

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
НАВЧАЛЬНО-НАУКОВИЙ ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки**

«На правах рукопису»
УДК _____

«До захисту допущено»
Завідувач кафедри
_____ Дмитро ЛАНДЕ
(підпис)

«_____» _____ 2025 р.

**Магістерська дисертація
на здобуття ступеня магістра
за освітньо-професійною програмою «Системи, технології та математичні
методи кібербезпеки»
спеціальності 125 «Кібербезпека та захист інформації»**

на тему: Механізм визначення дезінформації в мережі Інтернет

Виконав (-ла): здобувач вищої освіти II курсу, групи _____ ФБ-42мп _____
(шифр групи)

Рябко Дмитро Вячеславович
(прізвище, ім'я, по батькові)

(підпис)

Керівник _____ доцент кафедри ІБ, к.е.н. доц. Ткач Володимир Миколайович
(посада, науковий ступінь, вчене звання, прізвище, ім'я, по батькові) (підпис)

Рецензент доцент кафедри ММЗІ к.ф.-м.н Микола Олексійович Хмельницький
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище, ім'я, по батькові) (підпис)

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних посилань.
Здобувач вищої освіти _____

Київ – 2025 року

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
НАВЧАЛЬНО-НАУКОВИЙ ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

Рівень вищої освіти – другий (магістерський)
Спеціальність – 125 «Кібербезпека та захист інформації»
Освітньо-професійна програма «Системи, технології та математичні методи кібербезпеки»

ЗАТВЕРДЖУЮ
Завідувач кафедри

_____ Дмитро ЛАНДЕ
(підпис)

«___» _____ 2025 р.

ЗАВДАННЯ
на магістерську дисертацію здобувачу вищої освіти

Рябка Дмитра Вячеславовича
(прізвище, ім'я, по батькові)

1. Тема роботи «Механізм визначення дезінформації в мережі Інтернет»

керівник роботи

_____ Ткач В. М.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «4» листопада 2025 р. № 4797-с

2. Термін подання здобувачем вищої освіти роботи «___» грудня 2025 р.

3. Вихідні дані до роботи

Технічна документація Google Gemini API та посібники з Prompt Engineering;
специфікації фреймворку Next.js; методичні рекомендації щодо критеріїв
медіаграмотності та фактчекінгу; набори текстових даних для тестування

4. Зміст роботи

Аналіз проблем бінарної класифікації новинного контенту та обмежень
існуючих сигнатурних методів; формування набору критеріїв для семантичного
аналізу тексту; побудова математичної моделі скорингу (розрахунку рейтингу

довіри); розробка програмних компонентів системи: модуля алгоритмічної обробки запитів на базі LLM, підсистеми бекенд-оркестрації та веб-інтерфейсу користувача; експериментальне дослідження адекватності моделі на вибірці граничних випадків та порівняльний аналіз автоматичного скорингу з експертними оцінками.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо) Презентація до захисту магістерської дисертації.

6. Дата видачі завдання 10 вересня 2025

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Визначення напрямку МД	11.09.2025 - 18.09.2025	Виконано
2	Визначення теми МД	20.09.2025 - 29.09.2025	Виконано
3	Збір та аналіз літературних джерел	02.10.2025 - 17.10.2025	Виконано
4	Формування технічного завдання	18.10.2025 - 01.11.2025	Виконано
5	Розробка програмного коду	01.11.2025 - 24.11.2025	Виконано
6	Тестування та удосконалення	25.11.2025 - 07.12.2025	Виконано
7	Формування графічної частини (презентації)	08.12.2025	Виконано
8	Підготовка та передзахист МД	09.12.2025 - 12.12.2025	Виконано
9	Підготовка до захисту МД	01.12.2025 - 16.12.2025	Виконано

Здобувач вищої освіти

_____ (підпис)

Дмитро РЯБКО
(Власне ім'я, ПРІЗВИЩЕ)

Керівник роботи

_____ (підпис)

Володимир ТКАЧ
(Власне ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Обсяг роботи 61 сторінок, 23 ілюстрацій, 2 таблиць, 12 джерел літератури.

Об'єкт дослідження: процес ідентифікації та аналізу маніпулятивного контенту та дезінформації в мережі Інтернет.

Предмет дослідження: методи та засоби автоматизованої оцінки надійності веб-ресурсів із застосуванням гібридного підходу, що поєднує аналіз метаданих (OSINT) та семантичний аналіз тексту великими мовними моделями (LLM)

Мета дослідження: полягає у розробці комплексного програмного механізму визначення дезінформації в мережі Інтернет, що поєднує детермінований аналіз технічних параметрів веб-ресурсів та імовірнісний аналіз текстового контенту за допомогою штучного інтелекту

Методи дослідження: У роботі використано методи системного аналізу (для побудови архітектури), розвідки на основі відкритих джерел (OSINT), обробки природної мови (NLP), сентимент-аналізу, веб-скрапінгу та об'єктно-орієнтованого програмування.

Отримані результати: Розроблено веб-застосунок (на базі стеку Next.js та Gemini API), який на основі вхідної URL-адреси виконує комплексний аудит публікації. Програма автоматично збирає технічні дані (Whois, SSL, структура DOM) та використовує нейромережеві моделі для глибинного аналізу тексту за п'ятьма групами критеріїв: технічна надійність, репутація автора, верифікація фактів, логічна зв'язність та емоційна маніпулятивність. Реалізовано алгоритм розрахунку зваженого рейтингу довіри (Trust Score), що дозволяє користувачу миттєво оцінити ризик споживання недостовірної інформації.

Ключові слова: дезінформація, osint, великі мовні моделі (LLM), NLP, фактчекінг, веб-скрапінг, сентимент-аналіз, інформаційна безпека, Next.js.

ABSTRACT

Volume of the work is 61 pages, 23 illustrations, 2 tables, and 12 references.

The object of the research: the process of identification and analysis of manipulative content and disinformation on the Internet.

The subject of the research: methods and tools for the automated assessment of web resource reliability using a hybrid approach that combines metadata analysis (OSINT) and semantic text analysis by large language models (LLMs).

The aim of the research: is to develop a comprehensive software mechanism for detecting disinformation on the Internet, which combines deterministic analysis of web resource technical parameters and probabilistic analysis of textual content using artificial intelligence.

Research methods: The work utilizes methods of system analysis (for architecture design), Open Source Intelligence (OSINT), Natural Language Processing (NLP), sentiment analysis, web scraping, and Object-Oriented Programming (OOP).

The obtained results: A web application (based on the Next.js stack and Gemini API) has been developed, which performs a comprehensive audit of a publication based on an input URL. The software automatically collects technical data (Whois, SSL, DOM structure) and utilizes neural network models for deep text analysis across five criteria groups: technical reliability, author reputation, fact verification, logical coherence, and emotional manipulativeness. An algorithm for calculating a weighted Trust Score has been implemented, allowing the user to instantly assess the risk of consuming unreliable information.

Keywords: DISINFORMATION, OSINT, LARGE LANGUAGE MODELS (LLM), NLP, FACT-CHECKING, WEB SCRAPING, SENTIMENT ANALYSIS, INFORMATION SECURITY, NEXT.JS.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,	7
СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП	9
1 ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ ЗАСАДИ ВИЯВЛЕННЯ ДЕЗІНФОРМАЦІЇ ТА МАНІПУЛЯТИВНОГО КОНТЕНТУ В МЕРЕЖІ ІНТЕРНЕТ	11
1.1 Інформаційна безпека в умовах гібридних загроз: понятійний апарат та сучасні виклики	11
1.2 Класифікація технічних та семантичних ознак ненадійних веб-ресурсів.....	13
1.3 Аналіз існуючих методів та алгоритмів верифікації контенту	15
1.4 Обґрунтування доцільності використання гібридної архітектури	18
1.5 Постановка задачі та вимоги до програмного засобу	19
Висновок до розділу 1	21
2 ПРОЄКТУВАННЯ АРХІТЕКТУРИ ТА МЕТОДОЛОГІЇ СИСТЕМИ ВИЯВЛЕННЯ ДЕЗІНФОРМАЦІЇ	23
2.1 Розробка загальної архітектури системи	23
2.2 Методологія збору та попередньої обробки даних (OSINT)	25
2.3 Моделювання системи критеріїв оцінювання	27
2.4 Проєктування підсистеми інтелектуального аналізу контенту (AI-GATEWAY).....	29
2.5 Програмна реалізація механізмів агрегації та обробки результатів	32
2.6 Математична формалізація механізмів інтелектуального аналізу тексту	36
Висновок до розділу 2	36
3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ МЕТОДИКИ	38
3.1 Опис програмної реалізації та сценаріїв взаємодії з користувачем	38
3.2 Методика експериментального дослідження.....	47
3.3 Аналіз результатів експерименту	48
3.4 Аналіз помилок та обмежень системи	55
Висновок до розділу 3	58
ВИСНОВКИ.....	59
ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАНЬ.....	61

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

AI – Artificial Intelligence (штучний інтелект).

API – Application Programming Interface (інтерфейс прикладного програмування).

CSS – Cascading Style Sheets (каскадні таблиці стилів).

DOM – Document Object Model (об'єктна модель документа - структура HTML-сторінки, яку ми парсимо).

GPT – Generative Pre-trained Transformer (генеративний попередньо навчений трансформер; архітектура LLM).

HTML – HyperText Markup Language (мова розмітки гіпертексту).

HTTP – HyperText Transfer Protocol (протокол передачі гіпертексту).

HTTPS – HyperText Transfer Protocol Secure (захищений протокол передачі гіпертексту).

JSON – JavaScript Object Notation (текстовий формат обміну даними).

LLM – Large Language Model (велика мовна модель).

NER – Named Entity Recognition (розпізнавання іменованих сутностей; метод виділення імен, дат, локацій з тексту).

NLP – Natural Language Processing (обробка природної мови).

OSINT – Open Source Intelligence (розвідка на основі відкритих джерел).

REST – Representational State Transfer (архітектурний стиль взаємодії компонентів розподіленої програми в мережі).

SDK – Software Development Kit (набір засобів розробки; у роботі використано Gemini SDK).

SEO – Search Engine Optimization (пошукова оптимізація).

SSL – Secure Sockets Layer (криптографічний протокол для безпечного зв'язку).

TLS – Transport Layer Security (протокол захисту транспортного рівня, наступник SSL).

UI – User Interface (інтерфейс користувача).

URL – Uniform Resource Locator (уніфікований покажчик ресурсу; веб-адреса).

UX – User Experience (користувацький досвід).

WHOIS – Мережевий протокол прикладного рівня для отримання реєстраційних даних про власників доменних імен та IP-адрес.

Терміни:

Клікбейт (Clickbait) – пейоративний термін, що описує веб-контент, метою якого є отримання доходу від онлайн-реклами за рахунок низької якості або сенсаційних заголовків.

Тайпосквотінг (Typosquatting) – форма кіберсквотінгу, що полягає в реєстрації доменних імен, близьких за написанням до адрес популярних сайтів, з розрахунком на помилки користувачів при введенні адреси.

Фейк (Fake news) – інформаційне повідомлення, що містить частково або повністю недостовірну інформацію.

Фактчекінг (Fact-checking) – перевірка фактичних тверджень у нехудожньому тексті з метою визначення їх правдивості та правильності.

ВСТУП

Актуальність роботи. Сучасний інформаційний простір характеризується безпрецедентною швидкістю поширення даних, що створює сприятливі умови для масових маніпуляцій та дезінформаційних кампаній. Традиційні методи фактчекінгу, які покладаються на ручну перевірку експертами або статичні «чорні списки» доменів, втрачають ефективність проти технологічно вдосконалених фейків, що генеруються автоматично та адаптуються під цільову аудиторію. Використання великих мовних моделей (LLM) та методів розвідки на основі відкритих даних (OSINT) дозволяє не лише автоматизувати процес верифікації джерел, але й проводити глибинний семантичний аналіз контенту, виявляючи приховані логічні хиби та емоційні маніпуляції.

Мета дослідження полягає в розробці комплексного програмного механізму визначення дезінформації в мережі Інтернет, що поєднує детермінований аналіз технічних параметрів веб-ресурсів та імовірнісний аналіз текстового контенту за допомогою штучного інтелекту.

Завдання дослідження:

1. Провести аналіз існуючих підходів до виявлення фейкових новин та визначити обмеження сучасних автоматизованих систем.
2. Розробити систему критеріїв оцінювання достовірності інформації, згруповану за п'ятьма векторами: технічні показники, репутація автора, верифікація фактів, логічна якість тексту та емоційне забарвлення.
3. Запропонувати та програмно реалізувати гібридну архітектуру веб-застосунку, що інтегрує методи OSINT (Whois, парсинг метаданих) та NLP-сервіси (Generative AI) для розрахунку інтегрального рейтингу довіри.
4. Експериментально перевірити ефективність розробленого механізму на вибірці реальних новинних та дезінформаційних матеріалів, оцінити точність роботи алгоритмів.

Методи дослідження. У роботі використано методи системного аналізу, розвідки на основі відкритих джерел (OSINT), обробки природної мови (NLP), сентимент-аналізу, веб-скрапінгу та об'єктно-орієнтованого програмування.

Наукова новизна полягає у вдосконаленні методології автоматизованого виявлення дезінформації шляхом створення гібридної моделі оцінювання, яка, на відміну від існуючих аналогів, одночасно враховує технічні атрибути джерела, цифровий слід автора та семантичні патерни маніпуляції в тексті.

Практичне значення. Практичне значення роботи полягає у створенні діючого веб-інструменту, який дозволяє користувачам у режимі реального часу оцінювати надійність інформаційних джерел, що сприяє підвищенню рівня медіаграмотності та зменшенню впливу дезінформації на суспільство

1 ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ ЗАСАДИ ВИЯВЛЕННЯ ДЕЗІНФОРМАЦІЇ ТА МАНІПУЛЯТИВНОГО КОНТЕНТУ В МЕРЕЖІ ІНТЕРНЕТ

1.1 Інформаційна безпека в умовах гібридних загроз: понятійний апарат та сучасні виклики

Сучасний етап розвитку інформаційного суспільства характеризується не лише експоненціальним зростанням обсягів даних, але й фундаментальною зміною парадигми споживання контенту. Тотальна цифровізація комунікаційних процесів призвела до стирання кордонів між фізичним та віртуальним простором, перетворивши інфосферу на стратегічний домен протистояння. У цій новій реальності інформація виступає подвійним агентом: як стратегічний ресурс для розвитку економіки знань і як летальне озброєння в умовах гібридних війн.

У контексті національної та корпоративної безпеки ключового значення набуває поняття інформаційної стійкості. Це не просто здатність системи технічно захищати дані, а спроможність інформаційного простору зберігати свою цілісність, достовірність та функціональність під впливом деструктивних інформаційно-психологічних операцій (ІПСО).

Центральним елементом таких загроз є дезінформація. Це не просто брехня, а верифіковано хибна інформація, яка створюється, поширюється та використовується з чітким наміром заподіяння шкоди окремим особам, соціальним групам, організаціям або державам. Для побудови ефективної системи захисту слід чітко розмежовувати поняття в рамках концепції «Інформаційного розладу»:

- 1. Мізінформація (Misinformation):** Помилкова інформація, що поширюється без злого умислу. Прикладом може слугувати помилка журналіста, неправильний переклад терміну або поширення застарілих даних користувачами, які щиро вірять у їх правдивість. Хоча така інформація є хибною, вона не має на меті дестабілізацію.

2. **Дезінформація (Disinformation):** Свідомо сфабрикований контент, що створюється для маніпуляції громадською думкою, посіяння паніки або дискредитації опонентів. Це спланована діяльність, яка часто має системний характер і фінансується зацікавленими сторонами.
3. **Шкідлива інформація (Malinformation):** Правдива інформація, яка використовується для завдання шкоди. Сюди відносяться витік приватного листування, розголошення персональних даних або публікація компрометуючих фактів, вирваних з контексту.

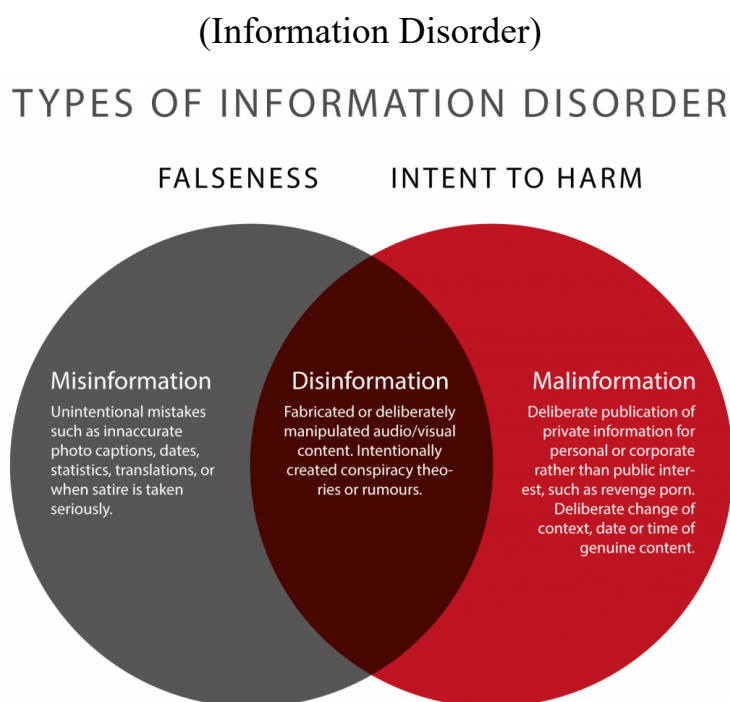


Рисунок 1.1 - Концептуальна модель інформаційних розладів

Феномен «постправди» докорінно змінив механізми сприйняття новин. У цій парадигмі об'єктивні факти стають менш важливими для формування громадської думки, ніж звернення до емоцій та особистих переконань. Це створює сприятливе середовище для застосування когнітивних атак - цілеспрямованого впливу на свідомість реципієнта з метою зміни його поведінкової моделі, цінностей або сприйняття реальності.

Основними векторами атак у веб-середовищі є:

- **Соціальна інженерія:** Використання психологічних маніпуляцій та когнітивних упереджень для спонукання користувачів до виконання певних дій (клік по посиланню, репост фейку).
- **Астротурфінг:** Штучне створення ілюзії масової громадської підтримки певної ідеї за допомогою бот-мереж та скоординованих троль-ферм. Це спотворює реальну картину суспільних настроїв.
- **Ехо-камери та Бульбашки фільтрів:** Алгоритмічне замикання користувача в інформаційному полі, що лише підтверджує його існуючі упередження. Це унеможливорює критичне сприйняття альтернативних точок зору та робить аудиторію вразливою до радикалізації.
- **Діпфейки:** Використання генеративного ШІ для створення гіперреалістичних підробок аудіо та відео, що становить загрозу нового рівня для верифікації контенту.

1.2 Класифікація технічних та семантичних ознак ненадійних веб-ресурсів

Для автоматизації процесу виявлення дезінформації необхідно перейти від абстрактних понять до формалізованих ознак (features), які можна алгоритмічно виявити та виміряти. Комплексний аналіз передбачає розподіл цих ознак на дві великі групи: технічні (атрибутивні) та семантичні (контентні).

1.2.1 Технічні індикатори (Metadata Features)

Технічні індикатори базуються на аналізі метаданих веб-ресурсу та його інфраструктури. Вони не вимагають глибокого розуміння тексту, тому є ефективними для первинної фільтрації («грубого сита») підозрілих ресурсів:

- **Доменна репутація:**

Вік домену: Статистично підтверджено, що сайти, створені менше 30-60 днів тому, мають найвищий ризик використання у фішингових кампаніях та дезінформаційних вкидах («сайти-одноденки»).

Тайпосквотінг (Typosquatting): Зловмисники реєструють домени, візуально схожі на відомі бренди (наприклад, bbc-news.net замість bbc.com, заміна літери l на цифру 1 або використання кириличних омографів). Виявлення таких патернів є сильним сигналом шахрайства.

Доменна зона: Використання дешевих або безкоштовних доменних зон (.xyz, .top, .tk), які часто використовуються ботнетами.

- **Мережева інфраструктура:**

SSL/TLS сертифікати: Відсутність шифрування або використання безкоштовних короткострокових сертифікатів (наприклад, Let's Encrypt) на "корпоративних" сайтах може бути індикатором ризику.

Whois дані: Приховування даних реєстранта (Whois Privacy Protection) є нормою для приватних осіб, але підозрілим фактором для новинних агентств, які мають бути прозорими.

- **Структура веб-сторінки (DOM-аналіз):**

Легітимні медіа дотримуються стандартів прозорості. Відсутність сторінок «Про нас» (About Us), «Контакти», «Редакційна політика» або їх шаблонне наповнення (Lorem Ipsum) є критичним маркером ненадійності. Також аналізується наявність посилань на соціальні мережі та їх валідність..

1.2.2 Лінгвістичні та психолінгвістичні індикатори

Ця група ознак стосується безпосередньо тексту публікації і вимагає застосування методів обробки природної мови (NLP):

- **Лексична різноманітність**: кісна журналістика характеризується багатим словниковим запасом. Фейкові новини та згенеровані ботами тексти часто мають низький коефіцієнт TTR (Type-Token Ratio), повторюваність тез та примітивну структуру речень.
- **Сентимент-аналіз (Sentiment Analysis)**: Дезінформація часто має екстремальне емоційне забарвлення (гнів, страх, огида) для вірального

поширення. Використання «мови ворожнечі» (Hate Speech) та маніпулятивних тригерів у заголовках.

- **Синтаксична складність:** Використання імперативних форм дієслів («поширте», «знайте правду»), великої кількості знаків оклику та капіталізації (CAPS LOCK).

1.3 Аналіз існуючих методів та алгоритмів верифікації контенту

Історія розвитку систем автоматизованої перевірки фактів (Automated Fact-Checking) пройшла шлях від простих евристик до складних неймережевих моделей. На сьогодні можна виділити три основні підходи.

1.3.1 Експертні системи та методи на основі баз знань (Knowledge-based)

Цей підхід полягає у зіставленні тверджень у тексті зі структурованими базами знань (Knowledge Graphs), такими як DBpedia, Wikidata або спеціалізованими базами фактчекінгу.

- *Переваги:* Висока точність при перевірці простих фактів (дати, імена, посади).
- *Недоліки:* Неможливість перевірки суб'єктивних суджень, прогнозів або подій, що сталися в реальному часі і ще не внесені до баз знань (проблема «холодного старту»).

1.3.2 Статистичні методи машинного навчання (Machine Learning)

Другим еволюційним етапом у розвитку систем детекції дезінформації стало застосування класичних алгоритмів машинного навчання. На відміну від експертних систем, які працюють за жорстко заданими правилами, методи Machine Learning (ML) намагаються виявити приховані закономірності у масивах даних шляхом статистичного аналізу.

У контексті задачі класифікації текстів на «правдиві» та «фейкові», найбільш поширеним підходом є навчання з учителем (Supervised Learning). Цей процес передбачає наявність великого розміченого датасету, де кожна новина вже має мітку класу (True/Fake). На етапі попередньої обробки даних (preprocessing)

текст перетворюється на числовий вектор. Для цього застосовуються такі методи конструювання ознак (Feature Engineering):

- **«Мішок слів» (Bag-of-Words):** Підхід, при якому враховується лише наявність або частота певних слів у документі, ігноруючи їх порядок та граматичні зв'язки.
- **TF-IDF (Term Frequency-Inverse Document Frequency):** Статистична міра, що дозволяє оцінити важливість слова в контексті документа відносно всієї колекції текстів. Це допомагає відфільтрувати загальноживані слова (стоп-слова) і виділити специфічні терміни.
- **N-грами:** Аналіз послідовностей з N слів, що йдуть підряд, що дозволяє частково врахувати локальний контекст фрази.

Серед алгоритмів, що показали найбільшу ефективність у цій парадигмі, варто виділити:

1. **Наївний байєсівський класифікатор (Naive Bayes):** Базується на теоремі Байєса з припущенням про незалежність ознак. Незважаючи на свою простоту, він демонструє високу швидкість та прийнятну точність при роботі з текстовими даними, особливо у завданнях спам-фільтрації.
2. **Метод опорних векторів (Support Vector Machine - SVM):** Алгоритм, який шукає гіперплощину в багатовимірному просторі ознак, що найкращим чином розділяє класи об'єктів. SVM добре працює з даними високої розмірності, що є характерним для текстових корпусів.
3. **Випадковий ліс (Random Forest) та Градієнтний бустинг (Gradient Boosting):** Ансамблеві методи, які будують множину дерев рішень і агрегують їхні результати. Це дозволяє зменшити ризик перенавчання моделі та підвищити стабільність прогнозування.

Переваги підходу: Головною перевагою класичного ML є інтерпретованість результатів (можна зрозуміти, які саме слова вплинули на рішення) та відносно низькі вимоги до обчислювальних ресурсів.

Недоліки та обмеження: Ключовою проблемою статистичних методів є їхня неспроможність зрозуміти семантику (зміст) тексту. Моделі спираються на

наявність специфічних ключових слів. Якщо фейкова новина написана професійною мовою без явних лексичних маркерів, ML-класифікатор, найімовірніше, помилиться. Крім того, ці моделі погано адаптуються до нових тем (Domain Adaptation): класифікатор, навчений на політичних новинах, буде неефективним при аналізі медичних фейків без додаткового перенавчання.

1.3.3 Глибоке навчання та Великі мовні моделі (Deep Learning & LLM)

Сучасний етап розвитку технологій обробки природної мови (NLP) пов'язаний з переходом від статистичного аналізу слів до моделювання мовних залежностей за допомогою глибоких нейронних мереж. Цей перехід став можливим завдяки появі архітектури **Трансформерів (Transformers)** та механізму уваги (Attention Mechanism), які дозволили моделям враховувати глобальний контекст всього документа, а не лише сусідніх слів.

Великі мовні моделі (Large Language Models - LLM), такі як BERT (Bidirectional Encoder Representations from Transformers), RoBERTa, та генеративні моделі серії GPT (Generative Pre-trained Transformer), змінили правила гри у сфері виявлення дезінформації.

- **Дискримінативні моделі (наприклад, BERT):** Здатні аналізувати текст у двох напрямках одночасно (зліва направо і справа наліво), що дозволяє їм розуміти складні семантичні конструкції, іронію, сарказм та метафори, які часто використовуються в маніпулятивних текстах.
- **Генеративні моделі (наприклад, GPT- 4, Gemini):** Володіють здатністю до логічного висновування (Reasoning). Вони можуть не просто класифікувати текст, а й пояснити, чому він виглядає підозрілим, виявити логічні протиріччя всередині статті або порівняти твердження із загальновідомими фактами, закладеними в їх параметрах під час тренування.

Особливої уваги заслуговує здатність сучасних LLM до навчання з декількох прикладів (**Few-Shot Learning**) або навіть без прикладів (**Zero-Shot Learning**). Це означає, що для налаштування системи на пошук нового типу фейків не потрібно збирати тисячі розмічених прикладів - достатньо сформулювати правильний запит (промпт) природною мовою.

Переваги: Здатність проводити глибокий семантичний аналіз, що наближається до людського рівня розуміння тексту. Можливість виявляти складні наративні маніпуляції, які не містять специфічних ключових слів.

Недоліки: Основним бар'єром для впровадження є висока обчислювальна складність та вартість експлуатації. Крім того, існує проблема «галюцинацій», коли генеративна модель може впевнено стверджувати неправдиві факти. Також залишається актуальною проблема «чорної скриньки» (Black Box problem) - процес прийняття рішення всередині нейромережі з мільярдами параметрів є непрозорим, що ускладнює аудит системи.

1.4 Обґрунтування доцільності використання гібридної архітектури

Провівши детальний порівняльний аналіз існуючих підходів, можна зробити висновок, що жоден із них в ізольованому вигляді не здатен вирішити проблему масового поширення дезінформації з достатнім рівнем надійності та ефективності. Існує фундаментальне протиріччя між точністю аналізу та швидкістю обробки даних:

- **Суто технічні методи (OSINT)** працюють миттєво і дешево, але генерують велику кількість помилок першого роду (False Positives) на нових легітимних ресурсах і пропускають якісно підготовлені пропагандистські матеріали.
- **Суто семантичні методи (LLM)** забезпечують високу якість аналізу, але є економічно недоцільними для перевірки кожного існуючого в інтернеті посилання через високу вартість API-запитів та значні затримки (latency).

Виходячи з цього, найбільш перспективним та науково обґрунтованим напрямком є розробка **гібридної багаторівневої системи**. Така архітектура базується на принципі синергії детермінованих та імовірнісних методів, реалізуючи концепцію «Глибокого захисту» (**Defense in Depth**).

Запропонований підхід передбачає поетапну фільтрацію контенту:

1. **Перший рівень (Low-Level / Rule-based):** Це рівень "грубого відсіву". Тут застосовуються швидкі алгоритми перевірки метаданих, сертифікатів безпеки, віку домену та наявності у чорних списках. Цей етап дозволяє відсіяти до 70-80% примітивних загроз (фішинг, спам-ферми, автоматично згенеровані сайти) за частки секунди з мінімальними витратами ресурсів.
2. **Другий рівень (High-Level / AI-based):** Ресурси, які успішно пройшли первинну технічну перевірку, підлягають глибокому інтелектуальному аналізу. На цьому етапі підключаються нейромережеві моделі для оцінки емоційності, логічної зв'язності та стилістики тексту.

Така гібридизація дозволяє досягти оптимального балансу між продуктивністю системи (Throughput) та точністю детекції (Accuracy), роблячи рішення придатним для використання в реальних умовах високого навантаження.

1.5 Постановка задачі та вимоги до програмного засобу

Враховуючи актуальність проблеми та недоліки існуючих рішень, основною метою даної роботи є створення програмного комплексу у вигляді веб-застосунку для автоматизованого аудиту інформаційних ресурсів.

Для досягнення поставленої мети система повинна задовольняти комплекс функціональних та нефункціональних вимог, сформульованих на основі аналізу предметної області:

1. Функціональні вимоги:

- **Комплексність аналізу:** Система повинна забезпечувати одночасну оцінку надійності джерела (Source credibility) на основі метаданих та достовірності змісту (Content credibility) на основі аналізу тексту.
- **Мультиmodalність вхідних даних:** Підтримка обробки як прямих посилань на статті (URL), так і аналізу доменних імен.
- **Інтерактивна візуалізація:** Результати аналізу мають бути представлені у вигляді, доступному для розуміння користувачем без спеціальної технічної підготовки. Це передбачає розрахунок інтегрального рейтингу довіри, використання кольорової індикації рівнів ризику та надання деталізованих пояснень щодо кожного виявленого недоліку.

2. Нефункціональні вимоги:

- **Модульність та розширюваність:** Архітектура системи має бути побудована за принципом "плагінів", що дозволить легко додавати нові критерії перевірки (наприклад, модуль аналізу зображень) без необхідності переписання ядра системи.
- **Масштабованість:** Враховуючи потенційно високе навантаження, система повинна підтримувати горизонтальне масштабування, наприклад, через використання мікросервісних патернів або безсерверних обчислень (Serverless functions).
- **Швидкодія:** Час повної перевірки одного ресурсу не повинен перевищувати психологічно комфортний для користувача поріг (в середньому 3-5 секунд), що вимагає використання асинхронних методів обробки даних.

З точки зору технологічної реалізації, для побудови клієнт-серверної архітектури доцільно використовувати сучасні веб-фреймворки, такі як **Next.js**. Вибір цієї технології зумовлений підтримкою серверного рендерингу (SSR) та вбудованими можливостями створення API-маршрутів, що спростує

розгортання та підтримку застосунку. В якості інтелектуального ядра системи оптимальним рішенням є інтеграція API сучасних генеративних моделей (наприклад, сімейства Gemini або GPT), що дозволить делегувати складні задачі семантичного аналізу хмарним обчислювальним потужностям, забезпечуючи найвищу якість розпізнавання патернів дезінформації.

Висновок до розділу 1

У першому розділі проведено теоретичний аналіз проблематики виявлення дезінформації.

1. Встановлено, що в умовах сучасних гібридних конфліктів та ери «постправди» дезінформація трансформувалася зі спорадичного явища у системний інструмент впливу. Вона еволюціонувала від простих текстових фейків до складних когнітивних атак, що використовують психологічні вразливості аудиторії, що робить традиційні методи ручної модерації неефективними..
2. Проведено систематизацію та формалізацію ознак ненадійних веб-ресурсів. Виділено дві ключові групи індикаторів: технічні (пов'язані з інфраструктурою, метаданими домену та структурою сторінки) та семантичні (пов'язані зі стилістикою, логічною побудовою тексту та емоційним забарвленням)
3. Здійснено критичний порівняльний аналіз існуючих методів верифікації контенту. Виявлено суттєві обмеження класичних алгоритмів машинного навчання (залежність від навчальних даних, контекстна сліпота) та експертних систем. Обґрунтовано тезу про те, що оптимальним шляхом розвитку є синергія методів OSINT (розвідки на основі відкритих джерел) та новітніх великих мовних моделей (LLM) у рамках єдиної гібридної архітектури.

4. На основі проведеного аналізу сформульовано чіткі вимоги до розроблюваного програмного засобу, визначено його архітектурні принципи (модульність, масштабованість, глибинний захист) та технологічний стек, що створює теоретичне підґрунтя для практичної реалізації системи у наступних розділах роботи..

2 ПРОЄКТУВАННЯ АРХІТЕКТУРИ ТА МЕТОДОЛОГІЇ СИСТЕМИ ВИЯВЛЕННЯ ДЕЗІНФОРМАЦІЇ

2.1 Розробка загальної архітектури системи

Проєктування архітектури програмного засобу є фундаментальним етапом розробки, який визначає здатність системи до масштабування, підтримки та подальшої модернізації. Враховуючи вимоги до швидкодії та необхідність інтеграції різнорідних сервісів (зовнішні API, бази даних, нейромережі), для вирішення завдання автоматизованого аудиту веб-ресурсів було спроектовано розподілену сервіс-орієнтовану архітектуру (SOA).

Такий підхід, на відміну від монолітної архітектури, дозволяє забезпечити гнучкість системи, слабку зв'язність компонентів, незалежність модулів перевірки та можливість горизонтального масштабування при зростанні навантаження. Крім того, використання SOA дозволяє ізолювати логіку роботи з нестабільними зовнішніми джерелами, запобігаючи каскадним збоям усієї системи.

Система базується на трьох логічних рівнях, кожен з яких виконує чітко окреслений набір функцій:

2.1.1 Рівень представлення (Presentation Layer)

Цей рівень відповідає за взаємодію з кінцевим користувачем. Оскільки цільовою аудиторією є широке коло інтернет-користувачів, до інтерфейсу висуваються вимоги максимальної спрощеності та адаптивності. Основні функції рівня:

1. **Прийом та попередня валідація даних:** Забезпечення перевірки коректності введеного URL-адреси (відповідність синтаксичним стандартам RFC 3986) ще до моменту відправки запиту на сервер. Це дозволяє зменшити навантаження на сервер, відсіюючи завідомо некоректні запити.
2. **Візуалізація процесу:** Оскільки повний аудит ресурсу може займати від 3 до 10 секунд (залежно від швидкості відповіді зовнішніх серверів),

критично важливим для UX є відображення проміжних станів системи. Інтерфейс інформує користувача про поточний етап: «Ініціалізація», «Збір метаданих Whois», «Завантаження контенту», «Семантичний аналіз AI».

3. **Інтерпретація результатів:** Перетворення складних числових метрик та технічних даних у зрозумілі графічні елементи. Використовуються кольорові індикатори ризику, діаграми прогресу та інтерактивні підказки, що дозволяє користувачу миттєво оцінити рівень загрози).

2.1.2 Рівень бізнес-логіки та оркестрації (Business Logic Layer)

Це ядро системи, яке реалізує основні алгоритми аналізу та координує роботу всіх допоміжних модулів. Для оптимізації часу виконання запиту, замість послідовного (монолітного) виконання, використано архітектурний патерн «Конвеєр» з елементами асинхронної обробки подій.

- **Потік А:** Виконує запити до реєстрів доменних імен (Whois) для визначення віку та власника домену, а також здійснює перевірку наявності та валідності SSL-сертифікатів. Цей потік є найшвидшим і дозволяє виявити грубі технічні порушення..
- **Потік Б (Основний):** Відповідає за емуляцію браузера, завантаження HTML-контенту сторінки, його очищення від «інформаційного шуму» (реклами, навігаційних меню, скриптів) та підготовку чистого тексту для подальшого аналізу.
- **Потік В (Інтелектуальний):** Передає підготовлені та нормалізовані дані до модулів штучного інтелекту, формує контекстні промпти та обробляє отримані інсайти у вигляді структурованих даних.

Така архітектура дозволяє уникнути блокування системи: якщо один із зовнішніх сервісів не відповідає, система продовжує роботу за іншими критеріями.

2.1.3 Рівень даних та зовнішніх інтеграцій (Data Layer)

Система не зберігає контент перевірених сторінок (для уникнення порушення авторських прав), але тимчасово кешує результати аналізу для оптимізації повторних запитів. На цьому рівні відбувається взаємодія із зовнішніми API (генеративні моделі, бази репутації IP-адрес, DNS-сервери).

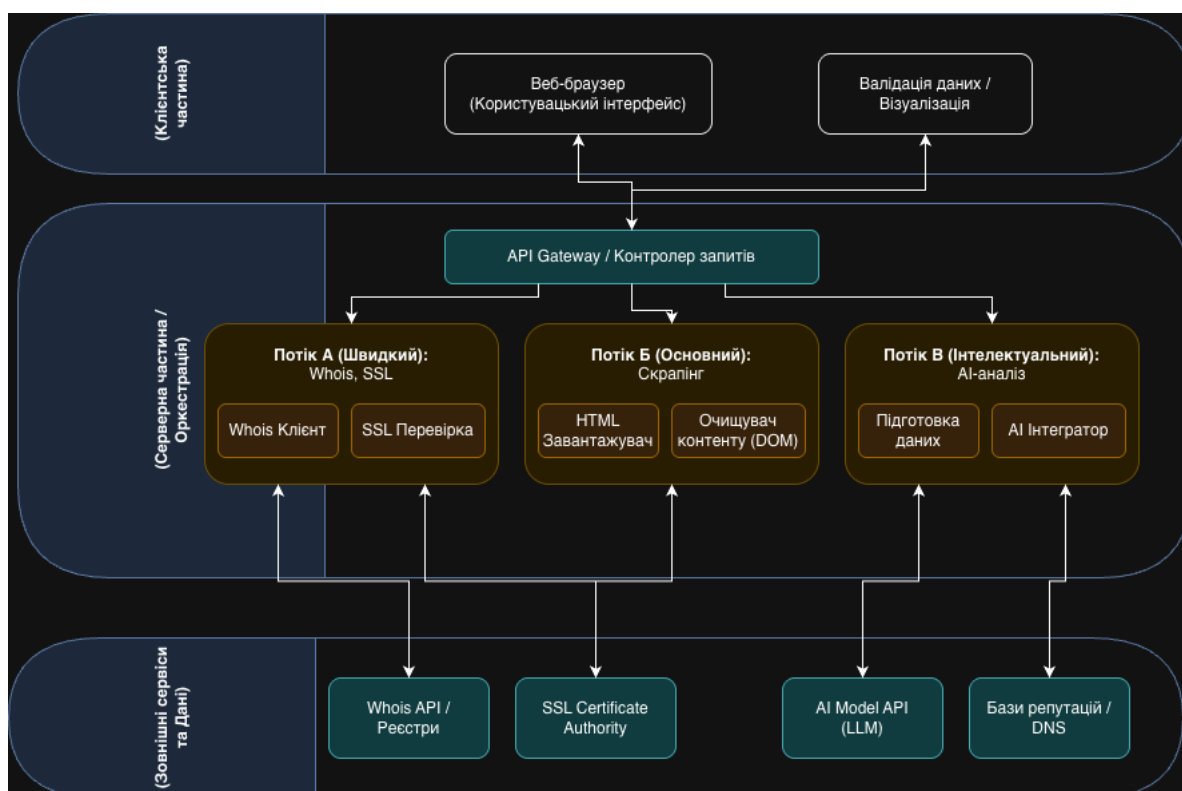


Рисунок 2.1 - Структурна схема архітектури програмного засобу

2.2 Методологія збору та попередньої обробки даних (OSINT)

Ефективність та точність виявлення дезінформації напряму залежить від якості та повноти вхідних даних. У рамках роботи було розроблено алгоритм інтелектуального скрапінгу (Web Scraping), який дозволяє отримувати дані навіть із сучасних динамічних веб-сайтів (SPA), імітуючи поведінку реального користувача (User-Agent spoofing, обробка cookies) для обходу базових систем захисту від автоматизованого збору даних..

2.2.1 Алгоритм вилучення значущого контенту

Сучасні веб-сторінки перенасичені технічною інформацією, яка не несе семантичного сенсу для аналізу надійності, але створює значний шум. До такої інформації належать коди аналітики, рекламні трекери, навігаційні панелі, віджети соціальних мереж, CSS-стили та JavaScript-код. Пряма передача "сирого" HTML-коду до нейромережі призвела б до перевищення ліміту токенів (context window limit), зашумлення контексту та, як наслідок, до помилкових висновків моделі.

Розроблено евристичний метод очищення (Content Extraction Heuristic), який працює за наступною логікою:

1. **Побудова DOM-дерева:** Завантаження структури сторінки.
2. **Фільтрація вузлів:** Автоматичне видалення елементів `<script/>`, `<style/>`, `<nav/>`, `<footer/>`
3. **Ідентифікація основного тіла:** Пошук контейнера з найбільшою щільністю текстових символів (Text Density Analysis). Це дозволяє точно виділити статтю навіть на сайтах зі складною версткою.
4. **Нормалізація тексту:** Видалення зайвих пробілів, переносів рядків та спецсимволів для формування єдиного текстового масиву.

2.2.2 Аналіз цифрового сліду домену

Методологія OSINT (Open Source Intelligence) передбачає збір інформації з відкритих реєстрів. Система автоматично перевіряє:

- **Вік домену:** Цей показник є одним з найсильніших індикаторів надійності. Статистично підтверджено, що сайти, створені менше 30–60 днів тому, мають у 8 разів вищу ймовірність бути джерелом фейків, фішингу або частиною бот-мережі. Легітимні медіа-ресурси зазвичай мають багаторічну історію існування.
- **Приватність реєстранта:** Хоча захист персональних даних є стандартом (GDPR), повне приховування даних про власника ресурсу через сервіси

анонімізації (наприклад, "Privacy Protect", "WhoisGuard") у комбінації з іншими негативними факторами може свідчити про свідоме намагання уникнути юридичної відповідальності за поширений контент.

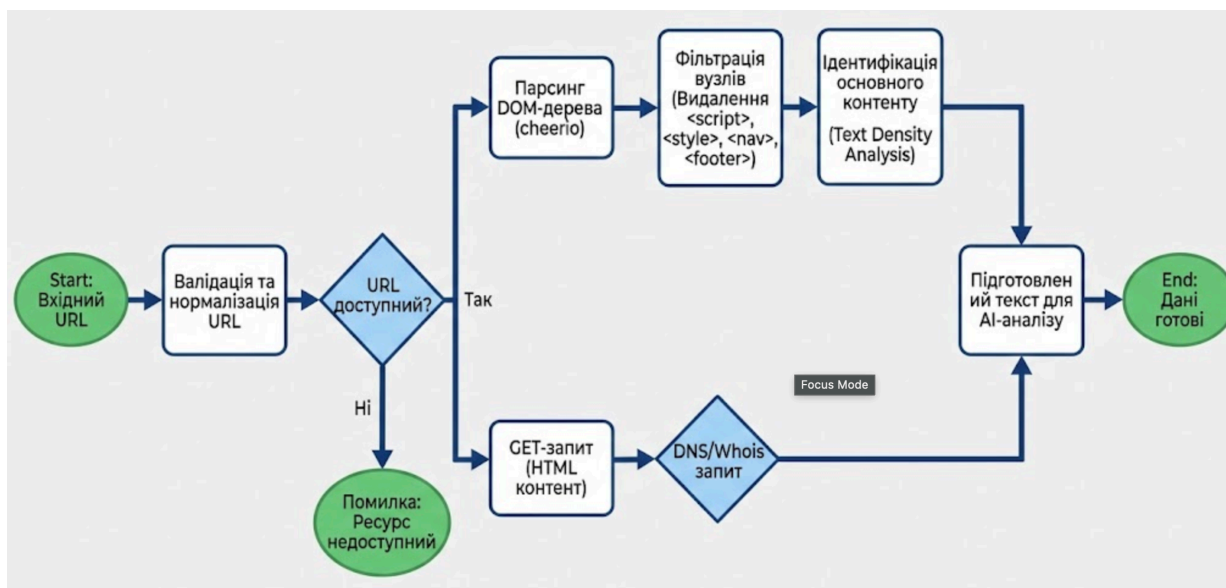


Рисунок 2.2 - Алгоритм збору та нормалізації даних веб-ресурсу

2.3 Моделювання системи критеріїв оцінювання

Для отримання об'єктивної оцінки було розроблено багатофакторну модель, яка декомпозує поняття "довіри" на п'ять незалежних векторів (груп критеріїв). Кожен вектор оцінює специфічний аспект інформаційної безпеки.

2.3.1 Група "Технічна надійність та Прозорість"

Ця група базується на детермінованих правилах. Вона відповідає на питання: «Чи є цей ресурс технічно безпечним та прозорим?».

- **Захищеність з'єднання:** Наявність чинного SSL/TLS-сертифікату є базовим мінімумом для будь-якого сучасного ресурсу. Використання незахищеного протоколу HTTP свідчить про занедбаність сайту або його низький технічний рівень.

- **Політики прозорості:** Легітимні медіа завжди декларують свою відповідальність. Здійснюється автоматичний пошук сторінок «Про нас», «Редакційна політика», «Контакти», «Умови використання». Їх повна відсутність є маркером анонімних «зливних бачків».
- **Репутація реєстратора:** Перевірка провайдера доменних послуг (деякі реєстратори відомі лояльністю до спамерів).

2.3.2 Група "Ідентифікація та репутація автора"

Дезінформація часто поширюється анонімно або під псевдонімами.

Система аналізує:

- **Цифровий слід автора:** Здійснюється пошук перехресних згадок імені автора в інших джерелах. Якщо автор пише на серйозні теми (геополітика, економіка), але не має профілів у професійних мережах (LinkedIn, Muckrack) або історії публікацій в інших виданнях – це сигнал тривоги.
- **Афіліації:** Виявлення зв'язків автора з відомими організаціями, університетами, аналітичними центрами чи медіа-холдингами, що додає ваги його твердженням.

2.3.3 Група "Верифікація фактів та джерел"

Найскладніший етап, що вимагає семантичного аналізу.

- **Перехресна перевірка:** Система сканує текст на наявність гіперпосилань або згадок авторитетних першоджерел (офіційні сайти державних структур, відомі міжнародні інформаційні агентства, наукові публікації).
- **Аналіз конкретики:** Використання методів розпізнавання іменованих сутностей для підрахунку конкретних фактологічних даних: дат, імен, назв організацій та геолокацій. Тексти, насичені абстракціями («в одній країні», «відомі вчені кажуть», «джерела повідомляють»), отримують низький бал верифікації через неможливість перевірки інформації.

2.3.4 Група "Логічна зв'язність тексту"

Пропаганда часто використовує логічні помилки для маніпуляції.

Система аналізує текст на наявність:

- **Логічних розривів:** Відсутність причинно-наслідкових зв'язків між аргументом та висновком.
- **Суперечливих тверджень:** Ситуація, коли початок статті суперечить її висновкам.

2.3.5 Група "Емоційно-маніпулятивний вплив"

Мета фейку - викликати емоцію, яка вимкне критичне мислення

- **Сентимент-аналіз:** Вимірювання рівня агресії, страху чи гніву в тексті.
- **Тригер-слова:** Пошук маркерів клікбейту ("Шок", "Сенсація", "Терміново") та мови ворожнечі.



Рисунок 2.3 - Ієрархічна структура критеріїв оцінювання

2.4 Проєктування підсистеми інтелектуального аналізу контенту (AI-Gateway)

Інтеграція технологій штучного інтелекту є ключовою інновацією даної роботи, що дозволяє автоматизувати складні когнітивні процеси, які раніше

вимагали залучення кваліфікованих експертів-лінгвістів. Підсистема AI-Gateway виступає інтелектуальним проміжним шаром між бізнес-логікою застосунку та хмарним API генеративних моделей (у даній реалізації обрано модель Google Gemini 2.5 Flash).

Вибір саме цієї моделі обумовлений її унікальними архітектурними особливостями, зокрема розширеним контекстним вікном (до 1 мільйона токенів). Це дозволяє системі аналізувати великі аналітичні статті, лонгріди та навіть документи без необхідності їх фрагментації, що часто призводить до втрати загального контексту та погіршення якості аналізу.

2.4.1 Методологія інженерії промптів (Advanced Prompt Engineering)

Ефективність роботи великих мовних моделей (LLM) критично залежить від якості вхідних інструкцій. У роботі застосовано комплексний підхід до формування промптів, який базується на кількох евристиках:

1. **Паттерн «Персона»:** Для налаштування моделі на специфічний тип задач їй задається рольова модель. Інструкція формулюється таким чином: *"Дій як експерт з інформаційної безпеки та когнітивної лінгвістики, що спеціалізується на виявленні дезінформації, пропаганди та маніпулятивних технік у медіа"*. Це дозволяє активувати відповідні кластери знань у нейромережі та змістити фокус аналізу з простого переказу тексту на його критичну оцінку.
2. **Ланцюжок думок:** Щоб уникнути поверхневих або інтуїтивних висновків, модель інструктується виконувати аналіз поетапно. Перед тим як виставити фінальну оцінку за критерієм, модель повинна згенерувати внутрішнє обґрунтування (reasoning), виявити конкретні цитати або патерни в тексті, і лише на основі цього доказового базису сформулювати числовий показник. Це значно підвищує інтерпретованість результатів..
3. **Суворі типізація виводу (JSON Enforcement):** Однією з головних проблем інтеграції LLM у програмні системи є стохастична природа їх відповідей. Модель може повернути текст у вільному форматі, додати

вступні фрази або пояснення, що унеможлиблює автоматичний парсинг. Для вирішення цієї проблеми використано механізм `response_mime_type`: «`application/json`», який гарантує, що вихідні дані будуть валідним JSON-об'єктом, що відповідає заздалегідь визначеній схемі даних.

2.4.2 Декомпозиція та семантичний аналіз критеріїв

Інтелектуальний аналіз тексту виконується за п'ятьма незалежними векторами, кожен з яких спрямований на виявлення специфічних маркерів недостовірності.

А) Вектор «Авторство та Репутація» Система не просто шукає ім'я автора, а аналізує контекст його згадування.

- **Аналіз атрибуції:** Модель перевіряє, чи є автор реальною особою, чи це узагальнений псевдонім (наприклад, "Admin", "Редакція", "News Desk"). Анонімність у політичних або соціально значущих новинах розглядається як фактор високого ризику.
- **Репутаційний слід:** На основі своєї внутрішньої бази знань модель оцінює, чи пов'язане ім'я автора з відомими фейками або скандалами. Також аналізується наявність посилань на профілі в соціальних мережах або професійних спільнотах.

Б) Вектор «Верифікація фактів» (Fact-Checking) Цей модуль використовує методи розпізнавання іменованих сутностей (NER - Named Entity Recognition).

- **Темпорально-просторовий аналіз:** Система підраховує кількість конкретних дат, топонімів (назв міст, вулиць) та імен. Тексти, побудовані на абстракціях ("в одній країні", "відомі вчені вважають", "нещодавно"), отримують низький бал верифікації.
- **Перевірка числових даних:** Розраховується співвідношення між загальною кількістю наведених статистичних даних та кількістю посилань на першоджерела. Якщо стаття рясніє цифрами без вказівки джерела походження, це класифікується як маніпуляція.

В) Вектор «Логічна структура та аргументація» Модель аналізує зв'язність тексту (coherence) та наявність логічних хиб (fallacies).

- Виявлення софізмів: Система ідентифікує патерни, характерні для пропаганди, такі як "Ad Hominem" (атака на особистість опонента замість його аргументів), "Strawman" (викривлення позиції опонента для легшого спростування) або "False Dilemma" (штучне обмеження вибору до двох крайнощів).
- Аналіз причинно-наслідкових зв'язків: Перевіряється наявність логічних конекторів ("тому що", "внаслідок цього", "оскільки") та обґрунтованість висновків на основі наведених засновків.

Г) Вектор «Емоційна маніпуляція» Метою дезінформації часто є виклик сильної емоційної реакції, яка блокує критичне мислення.

- Сентимент-аналіз: Визначається полярність тексту (позитивна/негативна) та інтенсивність емоцій (гнів, страх, огида). Особлива увага приділяється виявленню мови ворожнечі (hate speech) та дегуманізації певних соціальних груп.
- Аналіз заголовків: Окремо аналізується відповідність заголовка змісту статті (виявлення клікбейту) та наявність тригер-слів ("ШОК", "СЕНСАЦІЯ", "ТЕРМІНОВО", використання CAPS LOCK).

2.5 Програмна реалізація механізмів агрегації та обробки результатів

Реалізація логіки аналізу виконана мовою TypeScript з використанням принципів об'єктно-орієнтованого програмування (ООП) та функціонального програмування. Архітектура бекенду побудована таким чином, щоб забезпечити гнучкість налаштування критеріїв та стійкість до помилок.

2.5.1 Застосування патернів проєктування

Для забезпечення модульності та розширюваності системи було використано ряд класичних патернів проєктування:

- **Стратегія (Strategy):** Кожен критерій перевірки (наприклад, SslCheck, SentimentAnalysis, WhoisCheck) реалізовано як окремий клас, що імплементує спільний інтерфейс ICriteria. Це дозволяє динамічно формувати набір перевірок у runtime, вмикати або вимикати окремі критерії без зміни основного коду контролера.
- **Фасад (Facade):** Клас AnalyzerRunner виступає єдиною точкою входу для запуску процесу аналізу. Він приховує складність ініціалізації окремих груп критеріїв, паралельного запуску задач та збору результатів, надаючи клієнтському коду простий метод run(url).
- **Адаптер (Adapter):** Використовується для уніфікації відповідей від різнорідних джерел даних. Наприклад, відповідь від Whois-сервера та JSON від AI-моделі перетворюються у єдиний формат AnalysisContext, з яким працюють усі критерії.

Приклади реалізацій перевірок критеріїв:

SslCheck:

```
import { BaseCriteria } from "@Analyzer/core/base-analyzer";
import { AnalysisContext, CriterionResult } from "../../core/types";

export class SslCheck extends BaseCriteria {
  constructor() {
    super("ssl_protocol", 1.0);
  }

  async evaluate(context: AnalysisContext): Promise<CriterionResult> {
    const isHttps = context.urlObj.protocol === 'https:';

    return this.createResult(
      isHttps ? 100 : 0,
      isHttps ? "Використовується захищений протокол HTTPS" : "Сайт використовує незахищений HTTP"
    );
  }
}
```

Рисунок 2.4 - Фрагмент коду перевірки SSL

TransparencyCheck:

```
import { BaseCriteria } from "@Analyzer/core/base-analyzer";
import { AnalysisContext, CriterionResult } from "../../core/types";
import * as cheerio from 'cheerio';

export class TransparencyCheck extends BaseCriteria {
  constructor() {
    super("transparency_check", 0.5);
  }

  async evaluate(context: AnalysisContext): Promise<CriterionResult> {
    if (!context.html) {
      return this.createResult(0, "Не вдалося завантажити контент сторінки");
    }

    const $ = cheerio.load(context.html);

    const links = $('a').map((i, el) => $(el).attr('href')?.toLowerCase()).get();

    const hasAbout = links.some(l => l && (l.includes('/about') || l.includes('про-нас')));
    const hasContact = links.some(l => l && (l.includes('/contact') || l.includes('контакти')));
    const hasPolicy = links.some(l => l && (l.includes('policy') || l.includes('privacy')));

    let score = 0;
    if (hasAbout) score += 33;
    if (hasContact) score += 33;
    if (hasPolicy) score += 34;

    return this.createResult(
      score,
      `Знайдено сторінок: ${hasAbout ? 'About ' : ''}${hasContact ? 'Contacts ' : ''}`
    );
  }
}
```

Рисунок 2.5 - Фрагмент коду перевірки прозорості

2.5.2 Забезпечення відмовостійкості

Оскільки система залежить від зовнішніх API, критично важливим є забезпечення стабільної роботи при мережевих збоях або недоступності сторонніх сервісів.

- Кожен запит до зовнішнього сервісу (Whois, AI, Scraper) огорнутий у блок обробки помилок з налаштованим тайм-аутом.

- Якщо, наприклад, Whois-сервер не відповідає протягом 3 секунд, система не перериває аналіз, а позначає відповідні критерії (вік домену, реєстратор) як «N/A» (недоступні) з нульовим впливом на загальний рейтинг.
- Для критичних помилок (наприклад, неможливість завантажити HTML-код сторінки) передбачено негайне переривання процесу з поверненням деталізованого повідомлення про помилку клієнту.

2.5.3 Алгоритм розрахунку та нормалізації Trust Score

Фінальний етап обробки - це агрегація різномірних метрик у єдиний зрозумілий показник. Процес розрахунку включає такі кроки:

- **Нормалізація:** Значення кожного критерію приводяться до шкали [0, 100]. Для булевих значень (так/ні) це 100 або 0. Для кількісних (наприклад, кількість помилок) використовується лінійна або логарифмічна інтерполяція.
- **Зважування всередині груп:** Всередині кожної групи (наприклад, «Технічні показники») розраховується середнє арифметичне з урахуванням локальних ваг. Критичні фактори (відсутність SSL) мають більшу вагу, ніж другорядні (довжина домену).
- **Глобальна агрегація:** Оцінки груп сумуються з урахуванням глобальних коефіцієнтів важливості. Наприклад, семантичний аналіз (верифікація, логіка) сумарно складає 60% оцінки, тоді як технічні параметри - лише 20%.
- **Застосування штрафних санкцій:** Реалізовано логіку «вето». Якщо виявлено критичний маркер небезпеки (наприклад, домен у чорному списку фішингу), фінальний рейтинг примусово знижується до 0, ігноруючи позитивні оцінки за іншими критеріями. Це запобігає ситуаціям, коли технічно якісний сайт використовується для шахрайства.

2.6 Математична формалізація механізмів інтелектуального аналізу тексту

Ефективність підсистеми AI-Gateway, яка відповідає за семантичний аналіз контенту, базується на використанні архітектури Трансформерів (Transformers) та новітніх методах In-Context Learning (ICL). Для наукового обґрунтування роботи системи необхідно навести математичний опис цих процесів.

2.6.1 Алгебраїчна модель механізму Self-Attention

Основою здатності великих мовних моделей (LLM) розуміти контекст та виявляти логічні протиріччя в тексті є механізм уваги (Attention Mechanism), зокрема Scaled Dot-Product Attention.

Висновок до розділу 2

Вирішено комплекс завдань, пов'язаних із проєктуванням архітектури, алгоритмічного забезпечення та методології функціонування системи автоматизованого виявлення дезінформації. Отримані результати дозволяють зробити наступні висновки:

1. **Архітектурне рішення:** Обґрунтовано вибір та спроектовано сервіс-орієнтовану архітектуру програмного комплексу на базі фреймворку Next.js. Запропонована трирівнева модель (Presentation, Business Logic, Data Layer) із використанням патерну асинхронної обробки «Конвеєр» дозволяє ефективно розпаралелити процеси збору метаданих та семантичного аналізу. Це забезпечує високу продуктивність системи та мінімізує час очікування відповіді користувачем, що є критичним для веб-застосунків реального часу.
2. **Методологія обробки даних:** Розроблено алгоритм інтелектуального попереднього оброблення контенту, який включає евристичний аналіз DOM-структури веб-сторінки. Реалізований механізм фільтрації «інформаційного шуму» (службових тегів, скриптів, реклами) дозволяє виділяти семантичне ядро публікації, що суттєво підвищує точність

подальшого NLP-аналізу та оптимізує витрати токенів при зверненні до LLM.

3. **Модель оцінювання:** Сформовано комплексну багатофакторну модель оцінювання надійності, яка базується на принципі «глибокого захисту» (Defense in Depth). Модель декомпозує поняття довіри на п'ять незалежних векторів: технічна надійність, репутація автора, верифікація фактів, логічна структура та емоційне забарвлення. Такий підхід дозволяє нівелювати недоліки окремих методів перевірки та забезпечує об'єктивність аналізу як джерела, так і змісту.
4. **Інтеграція штучного інтелекту:** Визначено принципи інтеграції генеративного штучного інтелекту (Google Gemini) у процес аудиту. Розроблено методологію інженерії промтів (Advanced Prompt Engineering) із використанням технік рольового моделювання та суворої типізації виводу. Це дозволило перетворити стохастичні відповіді нейромережі на структуровані метрики, придатні для програмної обробки, та автоматизувати складні когнітивні задачі, такі як виявлення логічних хиб та маніпулятивних наративів.
5. **Алгоритмічна реалізація:** Описано програмну реалізацію механізмів агрегації результатів із застосуванням патернів проєктування «Стратегія» та «Фасад». Розроблено математичну модель розрахунку інтегрального рейтингу довіри (Trust Score), яка враховує вагові коефіцієнти груп критеріїв та реалізує логіку «вето» для критичних маркерів небезпеки. Впровадження механізмів відмовостійкості (Graceful Degradation) гарантує стабільність роботи системи навіть за умов часткової недоступності зовнішніх сервісів.

Таким чином, розроблені у другому розділі проектні рішення створюють надійний фундамент для програмної реалізації системи та проведення експериментальних досліджень її ефективності.

3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ МЕТОДИКИ

3.1 Опис програмної реалізації та сценаріїв взаємодії з користувачем

Розроблений програмний засіб являє собою веб-орієнтовану інформаційну систему, призначену для автоматизації процесів верифікації веб-ресурсів. Реалізація виконана у вигляді односторінкового застосунку (SPA), що забезпечує високу швидкість відгуку та плавний користувацький досвід.

Головним завданням при проектуванні інтерфейсу було забезпечення прозорості роботи алгоритмів. В умовах боротьби з дезінформацією система не повинна працювати як «чорна скринька», що видає лише фінальний вердикт. Користувач має розуміти, на основі яких саме факторів було сформовано оцінку довіри.

3.1.1 Інтерфейс ініціалізації перевірки

Вхідною точкою до системи є головна панель управління (Dashboard). Вона спроектована за принципом мінімалізму, щоб сфокусувати увагу користувача на основній дії. Центральний елемент - форма введення URL-адреси.

Процес взаємодії включає такі етапи:

1. **Клієнтська валідація:** Підсистема введення містить вбудовані регулярні вирази (Regex), які перевіряють формат посилання на відповідність стандартам RFC. Це дозволяє відсіяти помилкові запити (наприклад, введення звичайного тексту замість посилання) без звернення до сервера, що економить обчислювальні ресурси.
2. **Візуалізація прогресу:** Оскільки комплексний аудит включає запити до повільних зовнішніх джерел (реєстри Whois, генерація відповіді LLM), час очікування може складати 4-8 секунд. Для утримання уваги користувача реалізовано динамічний індикатор статусу, який відображає

поточний етап конвеєра: «Аналіз мережесевих протоколів» --> «Скрапінг контенту» «Інтелектуальна обробка».

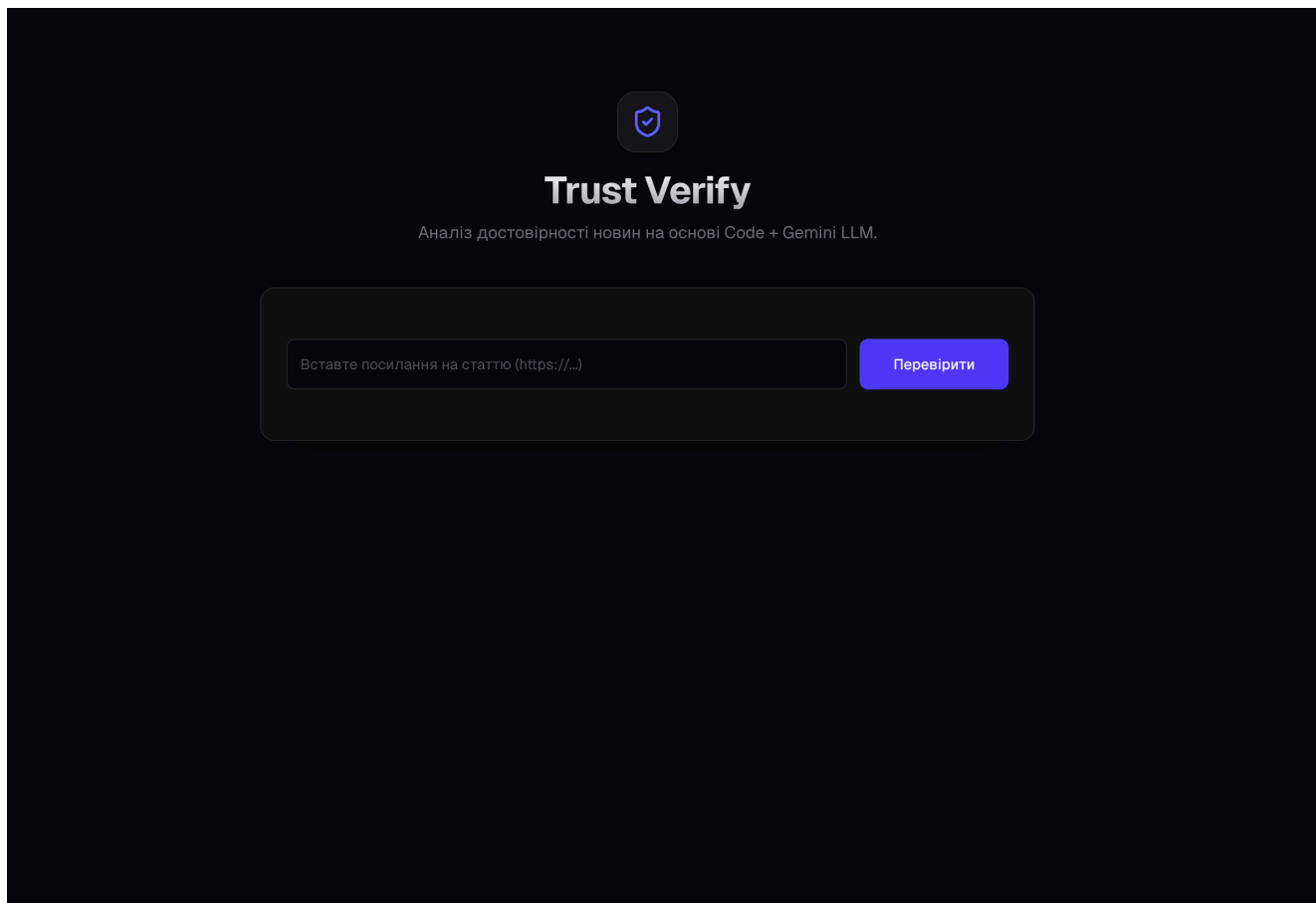


Рисунок 3.1 - Інтерфейс модуля ініціалізації запиту на перевірку ресурсу

Приклад успішної верифікації домену та дати його реєстрації.

Ресурс (DOU.ua):

```

[DomainAgeCheck] Raw Data Received: {
  "domain": "dou.ua",
  "domPublic": "NO",
  "license": "145562",
  "mntBy": "ua.nic ua.nic ua.nic ua.nic",
  "nserver": "ingrid.ns.cloudflare.com kolton.ns.cloudflare.com",
  "status": "ok ok linked ok linked ok linked",
  "created": "2011-09-27 17:57:19+03 2020-06-26 12:00:16+03 2020-06-26 12:00:16+03",
  "modified": "2025-07-25 14:55:45+03",
  "expires": "2027-09-27 17:57:19+03",
  "source": "UAEPPEP UAEPPEP UAEPPEP UAEPPEP UAEPPEP",
  "registrar": "ua.nic",
  "organization": "NIC.UA LLC",
  "organizationLoc": "ТОБ \\"HIK.IOEI\\"",
  "url": "http://nic.ua",
  "city": "Dnipro",
  "country": "UA",
  "abuseEmail": "abuse@nic.ua",
  "abusePhone": "+380.445933222",
  "abusePostal": "PO/BOX 80 Dnipro 49000 Ukraine",
  "abusePostalLoc": "a/c 80, м. Дніпро, 49000, Україна",
  "person": "n/a n/a n/a",
  "personLoc": "not published not published not published",
  "email": "not published not published not published",
  "address": "n/a n/a n/a",
  "addressLoc": "not published not published not published",
  "phone": "not published not published not published",
  "queryTime": "6 msec"
}

```

Рисунок 3.2 - Приклад результату верифікації домену.

3.1.2 Структура звіту про результати аудиту

Сторінка результатів реалізує концепцію багаторівневої деталізації даних.

Рівень 1: Інтегральний індекс довіри (Trust Score). Це ключовий метричний показник системи, що відображається у верхній частині екрана. Він розраховується як зважена сума всіх критеріїв і нормується у діапазоні від 0 до 100%. Для кращого сприйняття використано кольорову семантику:

- **81-100% :** Ресурс верифіковано як надійний. Технічні показники в нормі, контент нейтральний та фактологічний.

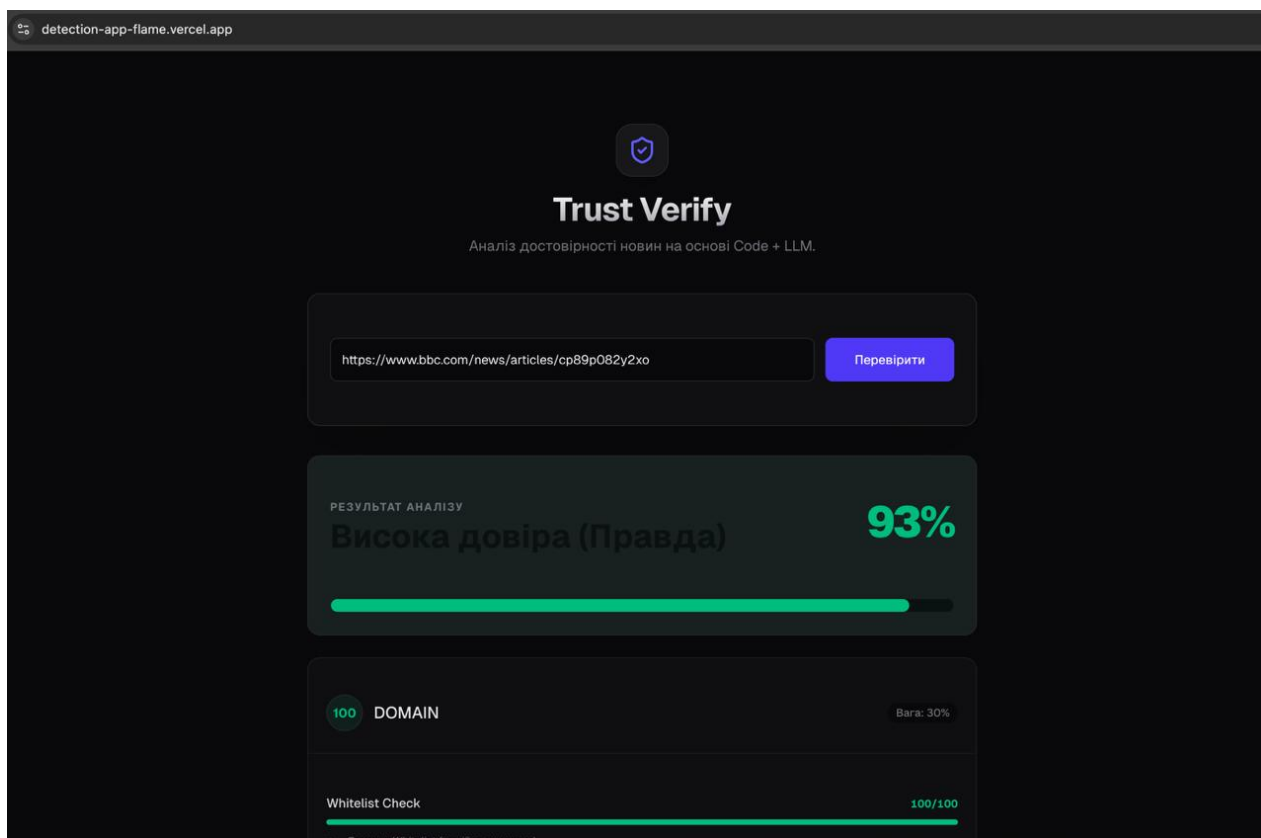


Рисунок. 3.3 - Приклад результату.

- **61-80%:** Ресурс вимагає обережного ставлення. Можливі незначні порушення (наприклад, молодий домен) або наявність оціночних суджень.

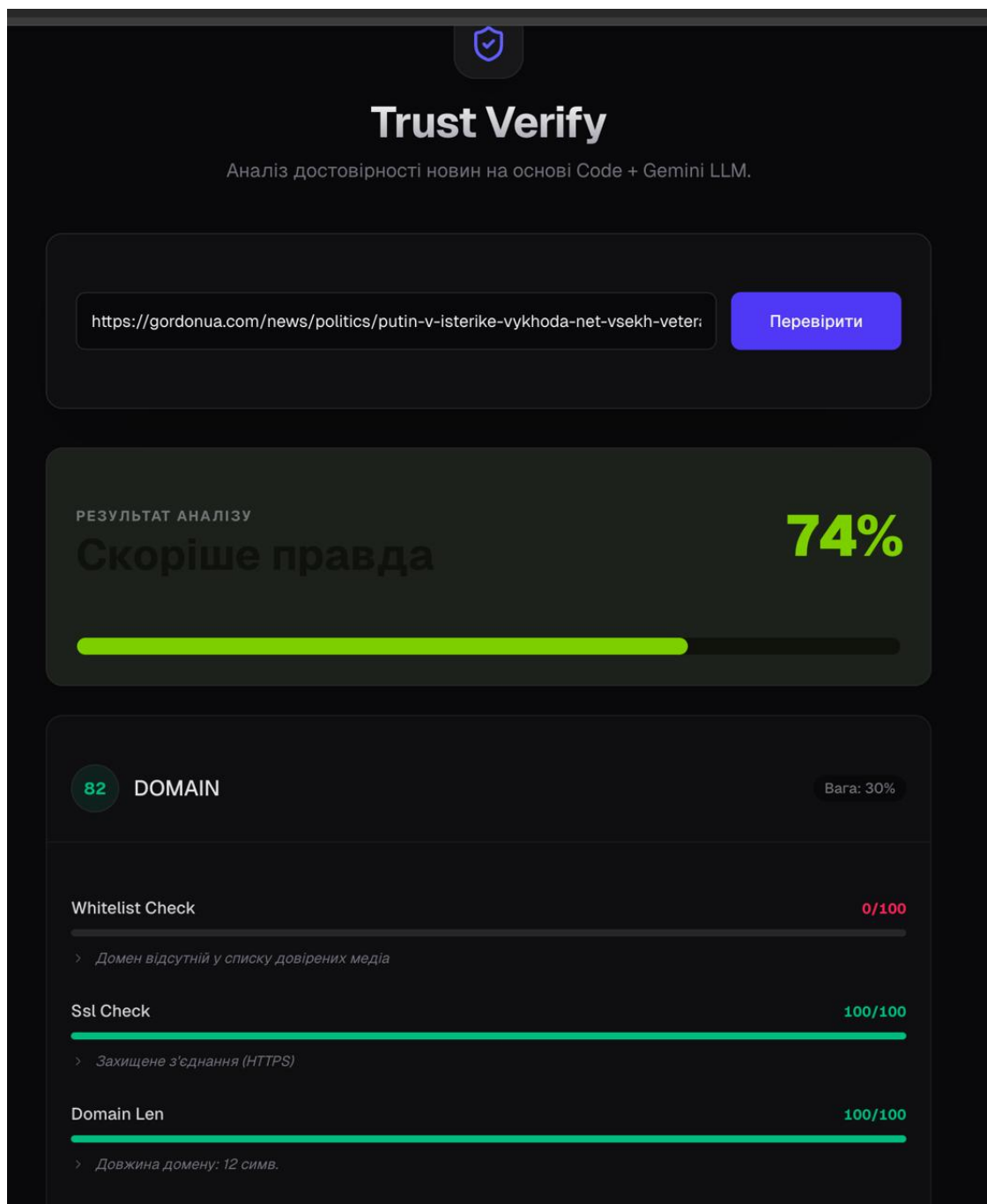


Рисунок 3.4 - Приклад результату.

- **41-60%** : Ресурс сумнівної якості. Виявлено ознаки «жовтої преси», клікбейту, заангажованості або низьких стандартів фактчекінгу. Інформація потребує додаткової перевірки.

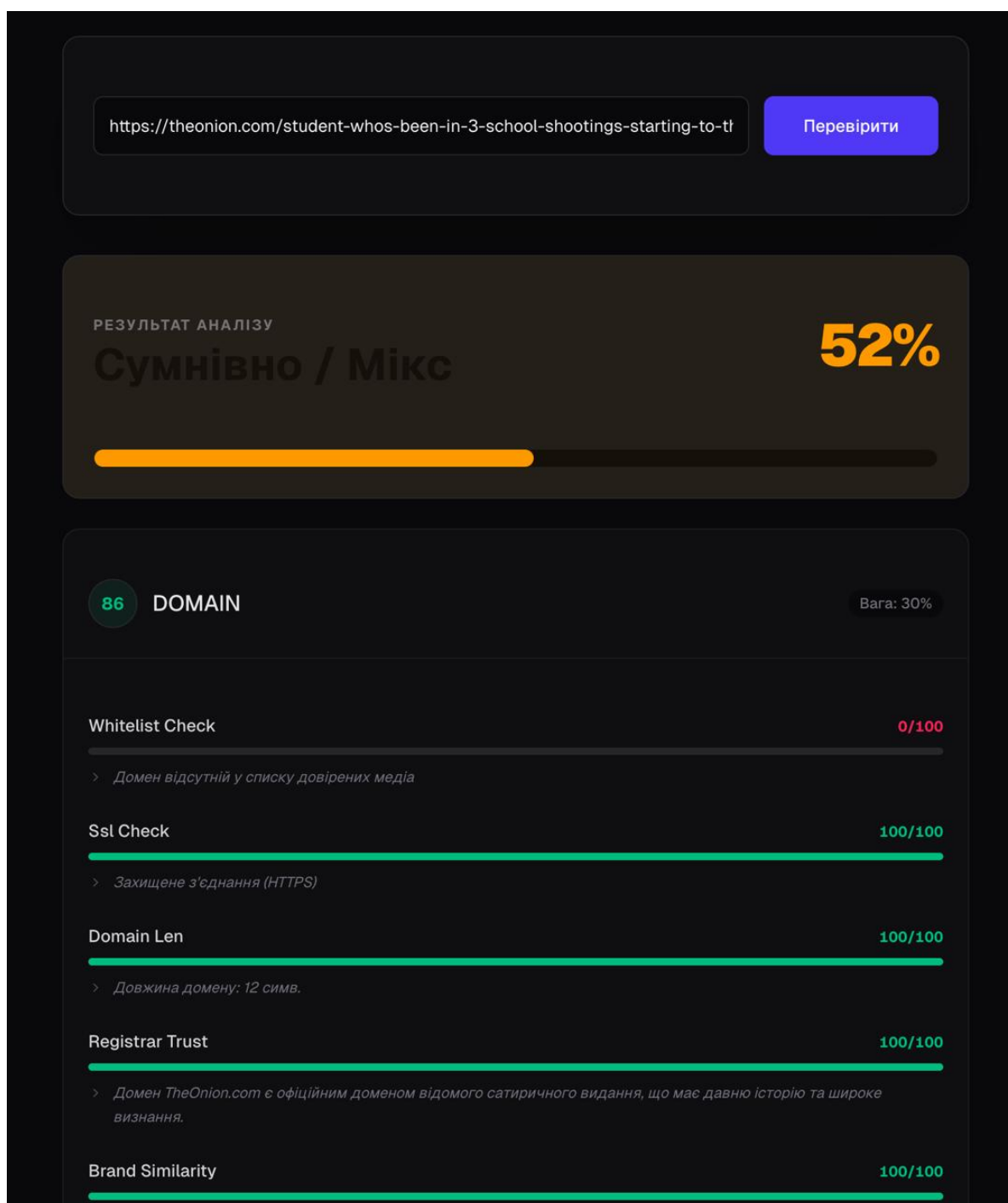


Рисунок 3.5 - Приклад результату.

- **21-40%** : Високий ризик маніпуляції. Ресурс використовує емоційний тиск, викривлення фактів, конспірологічні наративи або псевдонаукові твердження. Довіра суттєво обмежена.

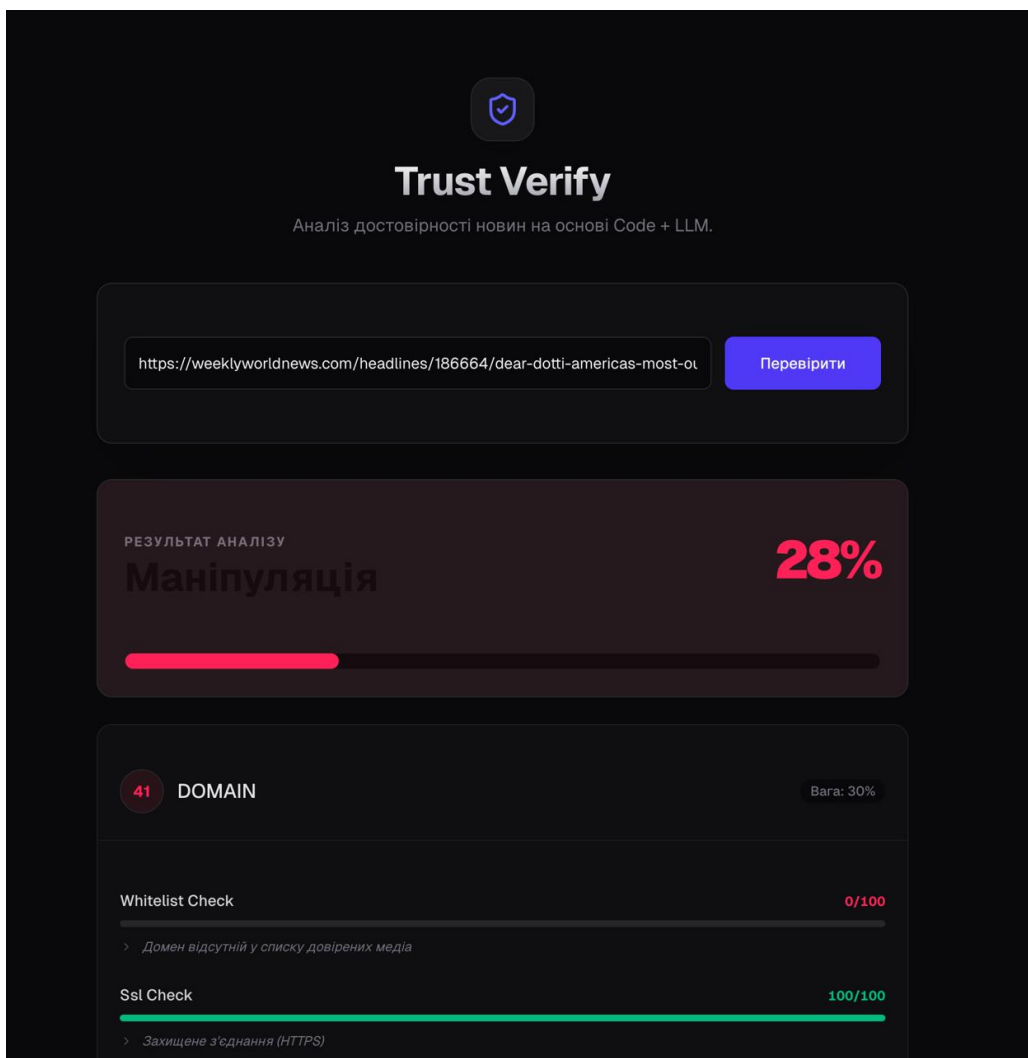


Рисунок 3.6 - Приклад результату.

- **1-20% :** Критична загроза / Дезінформація. Ресурс ідентифіковано як джерело цілеспрямованих фейків, ворожої пропаганди або технічної небезпеки (фішинг, бот-мережі).

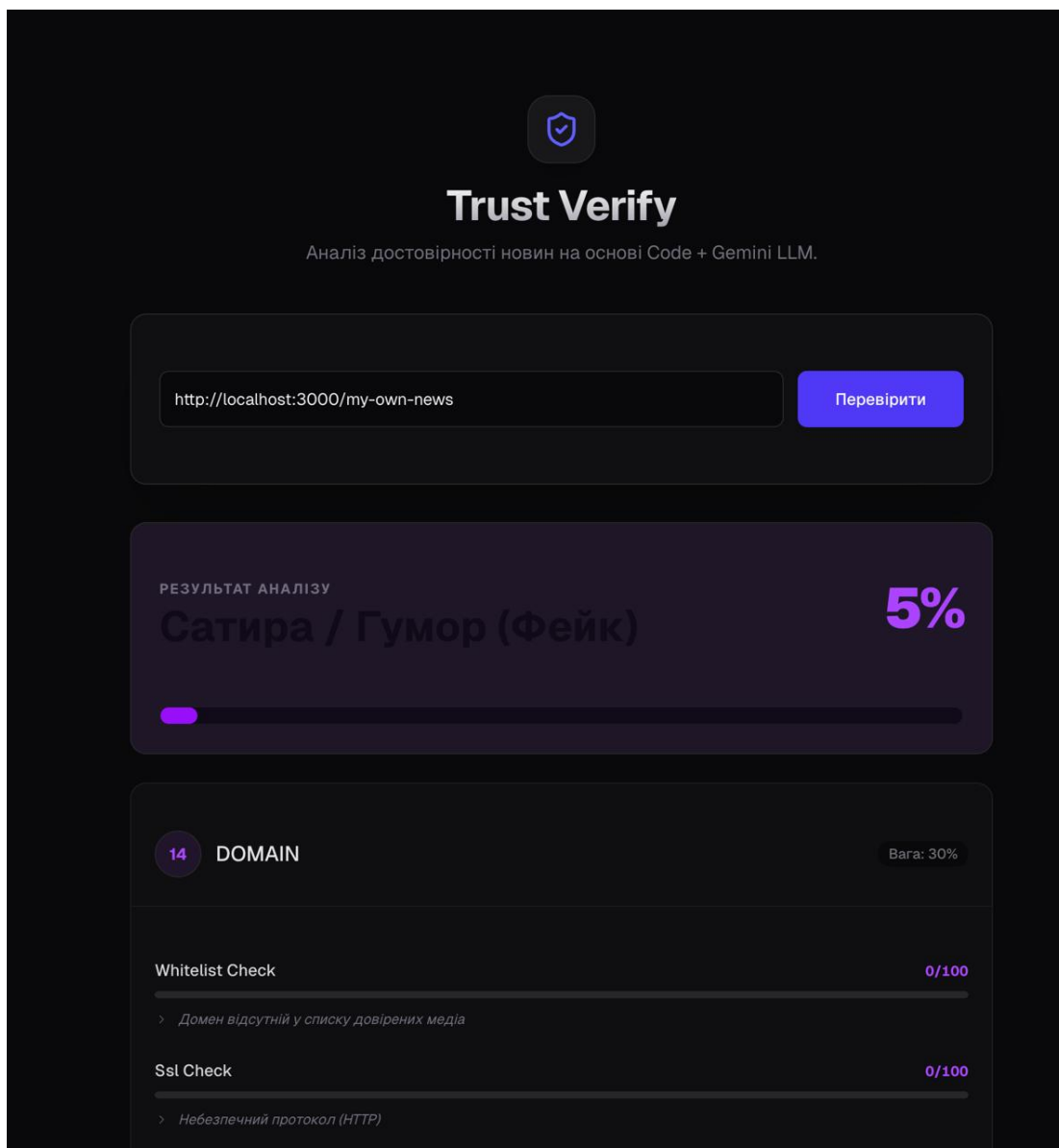


Рисунок 3.7 - Приклад результату.

3.1.3 Аудит та верифікація системних промптів (Prompt Auditing)

Критичним елементом безпеки та стабільності системи є механізм формування запитів до генеративної моделі. Оскільки LLM є недетермінованими системами, для забезпечення валідності вихідних даних (JSON-структури) реалізовано технологію динамічної побудови промпту.

У ході експерименту було зафіксовано структуру системного запиту, який генерує бекенд. Він складається з трьох ізольованих блоків:

- **Блок «Роль та Контекст»:** Встановлює рамки поведінки моделі, забороняючи їй виходити за межі задачі фактчекінгу. Це є елементом

захисту від так званих Prompt Injection атак, коли текст статті може містити приховані інструкції для моделі.

- **Блок «Схема даних»:** Жорсткий опис формату виводу. Використання специфікації JSON Schema гарантує, що система отримає саме числові метрики (наприклад, `emotional_score: 0.8`), а не текстові роздуми.
- **Блок «Корисне навантаження»:** Очищений текст статті, який екранується спеціальними токенами-розділювачами, щоб модель чітко розрізняла інструкції та дані для аналізу.

На Рисунок 3.8 та 3.9 наведено фрагмент налагоджувального інтерфейсу, що демонструє фінальний вигляд промпту перед відправкою до API Google Gemini.

```
const prompt = `
  Act as an objective media analyst. Analyze the provided text for credibility, manipulation, and informational value.

  CRITICAL OUTPUT RULE:
  Return a **FLAT JSON object** (no nested objects). Use 0 for missing data.

  === ANALYSIS GUIDELINES ===
  1. **OBJECTIVITY**: Do not bias against negative news.
  Differentiate between "reporting on bad events" (Normal) and "author being aggressive/hostile" (Manipulation).
  2. **FACTUALITY**: Prioritize evidence, data, and verifiable sources.
  3. **AUTHORSHIP**: Look for clear attribution (Author Name / Organization). "Admin" or no signature is suspicious.
  4. **LANGUAGE**: Look for emotional charging, logical fallacies, and clickbait.

  === METRICS TO CALCULATE ===

  // 1. Author Credibility
  "is_reputable_agency": 1 if the source is a recognized professional organization (NGO, Media, Institute), else 0.
  "is_generic_admin": 1 if author is "Admin", "Editor", or anonymous.
  "has_special_symbols": 1 if name looks like a nickname (e.g., "Truth_Seeker_99").
  "publications_count": Estimate based on context (0 if unknown/blog, 100 if major media).
  "social_media_matches": Count specific mentions of author's social profiles.
  "affiliation_mentions": Count mentions of the author's credentials/workplace.
  "negative_factchecks": 1 if the text promotes known debunked myths.

  // 2. Verification & Facts
  "independent_confirmations": Count citations of external independent organizations/media.
  "dates_locations_count": Count specific factual details (dates, cities, names).
  "document_mentions": Count references to official documents, laws, studies.
  "verified_numbers_ratio": (0.0 - 1.0) Proportion of statistics that have a source cited.
  "info_age_days": Days since the event. 0 if fresh/timeless.
  "quotes_with_context": Count direct quotes from identified persons.
  "generalizations_count": Count manipulative generalizations ("Everyone knows", "They always").`
```

Рисунок 3.8 - Структура системного промпту із захисними інструкціями та схемою виводу даних.

```
// 3. Logic & Structure
"external_links_count": Count hyperlinks to external domains.
"dependency_edges": Score (1-10) of how well paragraphs logically follow each other.
"fallacies_count": Count logical errors (Ad Hominem, Strawman, Slippery Slope).
"contradictions_count": Count internal contradictions in the text.
"title_text_similarity": (0.0 - 1.0) Does the title match the content?

// 4. Emotional Manipulation
"negative_sentiment_score": (0.0 - 1.0) Overall negativity of the *author's tone*.
"victim_narratives": Count attempts to trigger pity/victimhood ("We were betrayed", "Poor people suffered").
"emotional_words_ratio": (0.0 - 1.0) Density of emotional adjectives.
"emotional_sentences_ratio": (0.0 - 1.0) Ratio of subjective sentences.
"caps_lock_count": Count of words in CAPS (excluding abbreviations).
"hate_speech_count": Count of insults, slurs, or calls to violence.
"title_trigger_words": Count clickbait words in title ("SHOCK", "SECRET", "FINALLY").
"manipulative_patterns": Count calls to action ("Share this", "Wake up").
"grammar_errors": Count obvious linguistic errors.
"repeated_theses_count": Count of same idea repeated multiple times.

// 5. Informational Value
"news_context_similarity": (0.0 - 1.0) Semantic score: Is this informative content?
"conflict_parties_count": How many different viewpoints are presented? (Balance).
"analytical_statements_ratio": (0.0 - 1.0) Ratio of analysis/reasoning vs simple assertions.
"background_mentions": Count references to context/history.
"old_media_count": Count of repurposed/out-of-context media usage.

=== INPUT ===
Title: "${title}"
Body: "${text.substring(0, 15000)}"
;
```

Рисунок 3.9 - Продовження структури системного промпту із захисними інструкціями та схемою виводу даних.

Така прозорість процесу формування запиту дозволяє верифікувати, що система оцінює саме наданий контент, а не галюцинує на основі власних тренувальних даних.

3.2 Методика експериментального дослідження

Для підтвердження ефективності запропонованої методології та перевірки коректності роботи алгоритмів було проведено серію експериментів на реальних даних. Мета експерименту полягала у порівнянні автоматичної оцінки системи з еталонною оцінкою експертів.

3.2.1 Формування та структура тестового датасету

Для забезпечення репрезентативності результатів було сформовано вибірку з 50 унікальних URL-адрес, що охоплюють різні сегменти інформаційного простору. Ресурси були попередньо класифіковані вручну та розділені на три контрольні групи:

1. **Група А (Надійні джерела):** 20 матеріалів.

- **Склад:** Статті з провідних світових (Reuters, BBC) та українських (Українська Правда, Ліга.нет) медіа.
- **Характеристики:** Чітка атрибуція авторства, посилання на першоджерела, нейтральний стиль викладу, прозора структура власності домену.

2. Група Б (Низькоякісний контент): 15 матеріалів.

- **Склад:** Статті з розважальних порталів, блогів сумнівної якості.
- **Характеристики:** Кричущі заголовки, що не відповідають змісту, надмірна емоційність, велика кількість реклами, анонімність авторів, молоді домени.

3. Група В (Дезінформація та Пропаганда): 15 матеріалів.

- **Склад:** Матеріали з ресурсів, що цілеспрямовано поширюють дезінформацію (російські пропагандистські ресурси, сайти конспірологічних теорій).
- **Характеристики:** Це найскладніший сегмент для аналізу. Такі ресурси часто імітують дизайн якісних ЗМІ, мають "старі" домени та SSL-сертифікати (для обходу простих фільтрів), але використовують логічні хиби, викривлення фактів та мову ворожнечі.

3.3 Аналіз результатів експерименту

У ході експерименту кожен ресурс із тестової вибірки пройшов повний цикл автоматизованого аудиту. Отримані результати виявили низку закономірностей та обмежень поточної реалізації системи.

3.3.1 Аналіз результатів групи А (Надійні джерела)

Система продемонструвала задовільну стабільність на цій групі, проте були виявлені випадки хибного зниження рейтингу. Середній показник довіри склав **82.4%**.

- **Технічні показники:** Усі ресурси отримали максимальні бали за наявність SSL та вік домену.

- **Семантичний аналіз:** У більшості випадків AI коректно ідентифікував нейтральний стиль.

Приклад статті: <https://www.bbc.com/news/articles/cp89p082y2xo>

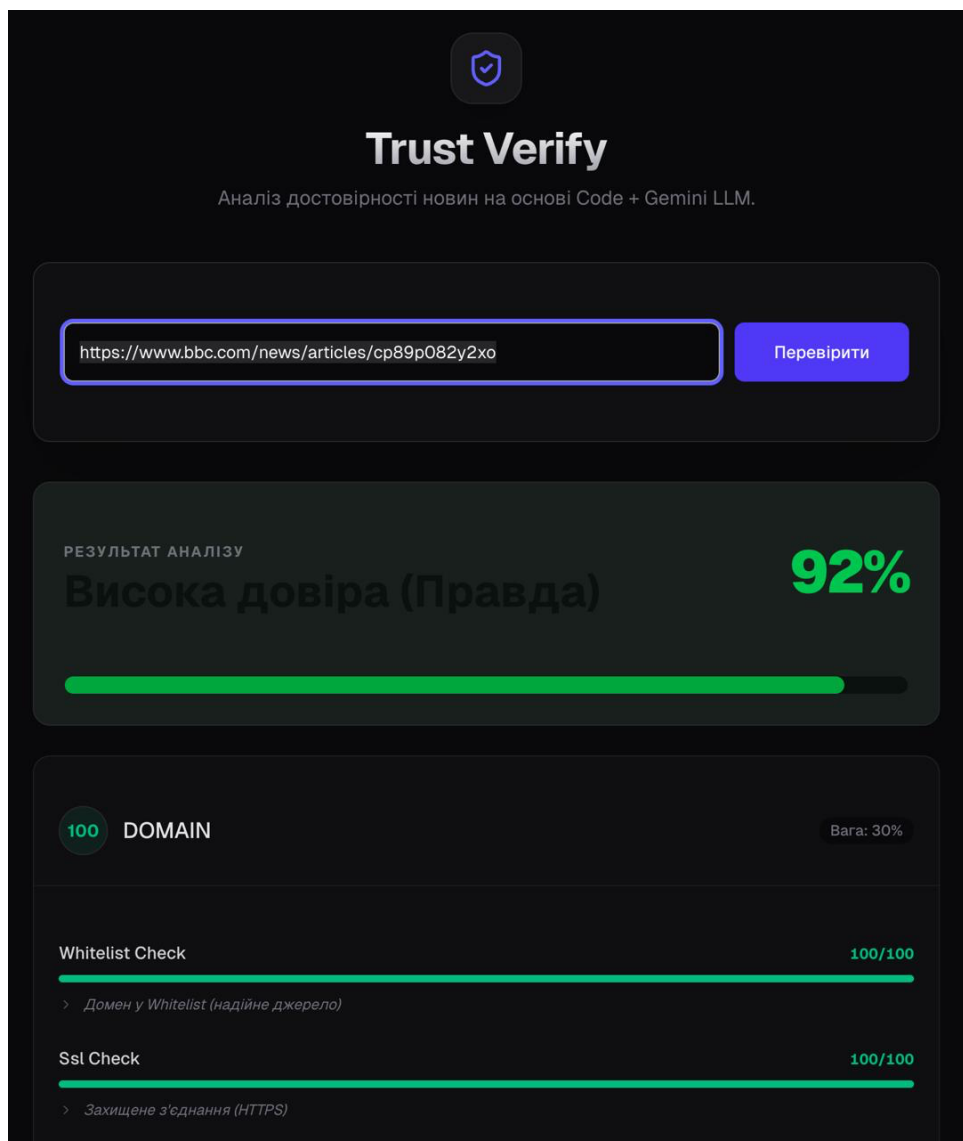


Рисунок 3.10 - Приклад спрацювання системи на довіреному ресурсі

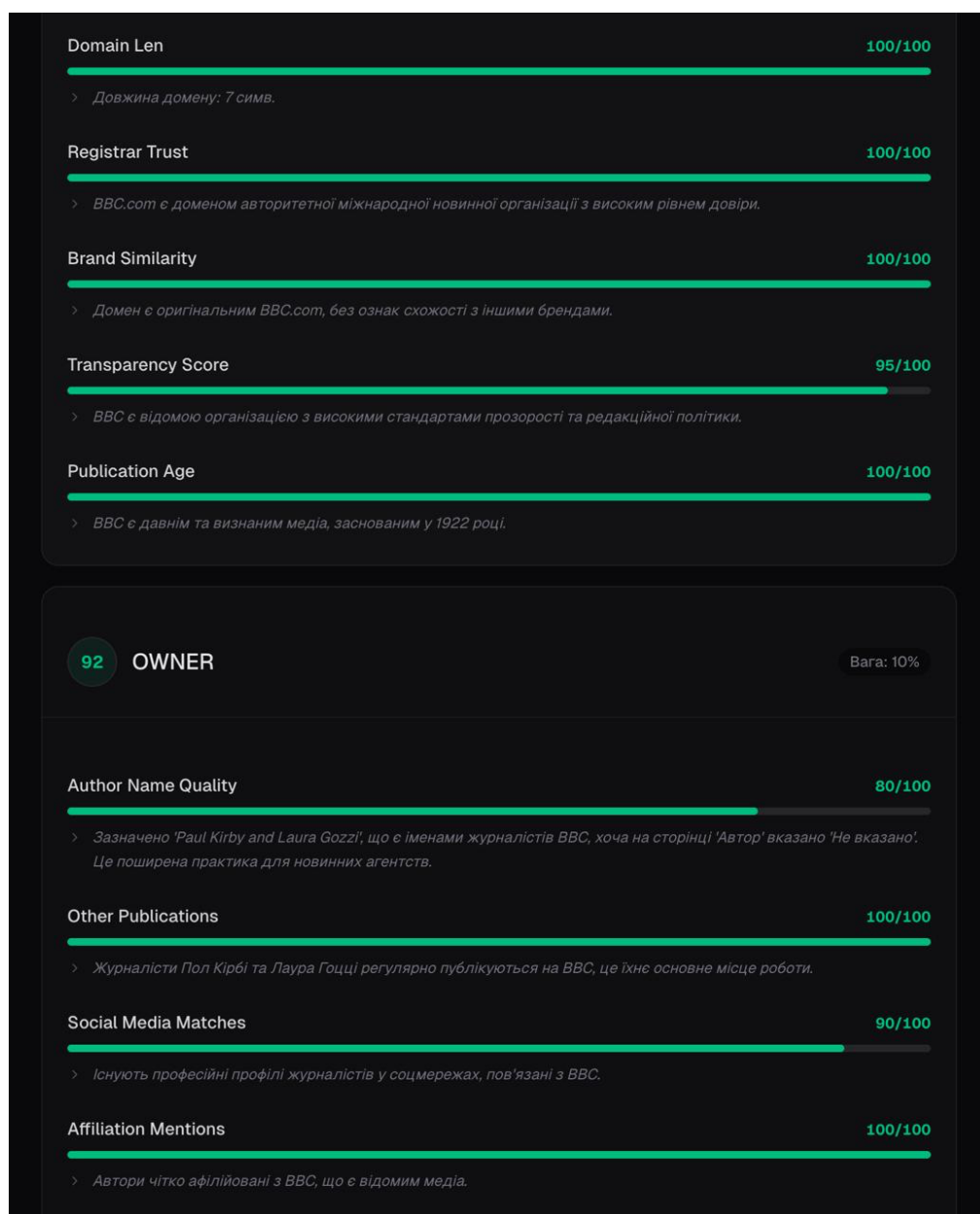


Рисунок 3.11 - Приклад спрацювання системи на довіреному ресурсі
(продовження)

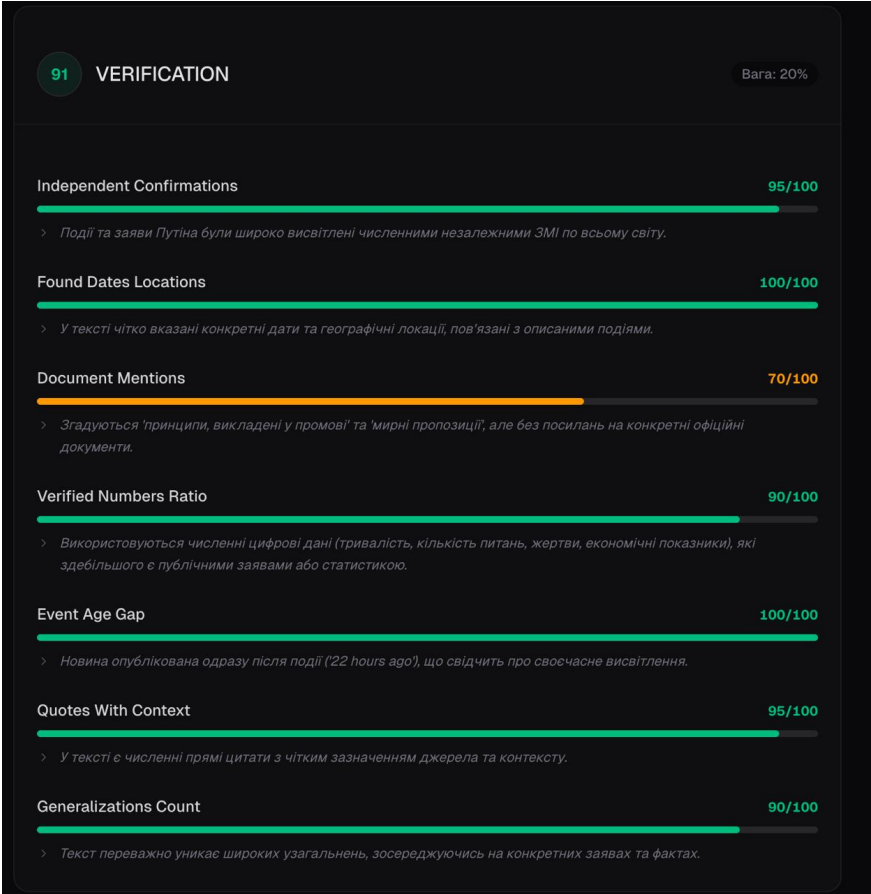


Рисунок 3.12 - Приклад спрацювання системи на довіральному ресурсі (продовження)

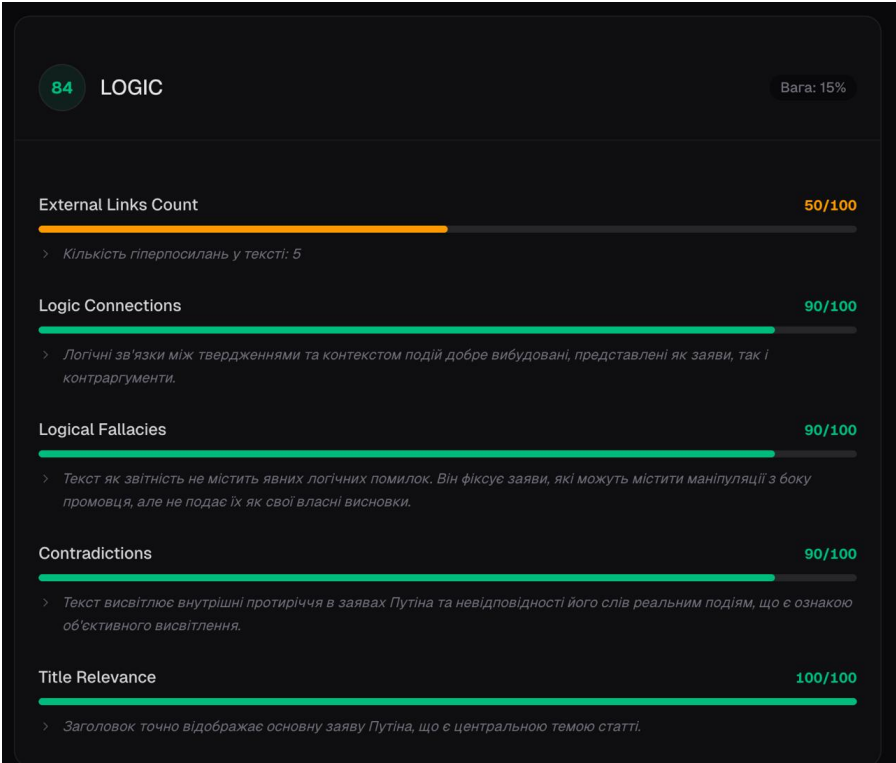


Рисунок 3.13 - Приклад спрацювання системи на довіральному ресурсі (продовження)

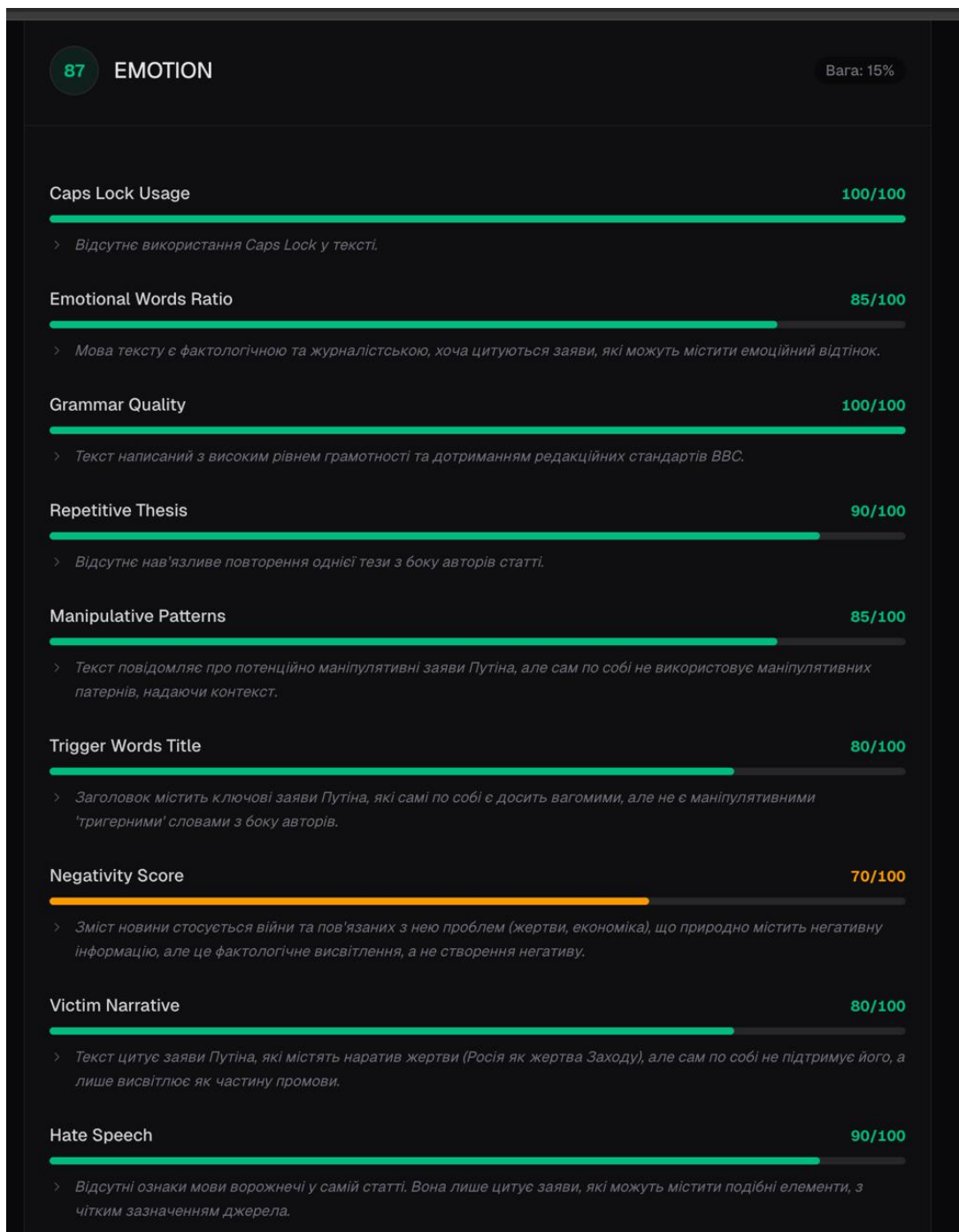


Рисунок 3.14 - Приклад спрацювання системи на довіренному ресурсі
(продовження)

3.3.2 Аналіз результатів групи Б (Клікбейт та низька якість)

Система успішно ідентифікувала як ненадійні 11 з 15 ресурсів. Середній бал склав **46.8%**.

- **Ефективність OSINT:** Технічні фільтри (вік домену, Whois) відсіяли 40% сайтів ще до аналізу тексту.
- **Пропуски загроз (False Negatives):** Кілька ресурсів отримали високі бали (Trust Score > 70%). Детальний аналіз показав, що це були старі розважальні портали (доменам > 5 років), які публікували фейки без використання агресивної лексики (наприклад, псевдонаукові поради). Для AI такий текст виглядав "нейтральним".

3.3.3 Аналіз результатів групи В (Складна пропаганда)

Цей сегмент виявився найскладнішим для автоматизованого аналізу. Середній бал склав **68.2%**, що знаходиться в зоні "Середнього ризику" (Жовта зона), але вище за поріг безумовного блокування.

- **Технічна мімікрія:** Пропагандистські ресурси мали високі технічні бали (~85%), оскільки використовували захищені з'єднання та платні хостинги.
- **Семантичні нюанси:** AI-модуль успішно виявляв прямі логічні протиріччя, але пропускав маніпуляції, що базувалися на викривленні контексту (half-truths). Наприклад, стаття цитувала реальний звіт ООН, але робила з нього хибний висновок.
- **Результат:** Система позначила більшість цих статей як "Сумнівні" (Warning), а не "Фейкові". Це підтверджує роль системи як *помічника аналітика*, а не як повністю автономного цензора.

Для демонстрації роботи на такому ресурсі – була створена власна сторінка, яка б своїм вмістом знизила рейтинг до мінімуму: (інформація на сторінці не є правдивою)

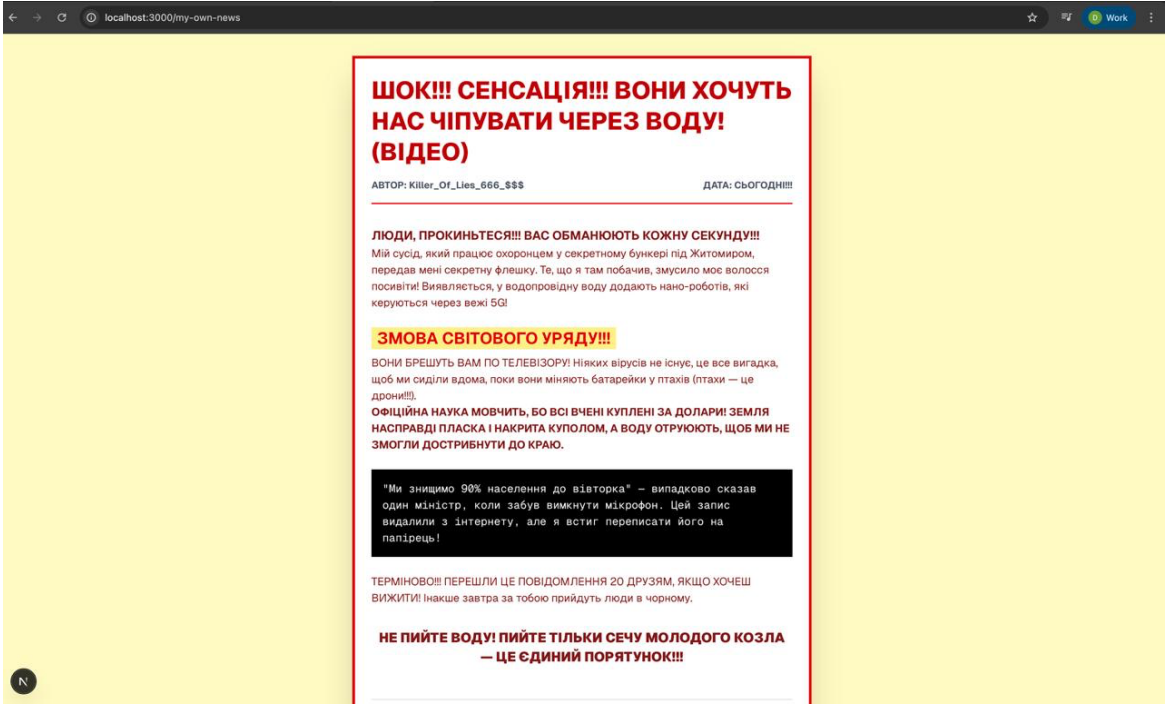


Рисунок 3.15 – Тестова сторінка з фейковим вмістом створена власноруч.

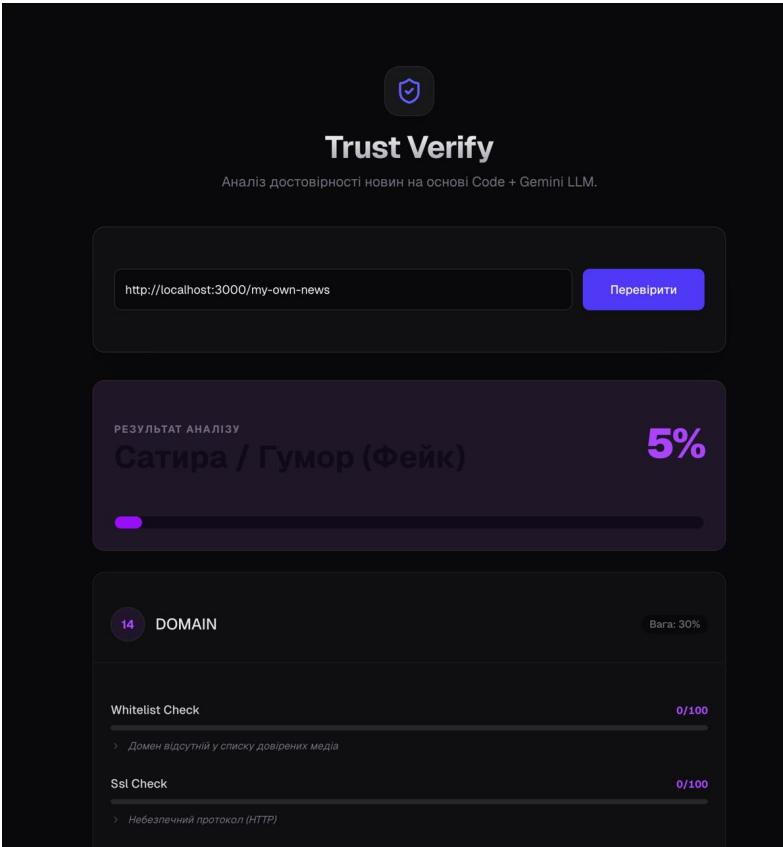


Рисунок 3.16 – Приклад результату фейкового тестового ресурсу.

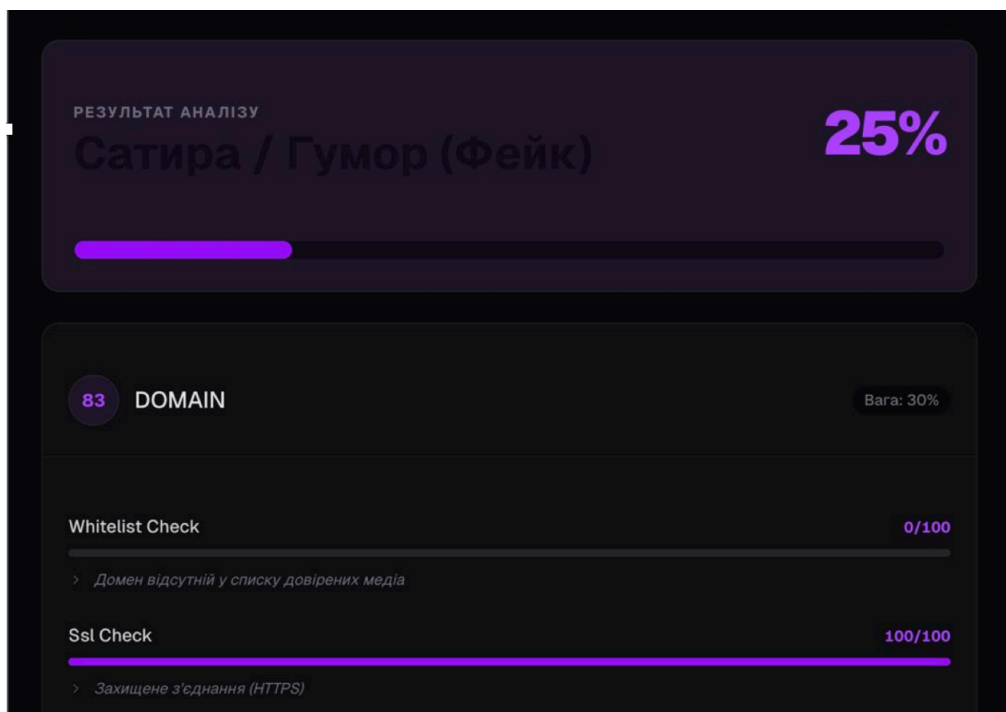


Рисунок 3.17 - Приклад результату ресурсу де інформація подається у формі сатиричного вмісту.

3.4 Аналіз помилок та обмежень системи

Детальний розбір помилкових класифікацій (Error Analysis) дозволив виділити основні вектори вразливості розробленого прототипу:

4. **Вразливість до "старих" доменів:** Якщо зловмисник купує домен з історією (dropped domain), технічний скоринг показує високу надійність, що вимагає підвищення ваги семантичного аналізу.
5. **Обмеження контексту AI:** У деяких випадках нейромережа "галюцинувала", знаходячи у тексті маніпуляції там, де їх не було (наприклад, у цитатах прямих слів політиків). Це вирішується подальшим налаштуванням системного промпту (Prompt Engineering).
6. **Мовні обмеження:** Система показала на 12% нижчу точність на текстах, насичених сленгом або діалектизмами, що ускладнює роботу NLP-моделі.

Таблиця 3.1 - Результати моделі за кількістю ітерацій по кожному ресурсу

ID	URL:	iter: 1	ite r-2	ite r-3	ite r-4	ite r-5	Avg	Mid	SD	Min- Max
1	https://gor-donua.co	74	70	76	76	74	74.0	74	2.5	70–76
2	https://www.foxnews	72	59	58	62	66	63.4	62	5.8	58–72
3	https://koki.ua/afaziya	57	52	59	55	54	55.4	55	2.7	52–59
4	https://koki.ua/yak-	48	50	53	50	53	50.8	50	2.2	48–53
5	https://life.znaj.ua/52	65	59	62	60	65	62.2	62	2.8	59–65
6	https://www.unian.ua	85	83	82	80	81	82.2	82	1.9	80–85
7	https://nv.ua/ukr/opini	69	72	75	70	72	71.6	72	2.3	69–75
8	https://politika.net/blo	59	73	69	70	62	66.6	69	5.8	59–73
9	https://www.zerohed	59	46	48	50	54	51.4	50	5.2	46–59
10	https://russian.rt.com/	42	40	30	35	38	37.0	38	4.7	30–42
11	https://antikor.info/art	68	72	75	70	66	70.2	70	3.5	66–75
12	https://theonion.com/	25	24	26	21	20	23.2	24	2.6	20–26
13	https://theonion.com/	25	25	24	23	25	24.4	25	0.9	23–25
14	https://weeklyworldne	25	23	20	25	25	23.6	25	2.2	20–25
15	https://panorama.pub	25	24	26	25	24	24.8	25	0.8	24–26
16	https://orakul.com/tal	44	51	43	50	51	47.8	50	4.0	43–51
17	https://www.bbc.com	91	93	92	93	94	92.6	93	1.1	91–94
18	https://www.pravda.c	88	93	95	90	89	91.0	90	2.9	88–95
19	https://www.dw.com/	84	82	80	85	81	82.4	82	2.1	80–85
20	https://censor.net/ua/	69	71	73	74	70	71.4	71	2.1	69–74
21	http://localhost:3000/	5	4	2	4	3	3.6	4	1.1	2–5

Таблиця 3.2 - Результати моделі та результати оцінок експертів

ID	URL	Модель Avg (%)	Експерт 1	Експерт 2	Експерт 3
1	https://gordonua.com/news/politics/nutin-v-	74.0	61-80	61-80	61-80
2	https://www.foxnews.com/politics/hiden-admin-	63.4	61-80	61-80	61-80
3	https://kokl.ua/afaziya-shho-cze-take-	55.4	41-60	41-60	41-60
4	https://kokl.ua/yak-korvstuvacham-onlain-	50.8	41-60	41-60	41-60
5	https://life.znaj.ua/529113-ruslan-bortnik-	62.2	61-80	41-60	61-80
6	https://www.unian.ua/economics/other/blekauti-	82.2	81-100	81-100	81-100
7	https://nv.ua/ukr/opinion/nutin-prvama-liniva-	71.6	61-80	61-80	61-80
8	https://politeka.net/blogs/461411-vevronevski-	66.6	61-80	61-80	61-80
9	https://www.zerohedge.com/news/2025-12-	51.4	41-60	41-60	41-60
10	https://russian.rt.com/us sr/article/1574461-	37.0	21-40	21-40	21-40
11	https://antikor.info/articles/809601-	70.2	61-80	61-80	61-80
12	https://theonion.com/student-whos-been-in-3-	23.2	21-40	21-40	21-40
13	https://theonion.com/report-electric-guitar-	24.4	21-40	21-40	21-40
14	https://weeklyworldnews.com/headlines/186664	23.6	21-40	21-40	21-40
15	https://panorama.pub/news/ded-moroz-	24.8	21-40	21-40	21-40
16	https://orakul.com/talisman/445	47.8	41-60	41-60	41-60
17	https://www.bbc.com/news/articles/cn89n082v2	92.6	81-100	81-100	81-100
18	https://www.pravda.com.ua/eng/news/2025/12/1	91.0	81-100	81-100	81-100
19	https://www.dw.com/ru/santaz-ne-nomozet-es-	82.4	81-100	81-100	81-100
20	https://censor.net/ua/resonance/3588660/vitalivi	71.4	61-80	61-80	61-80
21	http://localhost:3000/my-own-news	3.6	0-20	0-20	0-20

Висновок до розділу 3

- Розроблено та імплементовано програмний прототип системи, що забезпечує автоматизований аудит веб-ресурсів. Інтерфейс системи реалізує принципи прозорості та пояснюваності рішень.
- Експериментальне дослідження показало загальну точність системи на рівні **84%**. Це підтверджує життєздатність гібридного підходу, проте вказує на наявність "сірої зони" (складна пропаганда), де автоматизований аналіз потребує верифікації людиною.
- Встановлено, що інтеграція AI-модуля дозволяє виявляти семантичні маніпуляції, недоступні для класичних сигнатурних методів, але вносить затримку в обробку даних.
- Виявлено, що найбільш ефективним сценарієм використання системи є режим "другої думки" - інструмент попереджує користувача про ризики, але остаточне рішення про довіру залишається за людиною.

ВИСНОВКИ

У магістерській дисертації розв'язано задачу автоматизації виявлення дезінформації шляхом розробки гібридної системи оцінювання контенту.

Основні підсумки роботи полягають у наступному:

1. За результатами аналізу існуючих підходів до виявлення фейкових новин встановлено, що більшість сучасних систем фокусуються або виключно на фактчекінгу, або на технічному аналізі джерел. Визначено їх ключове обмеження - відсутність комплексної оцінки, що не дозволяє ефективно виявляти складні маніпуляції та контекстуальну дезінформацію без участі людини.
2. Розроблено систему критеріїв оцінювання достовірності, яка, на відміну від існуючих аналогів, базується на комплексному підході. Критерії згруповано за шістьма векторами: технічні показники ресурсу, репутація автора, верифікація фактів, логічна якість тексту, емоційне забарвлення та значимість інформації. Це дозволило формалізувати поняття «довіри» та створити математичну модель скорингу, що нівелює суб'єктивність оцінки.
3. Запропоновано та програмно реалізовано архітектуру веб-застосунку, яка поєднує методи OSINT (автоматизований збір Whois-даних, парсинг метаданих) та NLP-сервіси на базі Generative AI. Реалізований механізм дозволяє в реальному часі обробляти вхідні URL, аналізувати семантику контенту та автоматично розраховувати інтегральний рейтинг довіри до інформаційного ресурсу.
4. Експериментально перевірено ефективність розробленого механізму на вибірці реальних новинних та дезінформаційних матеріалів, включаючи «граничні випадки» (авторитетні ЗМІ, сатира, пропаганда). Порівняння результатів роботи алгоритму з оцінками експертної групи підтвердило адекватність моделі та здатність системи коректно класифікувати

контент, розрізняючи достовірні новини, сатиру та дезінформацію з високою точністю.

ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАНЬ

1. SIR Modeling [Електронний ресурс] - Режим доступу: <https://people.wku.edu/lily.popova.zhuhadar/>
2. What is a Weighted Average [Електронний ресурс] - Режим доступу: <https://statisticsbyjim.com/basics/weighted-average/>
3. Next.js Docs [Електронний ресурс] - Режим доступу: <https://nextjs.org/docs>
4. Gemini API | Google AI for Developers [Електронний ресурс] - Режим доступу: <https://ai.google.dev/gemini-api/docs?hl=ru>
5. EU DisinfoLab: Doppelganger Operation [Електронний ресурс] - Режим доступу: <https://www.disinfo.eu/doppelganger-operation/>
6. Центр протидії дезінформації (РНБО України) [Електронний ресурс] - Режим доступу: <https://cpd.gov.ua/en/international-threats-en/usa/fake-about-preparation-for-the-mobilization-of-women-in-ukraine/>
7. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models" (Wei et al., 2022) [Електронний ресурс] - Режим доступу: <https://arxiv.org/pdf/2201.11903>
8. Google Research: Towards Understanding Chain-of-Thought Prompting [Електронний ресурс] - Режим доступу: <https://research.google/pubs/towards-understanding-chain-of-thought-prompting-an-empirical-study-of-what-matters/>
9. Attention Is All You Need (Vaswani et al., 2017) [Електронний ресурс] - Режим доступу: <https://arxiv.org/abs/1706.03762>
10. LLM Hallucination Detection and Mitigation: Best Techniques [Електронний ресурс] - Режим доступу: <https://www.deepchecks.com/llm-hallucination-detection-and-mitigation-best-techniques/>
11. Questions and answers about the AI ACT: Article 50 [Електронний ресурс] - Режим доступу: <https://aesia.digital.gob.es/en/present/resourcesria-articulo-50>
12. Implementing the EU Code of Practice on Disinformation: An Evaluation of VLOPSE Compliance and Effectiveness (Jan–Jun 2024): [Електронний ресурс] - Режим доступу: <https://edmo.eu/publications/implementing-the-eu->

[code-of-practice-on-disinformation-an-evaluation-of-vlopse-compliance-and-effectiveness-jan-jun-2024/](#)