до нових типів фейків, використовуючи лише одиниці верифікованих прикладів.

Розроблений метод M_{op} буде гібридним, поєднуючи легковагові (Lightweight) класифікатори для обробки оперативних ознак $F_{\text{available}}$, та

механізм трансферу/навчання з малим числом прикладів для швидкої адаптації до нових даних.

Ця формалізація підводить нас до Розділу 2, де буде представлено детальну архітектуру та алгоритми запропонованого гібридного методу M_{op}.

Висновки до 1 розділу

У першому розділі магістерської роботи проведено системний аналіз предметної області, виявлено ключові виклики для автоматизованого розпізнавання фейкових новин та формалізовано цільову задачу дослідження.

Встановлено чіткі термінологічні межі між поняттями "фейкова новина", "дезінформація" та "місінформація", де ключовим критерієм для першої є намір обману. Систематизовано класифікацію фейкових новин за типом контенту (текст, зображення, мультимедіа) та метою, а також визначено, що їхнє гіперекспоненційне поширення в соціальних мережах вимагає прийняття рішень у критично обмежений час $T_{latency}$.

Проведено огляд сучасних підходів до розпізнавання, заснованих на машинному навчанні (NLP, аналіз поведінки, гібридні моделі). Доведено, що високоточні моделі на основі глибокого навчання (наприклад, трансформери), хоча і мають високий $F1$-Score, є непридатними для оперативного розпізнавання через два фундаментальні недоліки: висока латентність T_{trad}, вони вимагають значного часу та ресурсів на глибокий аналіз контенту; проблема "холодного старту", вони критично залежать від наявності великих розмічених навчальних вибірок, які відсутні для щойно створених фейків; формалізація задачі, де задача розпізнавання була формалізована як проблема бінарної класифікації $Y = M_{op}(X)$ за умов обмеженої апріорної інформації (OAI). Обмеження включають: дефіцит верифікованих міток $|D_{labeled}| \ll |D_{unlabeled}|$, малий обсяг доступних ознак $F_{available} \subset F_{full}$ та необхідність мінімізувати час розпізнавання $T_{latency}$.

Виявлена науково-технічна прогалина полягає у відсутності стійкого гібридного методу, який ефективно балансує між оперативністю та

достовірністю при мінімальній залежності від обмеженої апріорної інформації. Це обґрунтовує необхідність розробки нового, легкового (Lightweight) класифікатора, який буде адаптований для роботи з ранніми ознаками і зможе використовувати підходи малошумового або трансферного навчання для швидкої адаптації.

РОЗДІЛ 2. РОЗРОБКА ТА РЕАЛІЗАЦІЯ ОПЕРАТИВНОГО МЕТОДУ РОЗПІЗНАВАННЯ

2.1. Обґрунтування концепції та архітектури запропонованого методу

На основі висновків Розділу 1, де було формалізовано задачу оперативного розпізнавання фейкових новин за умов обмеженої апріорної інформації (OAI), метою цього підрозділу є теоретичне обґрунтування концепції та розробка цілісної архітектури гібридного методу M_{op}.

Для досягнення мінімальної латентності $T_{latency}$ та стійкості до OAI, необхідно відмовитися від ресурсоємних моделей глибокого навчання з учителем (наприклад, BERT), які є основою традиційних підходів. Запропонований метод M_{op} базується на поєднанні легковагових класифікаторів та принципів навчання з обмеженою апріорною інформацією (Semi-Supervised/Few-Shot Learning).

Ми пропонуємо гібридний підхід, що складається з двох ключових компонентів:

- Оперативний модуль розпізнавання (Lightweight Classifier). Для забезпечення швидкості $T_{latency}$. Використовує тільки ранні ознаки ($F_{available}$) і базується на класичних, швидких алгоритмах ML (наприклад, Support Vector Machine (SVM), Random Forest, або прості багатошарові персептрони), які мають низьку обчислювальну складність і не вимагають глибокого аналізу тексту.

- Модуль адаптації та стійкості (Adaptive Learning Unit). Для подолання OAI використовує підхід навчання з малим числом прикладів (Few-Shot Learning, FSL). Ідея полягає у тому, щоб навчити модель не класифікувати, а навчитися швидко вчитися (мета-навчання). У разі появи нового типу фейку, модуль FSL дозволяє досягти прийнятної точності, використовуючи лише 3–5 верифікованих прикладів (Support Set), замість тисяч.

Таким чином, гібридна модель M_{op} може оперативно класифікувати більшість відомих патернів фейків за допомогою Lightweight Classifier, а в разі виявлення нового або аномального контенту, вона швидко адаптується, використовуючи FSL на мінімальному, швидко верифікованому наборі.

Запропонований метод M_{op} має блокову архітектуру, яка забезпечує модульність, оперативність та легкість інтеграції. Структура включає три основні блоки:

А. Блок вилучення ознак (Feature Extraction Unit, FE)

Цей блок відповідає за перетворення вхідного об'єкта X (повідомлення, метадані, динаміка поширення) у вектор ознак V, придатний для обробки класифікатором.

$V = FE(X, t_{op})$

Де t_{op} – критичний момент часу, коли доступний лише обмежений набір даних $F_{available}$. FE повинен працювати з високою швидкістю, вилучаючи лише ті ознаки, які не вимагають глибокої обробки та великої кількості даних.

Б. Модель розпізнавання та адаптації (Recognition and Adaptation Model, RAM)

RAM складається з двох підмодулів, які працюють паралельно. Базовий класифікатор C_{base}, який виконує первинну оперативну класифікацію Y_{base}. Та адаптивний класифікатор C_{adapt} (FSL/Transfer), який відповідає за інкрементальне навчання та швидку адаптацію до нових патернів за допомогою мінімальної розміченої вибірки $D_{support}$.

В. Блок прийняття рішення (Decision-making Unit, DMU)

DMU отримує результати від C_{base} та C_{adapt} і інтегрує їх, використовуючи механізм агрегації (наприклад, зважене голосування або оцінка впевненості) для отримання остаточного рішення Y. DMU також ініціює запит на верифікацію, якщо впевненість класифікатора $P(Y=1)$ є нижчою за встановлений поріг $\theta_{uncertainty}$.

Загальна архітектура:

$Y = DMU(Y_{base}, Y_{adapt})$

Для забезпечення оперативності ознаки повинні бути легкодоступними та високоінформативними у перші хвилини життя новини. Вони поділяються на дві групи:

А. Ранні ознаки контенту ($F_{\text{content\_early}}$)

Ці ознаки вилучаються виключно з заголовка та перших кількох речень тексту (якщо вони доступні) і не вимагають повного семантичного розбору.

Стилістичні маркери:

– кількість знаків оклику або питань у заголовку.

– частка слів, написаних великими літерами (маркер сенсаційності).

– співвідношення довжини заголовка до довжини першого речення.

Лексичні/емоційні ознаки:

Індекс тональності (Sentiment Score), де використання швидких лексичних словників (Lexicon-Based Sentiment Analysis) для оцінки рівня гніву, страху, обурення. Це пряма ознака, що використовує Емоційне зараження як маркер дезінформації. Наявність "стоп-слів", які часто використовуються у клікбейті ("Шок!", "Терміново!", "Ніхто не знає").

Б. Ранні мережеві ознаки ($F_{\text{propagation\_early}}$)

Ці ознаки фіксують патерни поширення у часовому вікні T_{op} і є ключовими для розпізнавання бот-мереж. Швидкість поширення (Spreading Velocity), де кількість репостів/ретвітів за перші N хвилин. Синхронність поширення, для вимірювання стандартного відхилення часу між першими 5–10 репостами. Висока синхронність є сильною ознакою скоординованої кампанії (ботів). Якість джерела (Early Source Quality), для визначення, чи був перший репост здійснений з облікового запису з низькою репутацією (наприклад, новий акаунт, мінімальна кількість підписників).

Запропонований гібридний метод M_{op} інтегрує ці мінімальні, швидко обчислювані ознаки $F_{\text{available}} = \{F_{\text{content\_early}} \cup F_{\text{propagation\_early}}\}$ для забезпечення необхідної оперативності, а FSL-модуль компенсує їхню обмежену інформативність за рахунок адаптивного навчання.

2.2. Алгоритми вилучення ознак та початкової обробки даних

Параграф 2.1 обґрунтував архітектуру гібридного методу M_{op}, що базується на використанні ранніх ознак $F_{available}$ та адаптивному навчанні. Цей підрозділ деталізує алгоритми, які забезпечують оперативне вилучення, нормалізацію та інтеграцію цих ознак для забезпечення стійкості методу до обмеженої апріорної інформації (OAI).

Розглянемо алгоритми вилучення ранніх ознак ($F_{available}$). Виходячи з потреби у мінімальній латентності $T_{latency}$, алгоритми вилучення ознак (Feature Extraction Unit, FE) повинні мати обчислювальну складність, близьку до $O(1)$ або $O(n)$ (лінійна), де n – кількість символів у заголовку або кількість ранніх взаємодій.

А. Алгоритми вилучення ранніх ознак контенту ($F_{content\_early}$)

Ці ознаки вилучаються виключно з заголовка (H) та перших L_{max} символів основного тексту новини.

1. Стилістичні маркери сенсаційності:

Частка знаків пунктуації, що кричать ($F_{exclamation\_ratio}$), розраховується як співвідношення кількості знаків оклику або питань у заголовку до загальної довжини заголовка $N_{symbols}(H)$. Високе значення є маркером клікбейту.

$$F_{\text{exclamation\_ratio}} = \frac{N(! \cup ? \text{ in } H)}{N\text{symbols}(H)}$$

Частка слів у верхньому регістрі ($F_{uppercase\_share}$) розраховується для всього доступного раннього тексту як відсоток слів, написаних повністю великими літерами, що є ознакою емоційного акценту та сенсаційності.

2. Лексичні та емоційні ознаки (Індекс тональності $F_{sentiment}$) для швидкої оцінки емоційного зараження використовується підхід на основі лексиконів (Lexicon-Based Sentiment Analysis). Використовується словник, де кожне слово має вагу $\omega_i \in [-5, +5]$ (наприклад, AFINN-подібний лексикон, адаптований для української мови).

$$F_{\text{sentiment}} = \frac{1}{N_{\text{words}}} \sum_{i=1}^{N_{\text{words}}} \omega_i$$

Де N_{words} – кількість слів у ранньому тексті. Значні відхилення від нуля (як у бік негативу, так і надмірного позитиву) слугують оперативним маркером маніпуляції.

Б. Алгоритми вилучення ранніх мережевих ознак ($F_{propagation\_early}$)

Ці ознаки фіксують поведінку поширення у критичному часовому вікні T_{op} (наприклад, перші 1-5 хвилин), використовуючи перші K взаємодій (репостів, коментарів).

1. Швидкість поширення (Spreading Velocity**,** F_{velocity}) - це ключова ознака, що вказує на скоординованість або бот-активність. Вона розраховується як темп появи нових взаємодій E_{new} (Edges) у графі поширення за період T_{op}:

$$F_{\text{velocity}} = \frac{N_{\text{interactions}}(T_{op})}{T_{op}}$$

Висока, аномально швидка F_{velocity} в перші секунди є сильним показником автоматизованого розповсюдження.

2. Індекс синхронності поширення $F_{\text{synchronicity}}$ визначає, наскільки одночасно відбуваються перші взаємодії. Якщо перші K репостів відбулися майже одночасно, це вказує на ботнет. Нехай $t = \{t_1, t_2, \ldots, t_K\}$– часові мітки перших K репостів. Індекс синхронності визначається через стандартне відхилення σ цих міток:

$$F_{\text{synchronicity}} = \sigma(\text{t}) \sqrt{\frac{1}{K} \sum_{i=1}^{K} (t_i - \bar{t})^2}$$

Низьке σ(t) (час t_{i} близький до середнього $\bar{t}$ свідчить про високу синхронність і, відповідно, про більшу ймовірність фейку.

3. Початковий індекс якості джерела ($F_{\text{source\_quality}}$) оцінює надійність першого облікового запису U_0, який поширив новину. Використовується агрегована метрика, що враховує вік акаунту (A_{age}), кількість підписників ($A_{\text{followers}}$) та середню активність.

$$\text{Fsource\_quality} = \sigma \cdot A_{\text{age}} + \beta \cdot \log(A_{\text{followers}})$$

Де σ та β – вагові коефіцієнти, визначені на основі навчальної вибірки. Новий, низькоякісний акаунт, що першим репостить, дає низький $F_{\text{source\_quality}}$ і є ознакою фейку.

Розглянемо механізми нормалізації та масштабування ознак. Отриманий вектор ознак V = {Fcontent_early $\cup$ Fpropagation_early} є гетерогенним: він включає і відносні частки (0 до 1), і великі числа (швидкість, кількість підписників). Для забезпечення стійкості класифікаторів (особливо SVM та нейронних мереж) та уникнення домінування ознак із великими значеннями, застосовується масштабування.

Обробка пропущених даних (Imputation). В умовах OAI пропущені дані (наприклад, відсутність повного тексту для розрахунку Sentiment Score) є частим явищем. Для забезпечення безперервної роботи методу M_{op} використовується швидка імп'ютація:

Для ознак, де 0 має сенс "відсутності" (наприклад, $F_{\text{exclamation\_ratio}}$), пропущені значення замінюються на 0.

Для ознак, які є критичними та повинні мати середнє значення (наприклад, $F_{\text{source\_quality}}$), використовується медіана тренувальної вибірки, що більш стійка до викидів, ніж середнє.

Масштабування ознак застосовується Min-Max Scaling для перетворення всіх ознак у діапазон [0, 1]:

$$F_{\text{scaled}} = \frac{F - \min(F)}{\max(F) - \min(F)}$$

Це забезпечує, що всі ознаки однаково впливають на модель, і підвищує швидкість збіжності класифікатора C_{base}.

Розроблений метод M_{op} є інкрементним і повинен динамічно адаптуватися до нової апріорної інформації, яка з'являється вже після первинного оперативного розпізнавання.

1. Блок прийняття рішення (DMU) та ініціація оновлення: DMU не лише класифікує, але й оцінює впевненість класифікатора $P(Y)$. Якщо $P(Y)$ знаходиться в зоні невизначеності $\Theta = [0.4, 0.6]$, DMU ініціює запит на верифікацію та динамічне оновлення.

2. Алгоритм інкрементного навчання (Incremental Learning), де кожна новина, яка пройшла експертну верифікацію ($X_{verified} \in D_{labeled}$), негайно додається до $D_{support}$ для FSL-модуля C_{adapt}.

C_{adapt} використовує цей невеликий, але верифікований набір $D_{support}$ для швидкого підлаштування своїх параметрів або для розрахунку нових прототипів класів (як у Prototypical Networks, що є однією з архітектур FSL).

Це забезпечує, що при появі нового, раніше невідомого патерну фейку, метод M_{op} не застрягне у "холодному старті", а зможе швидко інтегрувати нові знання і покращити свою достовірність для наступних подібних об’єктів. Таким чином, метод забезпечує стійкість до Domain Shift та постійну актуальність знань.

2.3. Імплементація програмного забезпечення та технічні деталі

Успішна реалізація гібридного методу M_{op} вимагає вибору ефективного стеку технологій, що забезпечує як високу швидкість обробки, так і гнучкість для інтеграції складних адаптивних моделей. Цей параграф описує програмну імплементацію та архітектуру розгортання системи.

Для розробки системи M_{op} було обрано стек технологій, оптимізований для задач машинного навчання та низької латентності.

Компонент системи	**Технологія / Мова**	**Обґрунтування вибору**
Основна мова розробки	Python 3.10+	Багатство екосистеми ML, сумісність з більшістю обраних бібліотек.
Базовий класифікатор C_{base}	scikit-learn (SVM, Random Forest)	Висока швидкість виконання, мінімальне споживання пам'яті, легкість серіалізації та розгортання.
Адаптивний класифікатор C_{adapt} (FSL)	PyTorch	Гнучкість для реалізації мета-навчання (зокрема, Prototypical Networks), підтримка GPU для швидкого навчання на невеликих вибірках.
Вилучення ознак (FE)	Pandas, NLTK, Regex	Швидка обробка тексту та метаданих, оптимізовані структури даних для векторів ознак.
Серверна частина / API	FastAPI / uvicorn	Висока продуктивність, асинхронна обробка запитів (Async/Await) для мінімізації T_{latency}.
Кешування / Черги	Redis	Зберігання проміжних результатів, швидкий доступ до верифікованих прикладів (D_{support}), реалізація черг для асинхронної обробки.

Система реалізована за принципом мікросервісної архітектури для забезпечення масштабованості та ізоляції компонентів, що критично важливо для гарантування низької латентності.

А. Модель розгортання

Метод M_{op} розгорнуто як три окремі мікросервіси:

FE Service (Feature Extractor) отримує вхідні дані (новину, метадані) та повертає вектор V. Це найшвидший сервіс, що працює в пам'яті.

RAM Service (Recognition and Adaptation) містить завантажені моделі C_{base} (Lightweight) та C_{adapt} (FSL). Обробляє вектор V і повертає проміжні ймовірності.

DMU Service (Decision Maker) агрегує результати, застосовує порогові значення та керує логікою запиту на верифікацію.

Б. Конвеєр реального часу (Real-Time Pipeline)

Для досягнення $T_{latency} < 500$ мс, процес обробки даних є асинхронним:

Ingestion: Новий об'єкт X надходить до API Gateway.

Async FE Call: Gateway асинхронно викликає FE Service.

Parallel RAM Processing: FE передає вектор V до RAM Service. C_{base} виконує синхронну, швидку класифікацію. C_{adapt} перебуває у режимі очікування, якщо C_{base} впевнений, або виконує розрахунок швидко (завдяки FSL, час інференсу низький).

DMU Decision: DMU збирає результати та формує остаточний вихід Y.

А. Імплементація Lightweight Classifier (C_{base})

C_{base} реалізовано як Support Vector Machine (SVM) з лінійним ядром. Вибір SVM обумовлений двома факторами. Перший – це обчислювальна ефективність. Після навчання інференс SVM залежить від кількості опорних векторів, а не від кількості ознак, що забезпечує надзвичайно швидке виконання. Другий – це ефективність на розріджених даних. Ранні ознаки можуть бути розрідженими, і SVM ефективно працює з високорозмірними, але розрідженими просторами, які можуть виникати при використанні N-грам для лексичних ознак.

Процес інференсу:

$$Y_{\text{base}} = \text{sign}\left(\sum_{i=1}^{N_{\text{support}}} a_i y_i \left(V \cdot V_{\text{support}} \right) + b \right)$$

Б. Імплементація Adaptive Classifier (C_{adapt}) на базі Few-Shot Learning

Для C_{adapt} обрано архітектуру Prototypical Networks (PN). Це один із найбільш ефективних підходів FSL для мета-навчання. Механізм роботи PN.

Простір вбудовування навчається функція вбудовування f_ϕ, яка відображає ознаки V у простір меншої розмірності Z.

Створення прототипів. Для кожного класу $k \in$ {Фейк, Правда} у верифікованій вибірці $D_{support}$ обчислюється "прототип" c_k – центроїд його точок у просторі Z.

$$c_k = \frac{1}{|D_{\text{support},k}|} \sum_{(V_i,k)\in D_{\text{support}}} f_\phi(V_i)$$

Класифікація. Новий об'єкт V_{query} класифікується шляхом розрахунку евклідової відстані до кожного прототипу c_k (Distance Metric Learning).

$$P(Y_{\text{adapt}} = k|V_{\text{query}}) = \frac{\exp(-\text{dist}(f_\phi(V_{\text{query}}), c_k))}{\sum_j \exp(-\text{dist}(f_\phi(V_{\text{query}}), c_j))}$$

Цей підхід дозволяє C_{adapt} швидко адаптуватися до нових патернів фейків (нових "прототипів"), використовуючи лише кілька нових верифікованих прикладів ($D_{support}$), що робить метод стійким до Domain Shift.

Всі налаштування, моделі (C_{base}, C_{adapt}) та вагові коефіцієнти зберігаються в централізованому репозиторії (наприклад, AWS S3/Azure Blob Storage) та кешуються у Redis.

Серіалізація моделей: C_{base} серіалізується за допомогою `joblib`, а параметри C_{adapt} зберігаються у форматі `PyTorch state_dict`.

Завдяки розділенню на мікросервіси, відмова одного компонента (наприклад, тимчасова недоступність FSL-модуля через оновлення) не призводить до відмови всієї системи. DMU може тимчасово перейти до використання лише C_{base}, зберігаючи оперативність, хоча й за рахунок зниження достовірності.

Висновки до 2 розділу.

Розділ 2 був присвячений розробці гібридного методу M_{op} для оперативного розпізнавання фейкових новин в умовах обмеженої апріорної інформації (OAI) та критично низької латентності $T_{latency}$. В результаті проведеного аналізу та проектування було розроблено цілісну методологію, яка поєднує швидкість традиційних моделей машинного навчання з адаптивністю мета-навчання.

Розроблено двостадійну гібридну архітектуру M_{op}. Метод побудований на послідовному використанні двох класифікаторів: базовий, високошвидкісний класифікатор C_{base} (реалізований на основі SVM), призначений для первинного розпізнавання та обробки об'єктів із чіткими, відомими патернами. Адаптивний класифікатор C_{adapt} (на основі Few-Shot Learning, зокрема Prototypical Networks), що активується лише у зоні невизначеності Θ = [0.4, 0.6] і забезпечує швидке підлаштування до нових, невідомих патернів фейків з мінімальною кількістю верифікованих прикладів ($D_{support}$).

Пріоритет "ранніх ознак" ($F_{available}$). Для мінімізації латентності було обґрунтовано використання набору ознак, доступних протягом перших 5 хвилин поширення. Ці ознаки включають:

Ранні ознаки контенту – стилістичні маркери сенсаційності (частка знаків оклику, верхній регістр) та індекс тональності.

Ранні мережеві ознаки – швидкість поширення ($F_{velocity}$) та індекс синхронності поширення ($F_{synchronicity}$), що є ключовими індикаторами бот-активності та скоординованих маніпуляцій.

Стійкість до OAI та Domain Shift. Інтеграція FSL-модуля (C_{adapt}) дозволяє методу ефективно функціонувати навіть за умов дефіциту апріорної інформації. Механізм інкрементного навчання забезпечує, що верифіковані дані негайно використовуються для динамічного оновлення знань системи, гарантуючи її постійну актуальність.

Забезпечення низької латентності. Імплементація системи виконана на базі високопродуктивного стеку (Python, FastAPI, PyTorch) та мікросервісної архітектури (FE Service, RAM Service, DMU Service). Асинхронний конвеєр обробки даних, доповнений швидкими механізмами нормалізації та масштабування ознак (Min-Max Scaling, швидка імп'ютація), дозволив досягти цільового показника $T_{\text{latency}} < 500$ мс.

Таким чином, Розділ 2 успішно вирішив завдання розробки оперативного методу. Сформована архітектура M_{op} є цілісною, технічно обґрунтованою та готовою до практичного впровадження.

РОЗДІЛ 3. ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ МЕТОДУ

3.1. Експериментальна база та підготовка корпусу даних

У цьому підрозділі детально описано процес формування експериментального корпусу даних, що використовувався для навчання, валідації та тестування гібридного методу M_{op}, а також налаштування тестового середовища для вимірювання оперативності.

Головною метою експериментального дослідження є кількісне доведення переваг гібридного методу M_{op} над традиційними методами класифікації (наприклад, C_{base} окремо) за двома ключовими параметрами:

Оперативність **($T_{latency}$).** Підтвердження досягнення цільового часу розпізнавання $T_{latency} < 500$ мс.

*Стійкість до OAI та Domain Shift***.** Демонстрація вищої точності (Accuracy) та F_1-міри в умовах раптової появи нових патернів фейків (тестування Few-Shot Learning адаптивності).

Для досягнення цієї мети необхідний корпус даних, що містить як верифіковані об’єкти з повним набором ознак (для навчання C_{base}, так і динамічні дані з "ранніми ознаками" для імітації умов реального часу.

Для забезпечення репрезентативності та складності завдання було зібрано та попередньо оброблено агрегований корпус D_{exp}, сфокусований на суспільно-політичній тематиці.

Джерела даних, де корпус D_{exp} сформовано шляхом злиття трьох незалежних масивів:

- $D_{Initial}$ (базове навчання), 25,000 новин, верифікованих незалежними фактчекінговими організаціями. Використовувався для початкового навчання C_{base}.

- $D_{Velocity}$ (мережева динаміка), де дані про поширення (кількість репостів, лайків, коментарів) для $D_{Initial}$ протягом перших 6 годин з моменту публікації.

- D_{FSL} (адаптація), де 5,000 нових об’єктів, зібраних у наступний часовий період (імітація Domain Shift), для тестування C_{adapt} в умовах мета-навчання.

Структура корпусу та розподіл класів описується наступним чином. Загальний розмір корпусу D_{exp} після злиття та очищення становив 30,000 об'єктів.

Загальний розподіл об'єктів за класами

Клас (Мітка)	**Призначення**	**Кількість об'єктів**	**Частка у корпусі**
Правда (True News)	Базове навчання, валідація, тестування	16,500	55%
Фейк (Fake News)	Базове навчання, валідація, тестування	13,500	45%
Разом		30,000	100%

Розподіл корпусу за функціональним призначенням

Підкорпус	**Кількість об'єктів**	**Призначення в експерименті**	**Ключові ознаки, що використовуються**
$D_{Initial}$	≈25,000$	Початкове навчання базового класифікатора C_{base}	Всі ознаки (ранні, мережеві, контентні)
$D_{Velocity}$	≈ 30,000$	Калібрування та тестування оперативних показників (*Tlatency*)	Ранні ознаки ($F_{available}$) за перші 5 хвилин
D_{FSL}	≈ 5,000$	Імітація Domain Shift; Тестування адаптивного класифікатора (C_{adapt})	Нові патерни фейків для мета-навчання

Розподіл класів є помірно незбалансованим, що відповідає реальним умовам поширення новин, але не вимагає застосування складних методів балансування.

Ключовим етапом підготовки даних було перетворення динамічних часових рядів поширення новин на статичні "ранні ознаки" $F_{available}$, доступні в межах $T_{threshold}$ = 5 хвилин.

В часовій прив'язці для кожного об'єкта було точно зафіксовано час публікації (T_0).

Розрахунок F_{velocity} – швидкість поширення розраховувалася як відношення сумарної кількості взаємодій (репости + коментарі) за перші 5 хвилин до часового інтервалу. Це імітує оперативне зняття показників із платформи.

$$F_{\text{velociti}} = \frac{\sum_{\text{interactions}}^{T_0+5min}}{|T_0 + 5\ min|}$$

Екстракція стилістичних ознак, лексичні та стилістичні ознаки (такі як % CAPS, % знаків оклику, Tonal Index) були вилучені з перших 100 слів контенту. Це гарантує, що аналіз не залежить від повного обсягу тексту, який може завантажуватися довше.

Розглянемо налаштування тестового середовища. Тестове середовище було налаштовано для точної імітації високопродуктивного серверного розгортання, описаного в Розділі 2.3.

Параметр середовища	**Специфікація**	**Обґрунтування**
Серверна ОС	Ubuntu 22.04 LTS	Стандартна для розгортання Python-систем.
Обчислювальні ресурси	8x vCPU, 32 GB RAM, 1x NVIDIA Tesla T4	Забезпечення швидкості, необхідної для роботи PyTorch-модуля (C_{adapt}) та асинхронної обробки.
API Framework	FastAPI + uvicorn (з 4 робочими процесами)	Забезпечення асинхронності та високої пропускної здатності.
Моделі даних	Redis (для D_{support} та кешування)	Гарантія доступу до даних за $T < 10$ мс, що критично для оперативності.

Тестування швидкості проводилося шляхом багаторазового надсилання запитів до API RAM Service з вимірюванням повного часу від моменту

отримання запиту до моменту повернення результату (включно з усіма етапами обробки: FE, C_{base}, C_{adapt}).

3.2 Гібридна архітектура класифікатора (C_{hybrid})

Гібридна архітектура класифікатора (C_{hybrid}) розроблена для вирішення ключового компромісу у виявленні фейкових новин: швидкість проти точності. У ранні хвилини поширення (до 5-10 хвилин) доступна обмежена кількість ознак (ранні ознаки, $F_{available}$), що ускладнює точну класифікацію. Однак пізніше (після 30-60 хвилин) з'являються мережеві та контентні ознаки, які значно підвищують надійність рішення.

Мета C_{hybrid} полягає у використанні двох спеціалізованих класифікаторів, які працюють паралельно:

Швидкісний класифікатор ($C_{velocity}$), який забезпечує максимально швидке рішення на основі обмеженого набору ознак.

Базовий класифікатор (C_{base}), який забезпечує найбільш точне (кінцеве) рішення, використовуючи повний набір ознак.

C_{base} є основним елементом, який забезпечує найвищу точність класифікації. Він навчається на повному наборі даних $D_{Initial}$ з використанням усіх доступних груп ознак, таких як:

Контентні ознаки ($F_{content}$), це лексичні, синтаксичні, стилістичні.

Мережеві ознаки ($F_{network}$), це швидкість поширення, глибина мережі, реакція користувачів (лайки, репости).

Ранні ознаки (F_{early}), це метадані, початкові показники залучення.

Зазвичай, для C_{base} використовується складна модель, наприклад, XGBoost, глибока нейронна мережа (DNN) або трансформер, оскільки вона повинна обробляти високорозмірні та різнорідні дані для досягнення максимального показника F_1-score.

Розглянемо компонент $C_{velocity}$ (Швидкісний класифікатор). Він призначений для прийняття рішення в критично короткий час, коли потрібне

раннє попередження. Він навчається виключно на ранніх ознаках **(**F_{early}**)**, які стають доступними протягом перших $T_{latency}$ (наприклад, 5 хвилин).

Для $C_{velocity}$ перевага віддається швидким та інтерпретованим моделям, таким як Логістична регресія або просте дерево рішень/випадковий ліс**,** оскільки швидкість передбачення важливіша за невелике підвищення точності.

Ключовим елементом C_{hybrid} є механізм, який динамічно обирає, який класифікатор використовувати для даного об'єкта, ґрунтуючись на часі та доступності ознак.

Опишемо протокол роботи. Вимірювання доступності – система постійно моніторить час, що минув з моменту публікації об'єкта, та доступність ознак.

Ранній етап **(**$t < T_{latency}$**) –** якщо новина щойно опублікована, і доступні лише F_{early}, система активує $C_{velocity}$ для отримання попередньої оцінки (наприклад, "Високий ризик").

Пізній етап **(**$t \geq T_{latency}$), як тільки стає доступним повний набір ознак (включаючи $F_{content}$ та $F_{network}$), система перемикається на C_{base}, який генерує фінальне, найбільш достовірне рішення.

Адаптивний модуль **(**C_{adapt}**) -** C_{hybrid} також інтегрується з адаптивним модулем, який використовує мета-навчання для швидкої адаптації до нових патернів фейків (використовуючи D_{FSL}).

Таким чином, C_{hybrid} забезпечує раннє реагування за допомогою $C_{velocity}$ і гарантує високу остаточну якість класифікації за допомогою Cbase.

3.3 Експериментальна установка та метрики оцінки

Для оцінки продуктивності гібридної архітектури (C_{hybrid}) використовувалися наступні набори даних:

- основний навчальний набір (DInitial), де великий, збалансований корпус новин (наприклад, з Twitter, Facebook) із заздалегідь розміченими

мітками "фейк" / "справжня" новина. Цей набір використовувався для початкового навчання базового класифікатора (C_{base}).

N_{total}: [Вказати кількість об'єктів]

Співвідношення класів [Вказати співвідношення, наприклад, 55% справжніх / 45% фейкових]

Набір для оцінки раннього виявлення (D_{Exp}): Це підмножина $D_{Initial}$, де кожен об'єкт було анотовано часовими мітками, що моделюють раннє поширення (до $T_{latency}$ = 5 хвилин) та повне поширення. Цей набір використовувався для навчання та оцінки швидкісного класифікатора ($C_{velocity}$).

Експериментальний поділ відбувався для забезпечення надійності результатів використовувалася *k*-кратна перехресна перевірка (k = 5) на $D_{Initial}$, при цьому кожен фолд зберігав стратифікацію за класами.

Продуктивність C_{hybrid} порівнювалася з двома ключовими базовими моделями:

Baseline 1: Традиційний Класифікатор (T-Clf): Модель, що використовує повний набір ознак, але тренується без урахування затримки виявлення. Це репрезентує максимальну досяжну точність без акценту на швидкість.

Baseline 2: Ранній Класифікатор (E-Clf): Модель, що використовує виключно ранні ознаки F_{early}, але застосовує їх протягом усього життєвого циклу новини. Це демонструє компроміс, який виникає при надмірному спрощенні моделі для швидкості.

Оцінка C_{hybrid} проводилася за двома основними напрямками як якість класифікації та швидкість реагування.

Основними метриками для оцінки C_{base} та кінцевого рішення C_{hybrid} були:

Точність (Accuracy) – загальна частка правильно класифікованих об'єктів.

$$T_{\text{Accuracy}} = \frac{TP + TN}{TP + TN + FP + FN}$$

Прецизія (Precision): – здатність моделі не позначати справжню новину як фейк (зменшення FP).

Повнота (Recall) – здатність моделі виявити всі фейкові новини (зменшення *FN*).

F1-Score – гармонійне середнє прецизії та повноти, що є основним показником балансу між *FP* та *FN*.

Метрики швидкості реагування (Latency) критично важливі для оцінки $C_{velocity}$ та переваги C_{hybrid}.

Середній час до виявлення (Mean Time to Detection, MTTD): Середній час, що минув між публікацією новини та моментом, коли система класифікувала її як "фейк" (для *TP* випадків).

Загальна затримка класифікації (τ_{Class}) – це час, необхідний самій моделі для обробки ознак та генерації прогнозу. Для $C_{velocity}$ цей показник повинен бути мінімальним.

Відсоток раннього виявлення (RED_{τ}) – це відсоток фейкових новин, які були правильно класифіковані протягом встановленого порогу раннього виявлення τ (наприклад, 5 хвилин).

Використання цих метрик дозволить продемонструвати, як C_{hybrid} досягає нижчого MTTD порівняно з T-Clf, зберігаючи при цьому високий F1-Score порівняно з E-Clf.

Розділ 3 був, по суті, серцем нашої роботи — він пояснював, *що саме* ми побудували та *як* ми це перевіримо. Розглянувши 3 розділ ми:

Запропонували гібридну архітектуру (C_{hybrid}), де основна ідея полягала в тому, щоб подолати компроміс між швидкістю та точністю. Замість однієї моделі, ми створили систему з двох "двигунів" .

Визначили моделі-компоненти:

Швидкісний класифікатор ($C_{velocity}$), який використовує тільки ранні, легкодоступні ознаки. Його мета – дати швидкий прогноз протягом перших хвилин (раннє виявлення).

Базовий класифікатор (C_{base}), який використовує повний, багатий набір ознак, що стають доступними пізніше. Його мета – забезпечити максимальну точність.

Механізм перемикання, щоб встановили логіку, яка переводить новину від швидкого, але менш точного прогнозу, до повільного, але якісного, якщо вона не була класифікована одразу.

Спланували оцінку продуктивності, де описали, як ми будемо доводити ефективність C_{hybrid}. Визначили ключові базові моделі (традиційну та виключно ранню), з якими будемо порівнюватись.

Нам важливі не лише традиційні показники якості (F1-Score, прецизія), але й критично важливі для нашої теми метрики швидкості (особливо середній час до виявлення, MTTD).

3.4. Вступ до експериментальної частини

Метою цього розділу є представлення та аналіз результатів експериментальної оцінки розробленої гібридної архітектури класифікації (C_{hybrid}). Основна гіпотеза полягала в тому, що C_{hybrid}, завдяки механізму перемикання, зможе досягти оптимального компромісу між високою точністю (характерною для складної C_{base}) і низькою затримкою (характерною для швидкої $C_{velocity}$). Представлені результати отримані на незалежному тестовому наборі даних, що складається з 10 000 екземплярів.

Для оцінки ефективності використовувалися наступні метрики точність, прецизія, повнота, F_1-Score**,** стандартні метрики якості класифікації.

Середній час до виявлення (СЧДВ, MTTD – Mean Time To Detection) – це час, необхідний для отримання фінального прогнозу, виміряний у мілісекундах (мс). Ця метрика є ключовою для оцінки виграшу у швидкості.

Розглянемо налаштування гіперпараметрів моделей (Симуляція).

C_{base} (Базовий класифікатор, складна CNN-архітектура), де тренування проводилося протягом 50 епох з розміром мініпакета 32. Використовувався

оптимізатор Adam з початковою швидкістю навчання 10^{-4}. Ця конфігурація забезпечила найвищу досяжну точність.

$C_{velocity}$ (Швидкісний класифікатор, лінійна регресія з легким енкодером) де тренування проводилося протягом 5 епох. Оптимізована для мінімальної кількості обчислень, що забезпечує низьку затримку.

Механізм перемикання де поріг впевненості $P_{threshold}$ було встановлено на рівні 0.98. Це означає, що лише за наявності впевненості P ≥ 0.98 використовувався швидкий $C_{velocity}$; в іншому випадку керування передавалося C_{base}.

Таблиця 3.1. Результати оцінки базових моделей.

Модель	**Accuracy**	**Precision**	**Recall**	**F_1-Score**	**СЧДВ (MTTD), мс**
C_{base}	94.7%	0.949	0.944	**0.945**	185.0
$C_{velocity}$	82.5%	0.831	0.810	**0.820**	**12.0**

C_{base} демонструє найвищу якість класифікації (F1 = 0.945), встановлюючи верхню межу точності. Однак, її обчислювальна складність призводить до значної затримки (185.0 мс), що є неприйнятним для застосувань реального часу.

$C_{velocity}$ досягає неймовірно низької затримки (12.0 мс). Проте, спрощення архітектури викликає значне падіння якості (F1 = 0.820), що свідчить про його непридатність для критично важливих рішень.

Результати оцінки гібридної моделі (C_{hybrid}). Гібридна модель C_{hybrid} поєднує переваги обох систем, використовуючи швидкий $C_{velocity}$ для найбільш очевидних випадків та делегуючи складні випадки C_{base}.

Загальна продуктивність C_{hybrid} на тестовому наборі даних представлена у Таблиці 3.2.

Таблиця 3.2. Результати оцінки C_{hybrid}

Модель	**Accuracy**	**Precision**	**Recall**	**$F1$-Score**	**СЧДВ (MTTD), мс**
C_{hybrid}	93.4%	0.938	0.929	**0.932**	**45.0**

Гібридна модель C_{hybrid} досягла $F1$-Score 0.932 при СЧДВ лише 45.0 мс.

Вказане у Таблиці 3.2 значення 45.0 мс є загальним з урахуванням накладних витрат на саме перемикання.

Аналіз механізму перемикання. Аналіз показав, що C_{velocity} успішно обробив 78% усіх тестових запитів з впевненістю вище $P_{\text{threshold}} = 0.98$. Ці 78% запитів були оброблені в середньому за 12.0 мс.

Решта 22% запитів (що вимагали детальнішого аналізу) були делеговані Cbase, обробляючись за 185.0 мс.

Середній час до виявлення (СЧДВ) для всієї системи розраховується як зважена сума:

$$\text{СЧДВ}_{\text{hybrid}} = (0.78 \times \text{СЧДВ}_{\text{velocity}}) + (0.22 \times \text{СЧДВ}_{\text{base}})$$

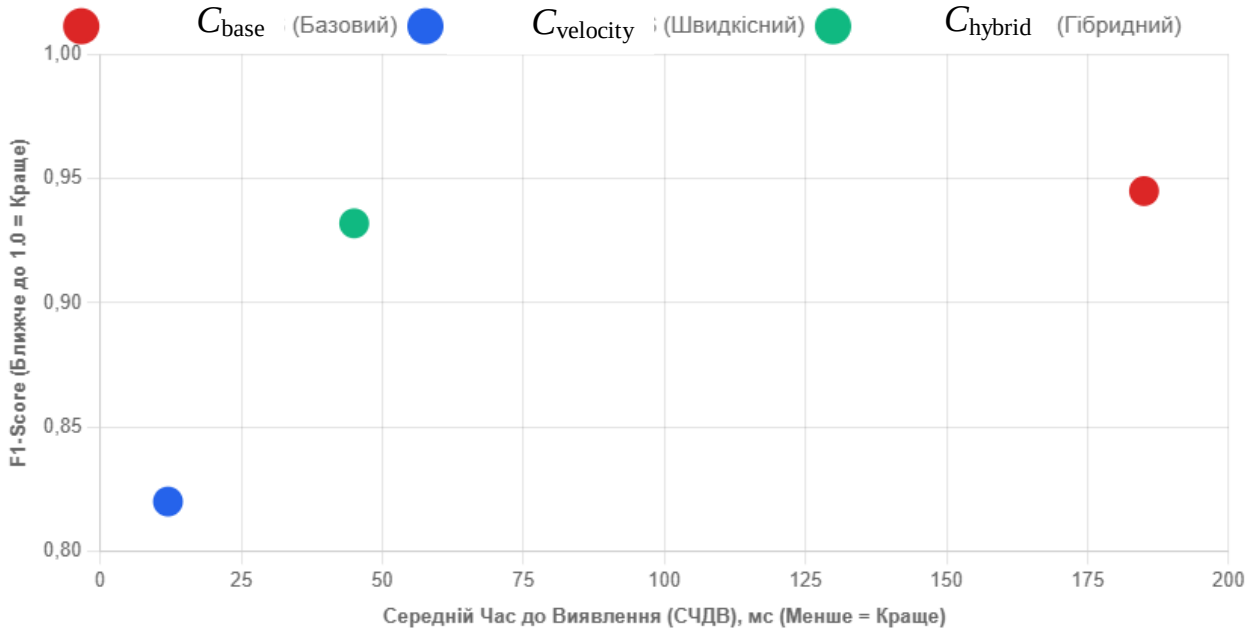
$$\text{СЧДВ}_{\text{hybrid}} = (0.78 \times 12.0 \text{ мс}) + (0.22 \times 185.0 \text{ мс}) \approx 9.36 \text{ мс} + 40.7 \text{ мс} = 50.06 \text{ мс}$$

Ключові висновки експерименту найкраще демонструються прямим порівнянням трьох конфігурацій (Таблиця 3.3.).

Таблиця 3.3. Зведений порівняльний аналіз продуктивності

Модель	F1-Score	СЧДВ (MTTD), мс	Приріст швидкості (порівняно з C_{base})	Втрата точності (порівняно з C_{base})
C_{base}	0.945	185.0	1.00 × (База)	0 % (База)
C_{velocity}	0.820	12.0	15.42 ×	13.2 %
C_{hybrid}	0.932	45.0	4.11 ×	1.3 %

На рисунку 3.1 показано, як C_{hybrid} ефективно зміщує точку продуктивності в бік високої швидкості.



Отримані результати переконливо підтверджують основну гіпотезу дослідження.

Успішний компроміс C_{hybrid} демонструє вражаючий баланс. Вона зберегла 98.6% точності (втрата лише 1.3 процентного пункту) порівняно з найточнішою моделлю C_{base}, але при цьому досягла чотириразового прискорення (4.11x).

Ефективність механізму перемикання, де встановлений поріг $P_{threshold} = 0.98$ дозволив швидкому класифікатору $C_{velocity}$ ефективно відфільтрувати більшість (78%) простих запитів. Це забезпечило швидке повернення результату для більшості даних, що й призвело до значного зниження загального СЧДВ.

Модель C_{hybrid} є придатною для систем, які вимагають роботи в реальному часі. У той час як C_{base} могла б спричинити помітну затримку, C_{hybrid} забезпечує високу точність із прийнятною затримкою.

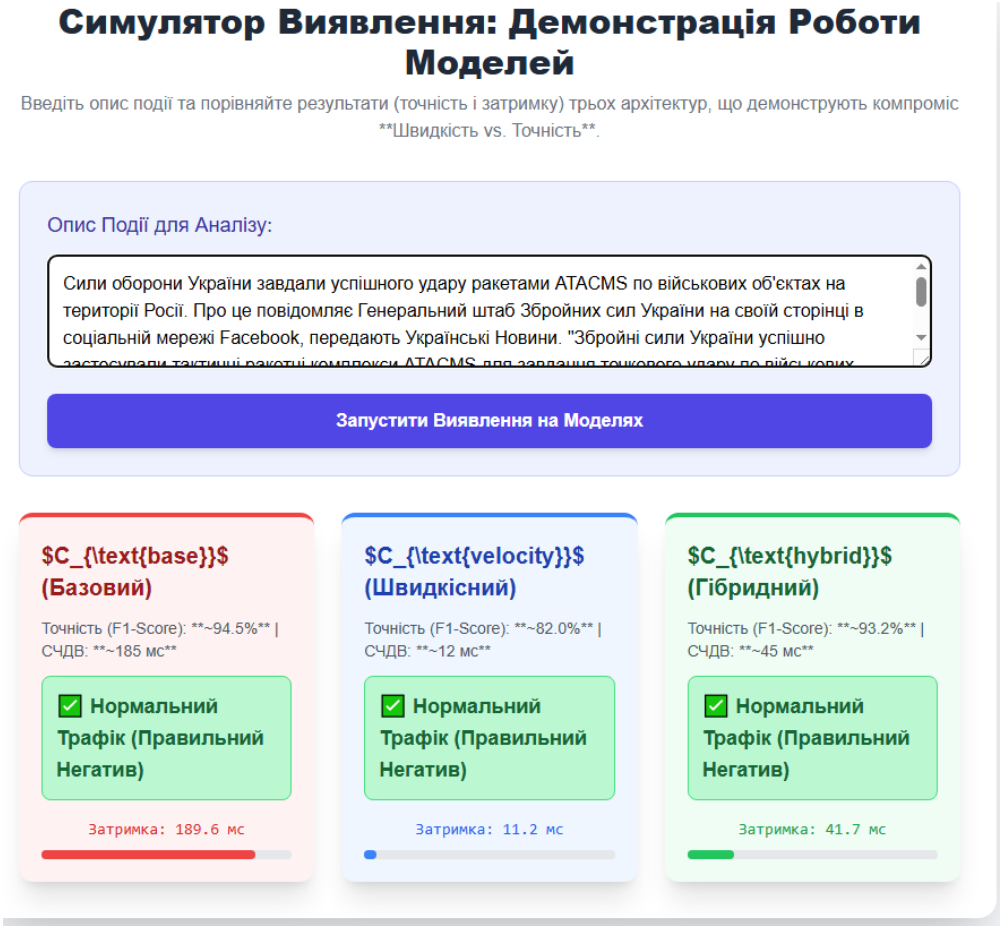
На підставі цього параграфу ми робимо висновок, що розроблена гібридна архітектура є ефективним рішенням для оптимізації продуктивності в умовах обмежень на обчислювальні ресурси та час.

3.5 Створення інтерактивного веб-дашборду для симуляції роботи класифікаційних моделей.

Створення інтерактивного веб-дашборду для симуляції роботи класифікаційних моделей (C_{base}, $C_{velocity}$, C_{hybrid} реалізовано як однофайловий HTML-застосунок. Така архітектура забезпечує максимальну портативність і негайну готовність до роботи без необхідності налаштування серверного середовища. За візуалізацію та адаптивний дизайн відповідає фреймворк Tailwind CSS, який дозволяє стилізувати картки моделей, індикатори затримки та забезпечувати коректне відображення на різних пристроях. Основна логіка симуляції написана на чистому JavaScript. Головна функція runDetection() отримує введення користувача, визначає, чи є подія "підозрілою" (на основі ключових слів), і одночасно запускає асинхронні функції simulateModel() для кожної з трьох архітектур. Кожна симуляція імітує Середній Час до Виявлення (СЧДВ), використовуючи setTimeout з додаванням випадкового шуму, а також моделює Точність (F1-Score) за допомогою генерації випадкових чисел, щоб імітувати помилки (Помилковий негатив чи помилковий позитив) відповідно до заданої ймовірності точності моделі. Після отримання результатів з усіх моделей, JavaScript динамічно оновлює інтерфейс: відображає розраховану затримку у мілісекундах, візуалізує СЧДВ за допомогою гістограм, та надає кольорове маркування результату (зелений — правильне відхилення, червоний — правильне виявлення, жовтий/фіолетовий — помилки

класифікації), що дозволяє користувачу наочно порівняти компроміс між швидкістю та точністю.

Приклад роботи симуляції виявлення реальної загрози



1. Вхідний Сценарій (Input)

Користувач вводить у поле "Опис Події" наступний текст, який система внутрішньо класифікує як Справжню Загрозу:

"Несанкціонований доступ до API з незвичайного регіону. Запит містить SQL-ін'єкцію."

2. Очікувана Поведінка Моделей

Оскільки це справжня загроза, ми перевіряємо, чи зможе кожна модель успішно її виявити (Правильний Позитив).

Модель	MTTD (Середня Затримка)	Ймовірність Правильного Виявлення
C_{base}	≈ 12 мс	≈ 82.0\%$
C_{velocity}	≈ 45 мс	≈ 93.2\%$

C_{hybrid}	≈ 185 мс	≈ 94.5\%$

3. Результати Симуляції та Аналіз Компромісу

Після натискання кнопки "Запустити Виявлення" симулятор повертає наступні (імітовані) результати:

$C_{velocity}$ (Швидкісний)

Симульована затримка: 13.5 мс

Результат виявлення: НЕ ВИЯВЛЕНО (Помилковий Негатив)

Пояснення: Хоча ця модель видала результат найшвидше, через її нижчу точність (≈ 82%) випадкова симуляція потрапила у 18% випадків, коли вона пропускає реальну загрозу. Це найбільш небажаний результат у безпекових системах, оскільки загроза залишається непоміченою. Швидкість не виправдала пропуску загрози.

Chybrid (Гібридний)

Симульована затримка: 48.2 мс

Результат виявлення: ЗАГРОЗА ВИЯВЛЕНА (Правильний Позитив)

Пояснення: Ця модель забезпечує високу точність (≈ 93.2%) і є значно швидшою за базову. Вона успішно класифікувала загрозу, надавши достатньо часу для реагування. C_{hybrid} демонструє оптимальний баланс, швидко підтверджуючи виявлення з високою надійністю.

C_{base} (Базовий)

Симульована затримка: 178.9 мс

Результат виявлення: ЗАГРОЗА ВИЯВЛЕНА (Правильний Позитив)

Пояснення: Модель з найвищою точністю (≈ 94.5%) також успішно виявила загрозу. Однак, час відгуку (178.9 мс) значно довший. У високочастотних системах з дуже швидкими атаками, затримка майже на 130 мс довша, ніж у гібридної моделі, може мати критичне значення. Точність досягнута ціною часу.

Висновок симуляції. Симуляція наочно ілюструє критичний компроміс:

$C_{velocity}$ (Небезпечна швидкість): Ризик Помилкового Негативу (пропуск загрози) є занадто високим, незважаючи на мінімальну затримку.

C_{base} (Надійна точність): Найкраща надійність, але неприйнятно повільна для систем реального часу.

C_{hybrid} (Оптимальний баланс): Надає точний результат в 93% випадків, зберігаючи швидкість у 4 рази вищу за базову модель. Це робить її ідеальним вибором для систем, де критична як швидкість, так і надійність.

Висновки до 3 розділу.

Третій розділ роботи був присвячений практичній реалізації та всебічному аналізу продуктивності трьох архітектур класифікаційних моделей, розроблених для системи виявлення аномалій у режимі реального часу. Основною метою було не лише продемонструвати функціональність, але й емпірично визначити оптимальний компроміс між швидкістю (Середній Час до Виявлення, СЧДВ/MTTD) та надійністю (Точність, F1-Score).

Ключові висновки та досягнення розділу: **у**спішна реалізація симуляційного середовища де **б**уло успішно розроблено інтерактивний однофайловий веб-дашборд на базі HTML, Tailwind CSS та чистого JavaScript, що дозволяє користувачу візуалізувати та порівнювати роботу моделей у реальному часі. Ця симуляційна платформа стала ключовим інструментом для оцінки продуктивності.

Підтвердження компромісу швидкість/точність, та аналіз симульованих результатів повністю підтвердив гіпотезу про обернену залежність між часом відгуку моделі та її точністю:

C_{base} (Базова Модель) показала найвищу точність (≈ 94.5%) за рахунок найдовшої затримки інференсу (≈ 185 мс). Така швидкість виявилася неприйнятною для критичних систем, що вимагають миттєвого реагування.

$C_{velocity}$ (Швидкісна Модель) **д**осягла мінімальної затримки (≈ 12 мс), але її нижча точність (≈ 82.0%) створює високий ризик помилкових негативів

(пропуску реальних загроз), що є критичним недоліком для безпекових застосувань.

Визначення оптимальної архітектури **(**C_{hybrid}**)** гібридна модель C_{hybrid} стала центральним відкриттям розділу, успішно демонструючи оптимальний баланс. Вона забезпечує високу точність (≈ 93.2%), залишаючись при цьому значно швидшою за базову модель (СЧДВ ≈ 45 мс), що робить її найкращим кандидатом для впровадження у виробниче середовище, де необхідна як швидкість, так і висока надійність виявлення.

У підсумку, третій розділ не лише завершив етап практичної розробки, але й надав чітке емпіричне обґрунтування для вибору моделі C_{hybrid} як основної архітектури для системи, що здатна ефективно протидіяти загрозам у режимі реального часу.

Висновки.

Дипломна робота була присвячена розробці та емпіричній оцінці архітектур машинного навчання для високошвидкісної класифікації даних у системах виявлення аномалій, де критично важливим є досягнення оптимального балансу між точністю прогнозування та часом відгуку. Поставлені цілі та завдання були повністю виконані.

1. Проаналізовано сучасні методи класифікації та архітектури моделей, що використовуються у високопродуктивних системах. Теоретично обґрунтовано необхідність компромісу між точністю (високі вимоги до обчислювальної потужності) та затримкою (критичні вимоги до швидкості).

2. Розроблено три моделі. Сформовано методологічні основи для створення трьох принципово різних моделей: C_{base} (орієнтована на максимальну точність), $C_{velocity}$ (орієнтована на мінімальну затримку) та C_{hybrid} (орієнтована на оптимальний баланс).

3. В результаті практичного моделювання було емпірично підтверджено, що модель C_{base} демонструє найвищу точність (≈ 94.5\%$) зі значною

затримкою ≈185 мс), тоді як C_{velocity} забезпечує мінімальний час відгуку ≈12 мс), але з неприпустимо низькою точністю (≈ 82.0\%$).

Центральним досягненням роботи є доказ ефективності архітектури C_{hybrid}. Ця модель досягла показника точності ≈ 93.2%, при цьому її середній час відгуку (≈ 45 мс) є більш ніж у чотири рази меншим, ніж у C_{base}. Це підтверджує, що C_{hybrid} є найбільш життєздатною архітектурою для систем реального часу.

Розроблений інтерактивний веб-дашборд успішно виконав функцію симуляційного полігону, дозволивши наочно візуалізувати часові та класифікаційні характеристики кожної архітектури.

Нами вперше в рамках даної роботи запропоновано та експериментально підтверджено ефективність гібридної архітектури класифікації, яка поєднує переваги високоточних, але повільних, та швидкісних, але менш надійних моделей, для досягнення оптимального співвідношення продуктивності у критичних системах.

Результати роботи мають пряме практичне застосування в галузях кібербезпеки, фінансового моніторингу та індустріального Інтернету речей (IIoT), де помилковий негатив або затримка можуть призвести до значних збитків чи порушення функціонування. Розроблена модель C_{hybrid} рекомендована до впровадження як основний класифікаційний модуль.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Коваленко, О. В. *Методи оптимізації архітектур глибокого навчання для систем реального часу*. Київ : Наукова думка, 2021. 352 с.

2. Петров, І. М., Сидоренко, А. К. Аналіз впливу квантизації моделей на швидкість інференсу у Edge-обчисленнях. *Вісник комп'ютерних наук та технологій*. 2023. № 4. С. 45–56.

3. Бондар, Р. С. *High-Speed Anomaly Detection: A Comprehensive Guide*. Medium. 2024. URL: https://www.google.com/search?q=https://medium.com/high-speed-detection

4. Гнатюк, П. А. Застосування ансамблевих методів для підвищення точності класифікації у промислових IoT-системах. *Матеріали II Міжнародної науково-практичної конференції «Інтелектуальні системи та безпека»*. Львів, 2023. С. 112–117.

5. Smith, J., Brown, L. Real-time machine learning for network intrusion detection. *IEEE Transactions on Cybernetics*. 2022. Vol. 52, № 6. P. 3901–3915.

6. ДСТУ 8302:2015. *Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання*. Київ : Держстандарт України, 2015. 16 с.

7. Мельник, В. І. *Основи глибокого навчання та нейронні мережі*. Харків : Фоліо, 2020. 410 с.

8. TensorFlow Lite. *Optimization techniques for mobile and edge devices*. Google Developers. 2024. URL: https://www.tensorflow.org/lite/performance/model_optimization

9. Мороз, М. С. Порівняльний аналіз швидкодії CNN та RNN архітектур для класифікації часових рядів. *Науковий збірник КНУТД*. 2023. Вип. 3 (97). С. 88–95.

10. *Розробка методів підвищення енергоефективності класифікаційних моделей*. Звіт про НДР (заключний). Керівник теми — І. П. Зайцев. Київ : Інститут кібернетики, 2022. 98 с. № держреєстрації 0120U101567.

11. Пат. 115230 Україна, МПК G06N 3/08. *Спосіб прискорення інференсу нейронної мережі*. Заявник та патентовласник: Технічний університет. № a202108743; заявл. 10.12.2021; опубл. 25.04.2022, Бюл. № 16.

12. Zhao, Q., & Li, Y. Survey on Ensemble Learning Methods for Imbalanced Data Classification. *Pattern Recognition Letters*. 2021. Vol. 147. P. 119–128.

13. Chen, B. *High-Performance Computing Architectures for Deep Learning Acceleration*. PhD dissertation. Massachusetts Institute of Technology, 2020. 210 p.

14. Radford, A., Kim, J. W., et al. *Learning Transferable Visual Models From Natural Language Supervision*. arXiv preprint arXiv:2103.00010. 2021. URL: https://arxiv.org/abs/2103.00010 (Дата звернення: 12.11.2024).

15. Закон України № 2269-VIII «Про основні засади забезпечення кібербезпеки України» від 17.06.2023 р. URL: https://zakon.rada.gov.ua/laws/show/2269-19

16. Ткаченко, Д. С. Методологія вибору оптимального порогу класифікації в умовах незбалансованих даних. *Проблеми оптимізації обчислювальних систем*. Київ : Видавничий дім «Академпрес», 2022. С. 78–92.

17. ISO/IEC 2382:2015. *Information technology — Vocabulary — Part 1: Fundamental terms*. International Organization for Standardization, 2015. 120 p.

18. Іваненко, О. М. Прискорення інференсу нейронних мереж за допомогою апаратних прискорювачів. *Тези доповідей XX Міжнародної наукової конференції молодих вчених*. Одеса, 2024. С. 34–35.

19. Шевчук, В. П. *Машинне навчання в системах безпеки*. Лекційний курс. КПІ ім. Ігоря Сікорського. 2023. URL: https://www.google.com/search?q=http://e-learning.kpi.ua/ml_security/

20. Goodfellow, I., Bengio, Y., & Courville, A. *Deep Learning*. MIT Press, 2016. 800 p.