МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ І ПРИКЛАДНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЛАПТЄВА МАРІЯ АРТЕМІВНА

Допускається до захисту:
в.о. завідувача кафедри
інформаційних технологій,
д-р. техн. наук, професор
_____Наталія ВЕСЕЛОВСЬКА
«____»_____ 2026 р.

РОЗРОБКА ЗАСТОСУНКУ ВИЯВЛЕННЯ МАНІПУЛЯТИВНОГО КОНТЕНТУ В СОЦІАЛЬНИХ МЕРЕЖАХ

Спеціальність 122 Комп’ютерні науки

Кваліфікаційна (бакалаврська) робота

Керівник:
Надія ПОТАПОВА, доцент кафедри
інформаційних технологій,
к. е. н., доцент
_____

Оцінка: _____ / _____ / _____
(бали/за шкалою ЄКТС/за національною шкалою)

Голова ЕК: _____

Вінниця - 2026

АНОТАЦІЯ

Лаптєва М. А. Розробка застосунку виявлення маніпулятивного контенту в соціальних мережах. Спеціальність 122 «Комп'ютерні науки», освітня програма «Комп'ютерні науки». Донецький національний університет імені Василя Стуса, Вінниця 2026.

У кваліфікаційній (бакалаврській) роботі досліджено теоретичні основи інформаційних загроз у соціальних мережах та класифіковано основні види маніпулятивного контенту. Розроблено повностековий вебзастосунок для автоматичного виявлення маніпулятивного контенту. Система реалізує восьмикроковий pipeline аналізу, що охоплює збір даних із соціальних мереж, NLP-класифікацію маніпулятивності, виявлення ботоподібної активності, побудову графу каскаду поширення.

Ключові слова: маніпулятивний контент, обробка природної мови, трансформерні моделі, FastAPI, Independent Cascade Model, соціальні мережі, інформаційна безпека.

76 ст., 15 рис., 5 табл., 2 дод., 40 джерел.

ABSTRACT

Lapteva M. A. Development of an Application for Detecting Manipulative Content on Social Networks. Specialty 122 «Computer Science», educational program «Computer Science». Vasyl Stus Donetsk National University, Vinnytsia 2026.

The bachelor's qualification thesis investigates the theoretical foundations of information threats in social networks and classifies the main types of manipulative content. A full-stack web application for automatic detection of manipulative content has been developed. The system implements an eight-step analysis pipeline encompassing social media data collection, NLP-based manipulation classification, bot-like activity detection, cascade propagation graph construction.

Keywords: manipulative content, natural language processing, transformer models, FastAPI, Independent Cascade Model, social networks, information security.

ЗМІСТ

ВСТУП

В умовах стрімкого розвитку соціальних мереж вони стали основним джерелом інформації для мільйонів користувачів – проте разом із корисним контентом зростає обсяг маніпулятивних технологій, фейкових новин та пропаганди, що спрямовані на деформацію суспільної думки та зниження медіаграмотності. Соціальні платформи стали основним середовищем для формування суспільної думки, а отже і головним інструментом інформаційного впливу. Фейки, емоційні маніпуляції, пропаганда та дезінформація поширюються з надзвичайною швидкістю, завдаючи шкоди як окремим користувачам, так і суспільству загалом. У зв'язку з цим виникає потреба у створенні інструментів, здатних допомагати користувачам ідентифікувати маніпулятивні ознаки в текстах та підвищувати рівень медіаграмотності. Особливо гостро ця проблема постала в умовах інформаційної війни, яку веде Росія проти України. За таких обставин розробка автоматизованих інструментів виявлення маніпулятивного контенту є не лише науково значущим завданням, а й нагальною практичною потребою.

Метою бакалаврської роботи є проєктування та розробка застосунку здатного аналізувати текстовий контент соціальних мереж на предмет наявності маніпулятивних ознак із застосуванням методів машинного навчання та обробки природної мови, а також наочно демонструвати результати аналізу для підвищення рівня критичного сприйняття інформації користувачами.

Об'єктом дослідження є процес автоматизованого аналізу текстового контенту соціальних мереж з метою виявлення ознак маніпулятивного впливу на користувачів.

Предметом дослідження є методи, алгоритми та програмні засоби для класифікації та автоматизованого виявлення лінгвістичних та логічних ознак маніпуляції у текстах, а також програмні засоби реалізації систем аналізу даних.

Для реалізації даного дослідження були поставлені наступні *завдання*:

1. Проаналізувати теоретичні аспекти маніпулятивного впливу в

соціальних мережах, визначити основні види та лінгвістичні ознаки маніпуляцій у текстовому контенті.

2. Обґрунтувати вибір методів аналізу тексту та засобів візуалізації результатів для забезпечення високої точності детекції.

3. Спроєктувати архітектуру застосунку для аналізу маніпулятивного контенту.

4. Програмно реалізувати додаток, забезпечивши функціонал для завантаження тексту, його обробки та візуального виділення маніпулятивних елементів.

5. Провести тестування розробленого застосунку на реальних прикладах контенту із соціальних мереж та оцінити його функціонування.

6. Проаналізувати отримані результати та визначити перспективи подальшого розвитку дослідження.

Основна частина роботи складається із 3 основних розділів. У першому розділі досліджено теоретичні відомості про маніпуляції та її типи. У другому розділі детально розглянуті технології та підходи проєктування застосунку. У третьому розділі наведено опис складових реалізації застосунку для виявлення маніпулятивного контенту в соціальних мережах.

Практична цінність даного дослідження полягає в тому, що розроблений додаток може бути використаний рядовими користувачами соціальних мереж для перевірки наявності маніпуляцій або нейтральності контенту, журналістами й медіааналітиками для моніторингу інформаційного простору, а також освітніми та державними установами в рамках програм із медіаграмотності. Система є масштабованою і може бути адаптована для аналізу контенту різних платформ та мов, що робить її універсальним інструментом протидії інформаційним маніпуляціям.

Апробація результатів даного дослідження була здійснена на III Всеукраїнської науково-практичної конференції «Комп’ютерні технології обробки даних» (м. Вінниця, 8 грудня 2022 року) , IV Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених «Прикладні

інформаційні технології» (м. Вінниця, 18 липня 2023 року), IV Всеукраїнської науково-практичної конференції «Комп’ютерні технології обробки даних» (м. Вінниця, 23 вересня 2024 року), V Всеукраїнської науково-практичної конференції здобувачів, аспірантів та молодих вчених «Прикладні інформаційні технології» (м. Вінниця, 20 лютого 2025 року), V Всеукраїнської науково-практичної конференції «Комп’ютерні технології обробки даних» (м. Вінниця, 31 жовтня 2025 року), Міжнародна наукова інтернет-конференція на тему: "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення", випуск 110 (м. Тернопіль, 7-8 травня 2026 року), журнал «Наука і техніка сьогодні» випуск №5 (м. Київ, 3 червня 2026 року).

РОЗДІЛ 1
ТЕОЕРИЧНІ АСПЕКТИ ВИЯВЛЕННЯ МАНІПУЛЯТИВНОГО КОНТЕНТУ В СОЦІАЛЬНИХ МЕРЕЖАХ

1.1. Сутність та складові інформаційних загроз у соціальних мережах

Соціальні мережі наразі домінують у сучасному медіасередовищі. За даними досліджень, 77,9% українців у 2023 році назвали соціальні мережі (Telegram, Viber, Facebook, YouTube, Instagram, TikTok, Twitter) основним джерелом інформації, тоді як популярність телебачення порівняно з попереднім роком скоротилася на 4%. Зростання обсягів цифрового контенту, розвиток алгоритмів персоналізації та поширення штучного інтелекту сприяли появі нових інформаційних загроз, які мають соціальний, психологічний, політичний та кібербезпековий характер. [1]

Дослідження, засноване на систематичному огляді 150 рецензованих наукових праць, опублікованих між 2014 і 2024 роками, засвідчило, що дезінформація поширюється у шість разів швидше, ніж достовірна інформація, а провідну роль у цьому процесі відіграють емоційний контент та алгоритми платформ [2]. Масштаб явища набув глобального характеру: у світі налічується 4,9 мільярда користувачів соціальних мереж, з яких 45% стикаються з неправдивими новинами саме у Twitter, а 65% – у соціальних мережах загалом [3]. Детальніша статистика зображена на рисунку 1.1.

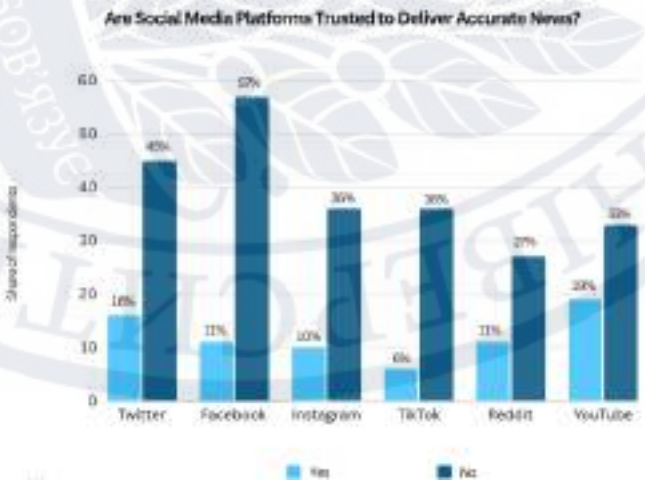

Рисунок 1.1 – Чи можна довіряти новинам на цих платформах

Основною небезпекою сьогодні є не просто наявність неправдивої інформації, а створення цілісних екосистем маніпуляції, які поєднують психологічний вплив, технічні вразливості та алгоритмічну специфіку платформ (Meta, X, TikTok). Особливого загострення ця проблема набуває в умовах інформаційної війни. Маніпуляція інформацією, поширення фейкових новин, дезінформація, пропаганда, створення паніки, підрив віри у перемогу – це лише частина арсеналу, який використовується в межах інформаційної війни. Країна агресор активно залучає соціальні мережі для проведення ІПСО.

Виявляють кілька ключових категорій інформаційних загроз у цифровому середовищі, які відрізняються за механізмом впливу, наміром автора та ступенем шкоди для суспільства. Зведену класифікаційну таблицю наведено у додатку А.

Дезінформація є одним із найпоширеніших і найбільш досліджених видів інформаційних загроз, і перебуває під умовною парасолькою пропаганди й завжди має негативний характер, тобто це завідомо неправдива чи маніпулятивна інформація, у якої завжди на меті нашкодити, і робить вона це системно. Дуже важливим є цей фактор системності: якщо медіа помилилося один раз – це не дезінформація, бо дезінформація – це про системну роботу.

На практиці дезінформація реалізується через широкий спектр форматів: текстові фейки, маніпулятивні зображення та відео, підроблені цитати реальних осіб, сфабриковані «офіційні документи», а також вирвані з контексту достовірні факти, яким надається хибна інтерпретація. Сучасні методи виявлення дезінформації зосереджені на аналізі інформаційного контенту, включно з текстом, зображеннями та відео. Дослідники витягують такі ознаки, як емоційне забарвлення, та застосовують нейронні мережі, зокрема графові нейронні мережі, для характеристики інформації. Крім того, контекст взаємодії користувачів, профілі акаунтів і зовнішні докази слугують корисними сигналами для виявлення дезінформації.

Соціальні мережі значно спростили механізми поширення фейкових повідомлень завдяки алгоритмам рекомендацій, репостам та високій швидкості комунікації між користувачами.

Окремої уваги потребує проблема deepfake-контенту. Розвиток генеративного штучного інтелекту значно спростив створення підроблених відео, фотографій та аудіозаписів, які складно відрізнити від реальних матеріалів і через це зараз, тобто у 2025–2026 роках кількість випадків використання deepfake-технологій у шахрайстві, політичних маніпуляціях та кібербулінгу суттєво зросла. Зокрема, за даними міжнародних медіа, фіксуються випадки створення фейкових зображень політичних діячів, таких як Джорджія Мелоні та інші [4].

Пропаганда є більш широким поняттям, ніж дезінформація, і не завжди оперує неправдивими фактами. Пропаганда – це насадження ідей людям, які можуть мати спільні ознаки за національними, соціальними або політичними вподобаннями. Це певна стратегія, що пропагує систему цінностей, тому можна зазначити про наявність як позитивної, так і негативної пропаганди.

Відмінність від дезінформації полягає передусім у функції: дезінформація оперує фактами (хибними), тоді як пропаганда – ціннісними конструктами і наративами. Онлайн-пропаганда є механізмом впливу на думки користувачів соціальних мереж, і становить зростаючу загрозу для громадського здоров’я, демократичних інституцій та суспільства загалом [5].

З точки зору лінгвістичного аналізу, пропаганда має характерні стилістичні маркери. Вона може бути успішно виявлена через певні стилістичні та синтаксичні особливості, що стоять за відповідними стратегіями – незалежно від ключових слів. Природно, що ключові слова змінюються залежно від перебігу подій, тоді як тактики залишаються схожими.

В основному, в соціальних мережах пропаганда реалізується через:

- масове поширення однотипних повідомлень;
- використання емоційно забарвленої лексики;
- створення образу “ворога”;
- нав’язування поляризованих поглядів;
- використання бот-мереж та координованих кампаній.

Маніпулятивний контент є найширшою категорією серед інформаційних

загроз, оскільки може містити як правдиву, так і хибну інформацію, але завжди з прихованим наміром вплинути на мислення та поведінку аудиторії. На відміну від дезінформації, яка є суто інформаційним обманом, маніпуляція діє передусім через психологічні механізми: апеляцію до емоцій, страху, групової ідентичності, тривоги – приклади таких маркерів наведено у таблиці 1.1 .

Таблиця 1.1 Маркери емоційної маніпуляції [6]

Тип емоції	Мовні маркери (приклади)	Мета маніпуляції
Страх	«загроза», «небезпека», «вже пізно», «вони прийдуть за тобою», «катастрофа», «нас знищать»	Заблокувати критичне мислення через тривогу; спонукати до імпульсивних дій
Гнів / Обурення	«ганьба», «неприпустимо», «доки ми будемо мовчати?», «зрада», «вони знущаються»	Мобілізувати аудиторію; посилити поляризацію «ми проти них»
Провина / Сором	«справжній патріот не може…», «як тобі не соромно», «ти зобов’язаний», «якщо байдуже – ти частина проблеми»	Примусити до певної поведінки через самозвинувачення
Жалість / Співчуття	«безпорадні діти», «ніхто не допомагає», «вони страждають», «поки ти читаєш це…»	Обійти раціональну оцінку; викликати безумовну підтримку
Терміновість / Паніка	«ЗАРАЗ», «тільки сьогодні», «поділися до видалення», «у нас залишилися години»	Спонукати до негайного поширення без перевірки інформації

Однією з причин ефективності маніпулятивного контенту є алгоритмічна природа соціальних мереж – алгоритми рекомендацій самі підсилюють контент,

який викликає сильну емоційну реакцію користувачів, що сприяє швидкому поширенню маніпулятивних повідомлень. Платформи типу TikTok, посилюють ці ефекти через специфіку алгоритмів – платформи з короткими відео залучають увагу миттєво й безпосередньо занурюють глядача у провокативний контент, що підсилює емоційні реакції. Побудова «фільтр-бульбашок» рекомендаційними системами посилює цей ефект.

Сучасні інформаційні атаки характеризуються трьома взаємопов'язаними компонентами: скоординованістю кампаній, масовим використанням генеративного штучного інтелекту у бот-мережах та глибоким психологічним впливом на цільові аудиторії. Ці елементи утворюють замкнений цикл дестабілізації, що важко ідентифікується традиційними засобами кіберзахисту.

Скоординована неавтентична поведінка (Coordinated Inauthentic Behavior, CIB) за визначенням експертів Meta та EEAS (2024) – це використання мереж фейкових облікових записів для маніпулювання публічними дискусіями з метою досягнення стратегічних цілей, що приховуються від аудиторії.

Особливістю 2024 року стала поява кампаній типу "Doppelgänger" (Двійник). Суть цієї технології полягає у створенні точних візуальних копій авторитетних медіа-ресурсів (таких як Le Monde, The Guardian, РБК-Україна), на яких розміщуються фейкові новини з метою надання їм легітимності [7]. За даними звіту European External Action Service, понад 60% виявлених іноземних втручань у 2024 році використовували саме такі методи крос-платформної імітації. Окрім візуальної схожості, нападники використовують техніку "typosquatting" (реєстрація доменів з помилками) та складні редиректи, що дозволяють обходити фільтри перевірки фактів у соціальних мережах.

Важливим аспектом скоординованості є також використання "нативних" рекламних мереж для просування дезінформації. Замість прямих постів від ботів, зловмисники купують таргетовану рекламу, що виглядає як звичайна спонсорська публікація, що дозволяє максимально точно потрапляти в психологічний профіль жертви.

Ключовою рисою сучасних кампаній є "платформний агностицизм" – одна

й та сама дезінформаційна лінія поширюється одночасно в Telegram, TikTok, Facebook та X (Twitter), адаптуючись під алгоритми та аудиторію кожного сервісу. Це створює "ефект правди" у користувача, який бачить ідентичний меседж з різних джерел.

Бот-активність у 2024 році досягла критичної точки – згідно зі звітом Imperva Bad Bot Report 2024, частка "поганих ботів" у глобальному інтернет-трафіку зросла до 32.1%, що є найвищим показником з моменту початку спостережень у 2013 році. Важливо зазначити, що 39.6% цих ботів класифікуються як "просунуті" (advanced), здатні імітувати рухи миші, патерни кліків та поведінку реального користувача.

Революційним фактором стало впровадження Великих Мовних Моделей (LLM). Сучасні боти не просто повторюють хештеги, вони здатні підтримувати осмислені дискусії в коментарях, генерувати унікальні аргументи та емоційно реагувати на критику. Це нівелює класичні методи детекції, засновані на повторюваності тексту. Більше того, у 2025 році зафіксовано випадки використання "автономних агентів", які самостійно планують час публікації, вибирають найбільш резонансні теми дня та взаємодіють між собою для імітації жвавої дискусії без втручання людини-оператора.

Центр протидії дезінформації при РНБО України зазначає, що основною метою атак у 2024–2025 роках є розмиття поняття істини як такої ("Truth Decay"). Коли аудиторія перевантажена суперечливими фактами, вона переходить у стан апатії, що є ідеальним ґрунтом для маніпуляцій. І тому виникає такий феномен, як "інформаційна втома" (Information Fatigue Syndrome). Постійний потік тривожних та суперечливих новин виснажує когнітивні здатості людини – в такому стані критичні фільтри сприйняття вимикаються, і навіть очевидні маніпуляції сприймаються як можлива "альтернативна точка зору".

На відміну від емоційного впливу, когнітивні маніпуляції діють більш приховано – через експлуатацію систематичних відхилень у мисленні, відомих як когнітивні упередження. Когнітивні упередження відіграють ключову роль у тому, як люди сприймають, приймають і поширюють дезінформацію. Ці

систематичні відхилення від раціонального судження глибоко вкорінені в еволюційних адаптаціях і ментальних евристиках та суттєво впливають на сприйняття й оцінку інформації. Одне з найбільш досліджуваних упереджень у цьому контексті – упередження підтвердження [8].

Поняття «фейкові новини» є широким і охоплює суттєво різні за характером та ступенем шкоди різновиди контенту. Найбільш поширеною у міжнародному науковому середовищі є класифікація, розроблена Claire Wardle, яка виокремлює типи контенту залежно від ступеня умисності та рівня фактичної хибності (таблиця 1.3).

Таблиця 1.3 Типологія fake news за критерієм ступеня умисності та фактичності

Тип контенту	Факти	Намір	Характеристика	Типовий приклад
Сфабрикований контент	100% хибні	Умисний	Повністю вигадані факти, розроблені для обману й завдання шкоди	«Вакцини містять мікрочіпи»
Маніпульований контент	Частково хибні	Умисний	Справжні факти або зображення навмисно спотворені для введення в оману	Deepfake-відео з висловлюванням реального політика
Контент-самозванець	Частково хибні	Умисний	Імітація авторитетних джерел — газет, офіційних органів, відомих осіб	Підроблений сайт BBC або НБУ
Хибний контекст	Факти правдиві	Умисний	Достовірні матеріали поширюються з хибним контекстом, датою або авторством	Фото з іншої країни/події подається як актуальне

Продовження табл. 1.3

Тип контенту	Факти	Намір	Характеристика	Типовий приклад
Оманливий контент	Вибіркові факти	Умисний	Вибіркове використання правдивих даних для формування хибного враження	Статистика без контексту, цитата без повного тексту
Хибний зв'язок	Частково хибні	Умисний	Заголовок, зображення або підпис суперечить реальному змісту матеріалу	Клікбейт-заголовок, що не відповідає статті
Сатира / пародія	Хибні, але явно	Ненавмисний	Гумористичний контент без наміру ввести в оману, але може бути сприйнятий буквально	Пародійні новинні сайти, жарти Onion
Міс-інформація	Хибні	Ненавмисний	Поширення хибних відомостей без умислу: помилки, застарілі дані, неперевірені чутки	Перепублікація застарілих новин як актуальних

Слід також звернути увагу і на misleading content (оманливий контент), який технічно може не містити жодної неправдивої факти, але через вибірковість, фреймінг і контекстуальні маніпуляції формує у читача хибне враження про реальний стан речей. Ця класифікація показує, що для оцінки загрози необхідно враховувати не лише зміст, але й намір, контекст, аудиторію та можливі наслідки [9].

Клікбейт є однією з найбільш поширених і найменш технічно складних

форм маніпулятивного контенту. Виявлено кілька типових лінгвістичних патернів клікбейту: гіперболічна мова, емоційні апеляції та неоднозначні формулювання – всі вони навмисно розроблені для провокування допитливості та стимулювання кліків через ефект страху пропустити щось важливе (FOMO). Крім того, клікбейт капіталізує на певних когнітивних упередженнях, зокрема потребі заповнити інформаційний пробіл і схильності людини до новизни.

Соціальні мережі вже впровадили низку стратегій для боротьби з клікбейтом і дезінформацією. Зокрема, позначення твітів як таких, що походять від ботів, або як таких, що містять дезінформацію, знижує позитивне ставлення користувачів до подібного контенту, хоча цей ефект є слабшим серед тих, хто часто користується соціальними мережами. Великі мовні моделі також продемонстрували суттєві можливості у виявленні клікбейту [10].

З погляду автоматичного виявлення, клікбейт має виразні ознаки на рівні тексту. Клікбейт характеризується непропорційно високою емоційною інтенсивністю відносно інформаційного змісту. Сучасні підходи до виявлення клікбейту спираються на модель емоційного простору «валентність-збудження-домінування» (VAD) для моделювання емоційної динаміки, що лежить в основі генерації клікбейту.

1.2. Методи обробки природної мови у задачах аналізу контенту

Обробка природної мови (Natural Language Processing, NLP) є базовою технологічною основою систем виявлення маніпулятивного контенту. Перш ніж текст може бути проаналізований за допомогою алгоритмів машинного навчання, він проходить низку послідовних підготовчих етапів – трансформацій, що перетворюють неструктурований текст на формалізовані числові представлення, придатні для математичної обробки. Цей процес у літературі прийнято позначати терміном «конвеєр попередньої обробки тексту» (text preprocessing pipeline) [11]. Якість виконання кожного з його етапів безпосередньо впливає на точність подальшої класифікації контенту. Найбільш значущими кроками на початкових фазах є токенізація, стемінг та лематизація,

які дозволяють нормалізувати текстові дані та зменшити розмірність словника.

Токенізація є першим і фундаментальним кроком будь-якого NLP-конвеєра. Під токенізацією розуміють процес розбиття суцільного текстового рядка на елементарні одиниці – токени. Залежно від завдання, токеном може виступати слово, підслово (subword), символ або навіть ціле речення.

- Рівень речень**:** Визначення меж висловлювання. Сучасні інструменти використовують нейромережеві моделі для коректної обробки скорочень та абревіатур, які часто плутають із крапкою в кінці речення.
- Рівень слів**:** Класичний підхід, де розділювачами виступають пробіли та пунктуація. Для української мови викликом залишається обробка дефісних слів та апострофів.
- Субслівна токенізація (Subword): Стандарт для сучасних моделей BERT та GPT. Слово "комп’ютеризація" може бути розбите на "комп’ютер" та "изація", що дозволяє ефективно працювати зі словником обмеженого розміру та обробляти рідкісні слова [11].

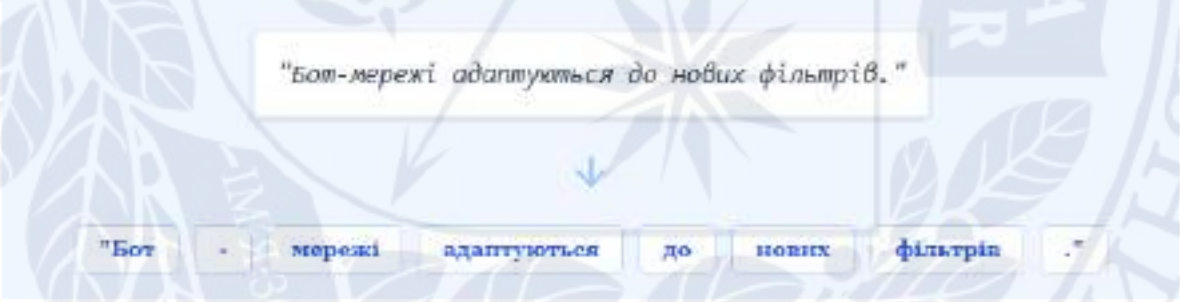

Рисунок. 1.2 – Схема процесу токенізації

Стемінг – це процес приведення слова до його основи шляхом видалення словозмінних та словотворчих елементів без урахування граматичного контексту. Для реалізації стемінгу використовуються спеціалізовані алгоритми, серед яких найбільш поширеними залишаються Porter Stemmer та Snowball Stemmer, що застосовуються у багатьох сучасних NLP-системах для попередньої обробки текстових даних [12].

Принципова особливість стемінгу полягає в тому, що він не спирається на

морфологічний словник: алгоритм механічно застосовує набір правил скорочення суфіксів. Результатом є так звана «основа» – рядок, що не обов'язково збігається з реальним словом. Наприклад, слова «пропаганда», «пропагандист» і «пропагандистський» зводяться до спільного стему «пропаганд», який не є повноцінним українським словом, проте достатньо ідентифікує спільну семантичну основу для потреб порівняння і пошуку. Але через це стемінг може призводити до помилок типу "over-stemming" (різні за змістом слова зводяться до однієї основи) або "under-stemming" (споріднені слова мають різні основи після обробки).

Лематизація – це більш інтелектуальний процес зведення слова до його канонічної, словникової форми (леми). На відміну від стемінг, лематизація враховує частину мови (Part-of-Speech, POS) та контекст вживання слова [11]. Так, усі словоформи «маніпулюю», «маніпулював», «маніпулювали» зводяться до інфінітива «маніпулювати», а «маніпулятивний», «маніпулятивного», «маніпулятивним» – до «маніпулятивний». Ключова відмінність від стемінгу полягає в тому, що результат лематизації завжди є реальним словом природної мови. Для обробки україномовного контенту в системах виявлення маніпуляцій можуть використовуватися морфологічні аналізатори pymorphy2 та pymorphy3, які забезпечують автоматичне визначення нормальної форми слова та його граматичних характеристик [13], а також бібліотека stanza [14].

Серйозним викликом для лематизації в контексті соціальних мереж залишаються навмисні орфографічні спотворення: суржик, транслітерація та цілеспрямовані заміни символів, що застосовуються для обходу автоматичних фільтрів.

1.3. Методи машинного навчання у задачах виявлення маніпулятивного контенту

Класичні алгоритми машинного навчання стали першим інструментарієм для автоматичного виявлення маніпулятивного контенту і досі залишаються конкурентоспроможними в умовах обмежених обчислювальних ресурсів та

невеликих навчальних вибірок.

Логістична регресія (Logistic Regression) є одним із найпоширеніших базових підходів у задачах класифікації тексту. Незважаючи на свою концептуальну простоту, вона демонструє стабільну продуктивність при роботі з TF-IDF-представленнями текстів.

Дослідження 2024 року засвідчило, що логістична регресія демонструє конкурентоспроможну продуктивність у задачах виявлення дезінформації, що відображає її ефективність у задачах класифікації тексту. Ключовою перевагою алгоритму є висока інтерпретованість: ваги окремих ознак безпосередньо вказують, які лексичні патерни модель вважає найбільш діагностичними для маніпулятивного контенту [15].

Метод опорних векторів (Support Vector Machine, SVM) традиційно вважається одним із найефективніших класичних алгоритмів для класифікації текстів. Алгоритм знаходить гіперплощину у багатовимірному просторі ознак, яка максимально розділяє класи через максимізацію відступу (margin) між опорними векторами.

Випадковий ліс (Random Forest) – це ансамблевий методом, що будує множину незалежних дерев рішень на різних підвибірках навчальних даних і агрегує їхні передбачення шляхом голосування. У контексті виявлення маніпулятивного контенту він ефективно працює з гетерогенними наборами ознак, поєднуючи лінгвістичні, статистичні та метадатні характеристики публікацій. Механізм обчислення важливості ознак (feature importance) дозволяє виявляти найдіагностичніші предиктори маніпулятивності, наприклад, частоту емоційно маркованих слів, довжину речень, наявність гіперболічних конструкцій. Порівняння основних алгоритмів наведена у таблиці 1.4.

Першою хвилею глибокого навчання для класифікації тексту стали рекурентні архітектури, зокрема мережі з довгою короткостроковою пам’яттю (Long Short-Term Memory, LSTM) – на відміну від класичних методів, LSTM здатний враховувати послідовну структуру тексту й зберігати контекст на відстані кількох десятків токенів, що робить його ефективними при виявленні

маніпулятивних конструкцій, де ключове слово може знаходитися далеко від контексту, який аналізується.

Таблиця 1.4 Порівняльна характеристика класичних ML-алгоритмів у задачах виявлення маніпулятивного контенту

Характеристика	Logistic Regression	SVM	Random Forest
Принцип	Лінійна межа + сигмоїда	Максимізація margin	Ансамбль дерев рішень
Інтерпретованість	Висока	Середня	Середня
Вимоги до даних	Низькі	Середні	Середні
Стійкість до шуму	Середня	Висока	Висока
Типова точність (accuracy)	95-99%	97-100%[16]	99-99,2%[16]
Робота з дисбалансом	Потребує коригування	Потребує коригування	Задовільна

Революція у виявленні маніпуляцій відбулася з появою механізму Self-Attention (самоуваги), який лежить в основі моделей Transformer. На відміну від рекурентних мереж (RNN), трансформери аналізують усі слова в реченні одночасно, встановлюючи контекстні ваги для кожного токена. Для задач виявлення маніпуляцій це принципово важливо оскільки маніпулятивні тексти часто будуються на тонких семантичних зв’язках між різними частинами речення або абзацу, які класичні методи і навіть LSTM не здатні вловити.

Модель BERT (Bidirectional Encoder Representations from Transformers) працює за принципом двостороннього аналізу тексту. Це означає, що вона одночасно враховує як попередні, так і наступні слова у реченні. Наприклад, слово може мати різні значення залежно від контексту, і BERT здатний це врахувати. Під час навчання модель отримує завдання відновити пропущені слова у тексті, що дозволяє їй “зрозуміти” зв’язки між словами та їх роль у реченні. Завдяки цьому BERT добре підходить для задач класифікації тексту,

визначення тональності та виявлення маніпулятивних конструкцій [17].

Модель RoBERTa (Robustly Optimized BERT Approach) є вдосконаленою версією BERT. Основна відмінність полягає не стільки у принципі роботи, скільки у способі навчання – RoBERTa тренується на більшому обсязі даних, довше та без деяких обмежень, які були в самого BERT. Вона також використовує динамічне маскування слів під час навчання, що дозволяє краще узагальнювати знання. У результаті RoBERTa ефективніше працює з різними типами текстів і краще адаптується до нових даних [17].

Окремим напрямок розвитку це – мультимовні моделі, такі як XLM-RoBERTa, які здатні працювати з текстами різними мовами. Це є критично важливим для аналізу соціальних мереж, де інформація поширюється одночасно кількома мовами. Такі моделі дозволяють виявляти маніпулятивний контент незалежно від мовного середовища.

Також перспективним напрямом є застосування великих мовних моделей (LLM), які можуть виконувати класифікацію текстів навіть без спеціального донавчання (zero-shot) або з мінімальною кількістю прикладів (few-shot). Це відкриває можливості для швидкого впровадження систем аналізу без необхідності створення великих розмічених датасетів.

Незважаючи на прогрес у розробці методів виявлення маніпулятивного контенту, існуючі підходи мають принципові обмеження, що передбачають необхідність подальших досліджень.

Основна складність полягає у властивій природній мові полісемії, омонімії та залежності значення від контексту. Одне й те саме речення може бути маніпулятивним або нейтральним залежно від того, хто його вимовляє і в якому дискурсивному оточенні. Класичні ML-моделі, що оперують локальними лексичними ознаками, не здатні розрізнити ці контексти, а маніпулятори користуються цим і часто використовують "м’яку" дезінформацію – тексти, де кожне речення формально правдиве, але загальний висновок спотворює реальність через пропуск важливих фактів. Моделі NLP зазвичай оцінюють лише локальні зв’язки, не помічаючи глобальної маніпулятивної структури.

Але найбільше проблем для автоматичних систем є мовні явища, пов’язані з непрямим вираженням змісту. До них належать сарказм, іронія та підтекст, які широко використовуються в соціальних мережах як інструменти як комунікації, так і маніпуляції. Сарказм є формою висловлювання, при якій справжній зміст протилежний до буквального. Наприклад, фраза «Чудова новина, просто ідеально!» може фактично означати негативну оцінку ситуації. Для людини така інтерпретація є інтуїтивною завдяки знанню контексту, інтонації або попереднього досвіду, тоді як алгоритм зазвичай сприймає її як позитивну.

Іронія є більш тонкою формою непрямого вираження, ніж сарказм. Вона не завжди має чітко протилежний зміст, а часто створює подвійне значення. Наприклад, висловлювання може одночасно містити як буквальний, так і прихований сенс.

Для автоматичних систем складність полягає в тому, що іронія:

- часто не порушує граматичних або лексичних норм;
- не містить явних ознак маніпуляції;
- вимагає розуміння ширшого контексту (соціального, культурного, ситуаційного).

Підтекст є ще більш складним явищем, оскільки він взагалі не виражається прямо в тексті – а характеризується використанням натяків, метафор або алегорій, які зрозумілі людині через спільний культурний бекграунд, але сприймаються алгоритмом як звичайний опис подій.

Зазначені обмеження зумовлюють перехід від суто текстового аналізу до комплексних підходів, що поєднують кілька категорій ознак. Оскільки більшість сучасного контенту в соціальних мережах включає як візуальні, так і текстові елементи, їхнє спільне використання дозволяє моделям виявляти суперечності, які не є очевидними при аналізі тільки з однієї сторони. Перехід від простого текстового контенту до складніших форматів, що включають зображення, аудіо та відео, потребує більш просунутих методів виявлення, здатних інтегрувати різні типи даних. Ефективна протидія гібридним загрозам у 2025–2026 роках вимагає переходу від ізольованого аналізу змісту до мультимодальних стратегій,

що поєднують лінгвістику, поведінкову психологію та аналіз графів поширення. Тому доцільним є:

- поєднання семантичного аналізу тексту з аналізом поведінкових даних користувачів;
- врахування контексту поширення інформації (час, джерело, мережеві зв’язки);
- використання гібридних моделей, що комбінують машинне навчання та rule-based підходи;
- інтеграція метаданих та історії взаємодії користувачів.

1.4. Аналіз існуючих систем модерації контенту

Ефективність боротьби з інформаційними загрозами безпосередньо залежить від архітектури систем модерації, які впроваджують власники платформ. Сучасна модерація пройшла шлях від ручного перегляду скарг до складних автоматизованих комплексів, що працюють у реальному часі, проте їхнє функціонування супроводжується низкою системних проблем, тому аналіз підходів трьох ключових платформ, X (Twitter), Meta та YouTube – дозволяє виявити загальні тенденції та принципові відмінності в організації автоматизованих систем виявлення маніпулятивного контенту.

X – після переходу платформи під управління Ілона Маска у 2022 році система модерації зазнала кардинальних змін. Платформа відмовилась від значної частини штату модераторів-людей та перейшла до гібридної моделі, що поєднує автоматизовані алгоритми. Замість класичного видалення контенту або блокування користувачів, сама спільнота додає контекст до потенційно маніпулятивних постів, якщо користувачі з різними поглядами погоджуються, що нотатка є корисною, вона відображається під твітом. Попри високу прозорість, така система має затримку (time-to-note), що дозволяє маніпулятивному контенту ставати вірусним до моменту появи спростування.

Відповідно до DSA-звіту про прозорість, X застосовує комбінацію евристик і алгоритмів машинного навчання для автоматичного виявлення

контенту, що порушує правила платформи: використовуються моделі обробки природної мови, моделі обробки зображень та інші методи машинного навчання. Вхідними даними для моделей є текст публікації, прикріплені зображення та інші характеристики. Перед запуском будь-якого алгоритму платформа верифікує його ефективність, відбираючи статистично значущу тестову вибірку та проводячи поелементну перевірку людиною [18].

У Глобальному звіті про прозорість за перше півріччя 2024 року X повідомив, що 0,0123% його публікацій порушували правила платформи за цей період – це одна публікація на кожні 10 000. Найпоширенішими категоріями порушень стали: агресивна поведінка, зловживання та переслідування та контент із зображенням насильства [19].

Meta (Facebook та Instagram) – побудувала найбільш розгалужену серед соціальних платформ інфраструктуру модерації, що поєднує автоматизовані класифікатори, великі мовні моделі, людський огляд і механізм апеляцій через наглядову раду. Із величезної кількості контенту, опублікованих на Facebook та Instagram у третьому кварталі 2025 року, менше 1% видалено за порушення правил платформи і менше 0,1% видалено помилково. Точність правозастосування для видаленого контенту перевищила 90% на Facebook та 87% на Instagram. У 2025 році Meta видалила понад 159 мільйонів шахрайської реклами, з яких 92% було заблоковано до появи будь-яких скарг [20].

YouTube ж орієнтований на мультимедійний контент, тому модерація базується на автоматизованому аналізі аудіодоріжок (через технології Speech-to-Text) та відеоряду на етапі завантаження. Платформа застосовує жорстку політику демонетизації та обмеження рекомендацій для каналів, які систематично поширюють теорії змови або медичну дезінформацію, спрямовуючи користувачів на панелі з посиланнями на авторитетні джерела. Протягом 2024-2025 років YouTube переписав свої правила для протидії сплеску deepfake-імітацій, шахрайства з клонуванням голосу та ШІ-редагувань, що розмивають автентичність [21]. Генеральний директор YouTube Ніл Мохан заявив, що можливості ШІ-модерації покращуються «буквально щотижня» і

допомагають «виявляти порушення точніше, із більшою спроможністю відповідати на масштаб». Водночас ця заява пролунала на тлі повідомлень авторів про щоденні випадки неправомірного видалення каналів автоматизованими системами, що відображає зростаючий скептицизм щодо надійності ШІ-рішень у питаннях, що зачіпають засоби до існування людей.

Автоматизована модерація неминуче генерує помилкові видалення правомірного контенту. У лютому 2024 року Meta отримала понад сім мільйонів апеляцій від користувачів, чий контент було видалено за правилами щодо мови ворожнечі. Восьмеро з десяти тих, хто подав апеляцію, скористалися можливістю надати додатковий контекст. Кожен п’ятий із цих користувачів зазначив, що контент мав на меті «підвищити обізнаність», тоді як кожен третій вказав, що це «був жарт». У IV кварталі 2025 року Meta зафіксувала різке зростання хибнопозитивних видалень більш ніж на 100% протягом короткого часу через помилку в алгоритмі. Цей інцидент підкреслює крихкість навіть найбільш розвинених систем автоматизованої модерації.

На X проблема хибних пропусків незафіксованого порушуючого контенту є ще гострішою. На початку 2024 року ШІ-згенеровані зображення сексуального насилля над відомою співачкою набрали понад 27 мільйонів переглядів і 260 000 вподобань до призупинення акаунту-порушника. X виявився нездатним зупинити подальше поширення матеріалів навіть після блокування початкового акаунту, що підтвердило: системи контентної модерації платформи не були достатніми для стримування проблеми в реальному часі.

Як і згадувалось раніше – принципове обмеження автоматизованих систем – це нездатність враховувати соціальний, культурний та ситуативний контекст публікації. Хибні новини поширюються швидко і тому потребують перевірки у швидшому темпі, однак більшість традиційних систем фактчекінгу нездатні функціонувати в реальному часі, на кількох платформах, різними мовами і в різних культурних контекстах. Повідомлення може бути помилковим, але не становити значної загрози через низьке охоплення; навпаки, частково правдиве повідомлення може бути небезпечним, якщо воно подане в маніпулятивному

контексті та адресоване вразливій аудиторії в умовах поляризації [9].

Іншим системним недоліком є нерівномірне охоплення мов у системах модерації. Найбільші та найшвидше зростаючі бази користувачів соціальних мереж зосереджені у Глобальному Півдні, де мільярди людей генерують контент своїми мовами. Технологічні компанії схильні надавати пріоритет модерації для англомовних користувачів Заходу, залишаючи шкідливий контент іншими мовами значною мірою поза увагою.

Мультимовні мовні моделі навчаються на даних різних мов одночасно, однак частина даних для низькоресурсних мов є перекладеною або неправильно перекладеною чи зібраною з неякісних джерел – що призводить до неефективної автоматизованої модерації. Проблема нестачі ресурсів пояснюється не лише дефіцитом даних, а й економічними та політичними чинниками, браком людської експертизи та обмеженим доступом до цифрової інфраструктури.

Дослідження 2024 року, що є найбільш вичерпною оцінкою реальних обмежень зусиль з фактчекінгу під час виборів, зафіксувало три стійкі дефіцити:

- вибірковість – на жаль лише незначна частка дезінформаційних наративів отримує відповідний фактчек;
- швидкість – перевірки публікуються із запізненням, коли більша частина охоплення вже зафіксована;
- охоплення аудиторії – спростування рідко досягають тих, хто бачив оригінальний неправдивий контент [22].

Розглянемо уже наявні аналоги. Усі наявні рішення можна згрупувати у чотири категорії: інструменти фактчекінгу, браузерні розширення для оцінки достовірності джерел, інструменти виявлення ШІ-контенту та дослідницькі прототипи систем виявлення фейків.

Інструменти фактчекінгу. До найбільш авторитетних належать Snopes, PolitiFact, FactCheck.org та Reuters Fact Check. У 2025 році Snopes дебункував понад 22 000 тверджень, партнерує з Facebook для перевірки в реальному часі та адресує питання deepfakes під час виборів. PolitiFact використовує шкалу «Truth-O-Meter» з шістьма градаціями правдивості і спеціалізується на перевірці

висловлювань публічних осіб [23]. Принциповим обмеженням цих інструментів є ручна природа роботи: вони не здатні масштабуватися до обсягів контенту соціальних мереж і перевіряють переважно англомовні матеріали.

Браузерні розширення. NewsGuard і Media Bias/Fact Check (MBFC) оцінюють достовірність новинних джерел, а не конкретного контенту. MBFC оцінює «політичну упередженість» та «фактографічність» медіаджерел, спираючись на комбінацію об'єктивних вимірів і суб'єктивного аналізу. Рейтинги MBFC залишаються еталонними у численних дослідженнях виявлення упередженості джерел, класифікації фейкових новин та систем автоматичної перевірки фактів. Дослідження 2022 року, що аналізувало поширення URL-посилань у Twitter і Facebook, встановило, що рейтинги MBFC демонструють сильну кореляцію з оцінками NewsGuard (r = 0,81), що підтверджує придатність обох інструментів як взаємозамінних проксі-мір достовірності новинних джерел [24]. Порівняльний аналіз наведено в таблиці 1.5.

Таблиця 1.5 Порівняльний аналіз існуючих рішень виявлення маніпулятивного контенту

Критерій	Snopes / PolitiFact	NewsGuard / MBFC	GPTZero / Originality	Запропонований застосунок
Тип аналізу	Ручна перевірка фактів	Оцінка джерела	Виявлення ШІ-авторства	Класифікація маніпулятивного контенту
Реальний час	Ні	Частково	Так	Так
Підтримка укр. мови	Ні	Обмежено	Ні	Так (цільова мова)
Аналіз контенту соцмереж	Ні	Ні	Частково	Так
Пояснюваність результату	Так (ручна)	Частково	Обмежена	Так (клас + ймовірність)
Доступність як веб-застосунок	Так	Так (розширення)	Так	Так
Безкоштовний доступ	Так	Обмежено	Обмежено	Так
Орієнтація на маніпуляції	Частково	Ні	Ні	Так

Інструменти виявлення ШІ-контенту. GPTZero, Originality.ai, Copyleaks та Winston AI орієнтовані на виявлення ШІ-авторства тексту, а не маніпулятивного контенту як такого. Ринок виявлення ШІ-контенту прогнозовано перевищить 1 млрд доларів до 2028 року, проте поточні інструменти мають суттєві обмеження: точність часто нижче 50%, особливо для змішаного або перефразованого контенту. Використання кількох інструментів одночасно підвищує надійність, але жодне єдине рішення не є надійним у ситуаціях з високими ставками [26].

Підходи, засновані на NLP, не враховують особливості поширення інформації, тоді як графові методи не дозволяють оцінити зміст повідомлень. Жоден із цих підходів окремо не забезпечує повного розв’язання задачі виявлення маніпулятивного контенту, що обумовлює необхідність розробки інтегрованих систем [27]. Аналіз таблиці також засвідчує, що жодне з наявних рішень не задовольняє одночасно всім ключовим вимогам: підтримки україномовного контенту, роботи в реальному часі, орієнтації саме на маніпулятивні патерни і відкритого безкоштовного доступу.

РОЗДІЛ 2
РОЗРОБКА МОДЕЛІ ТА АЛГОРИТМУ ВИЯВЛЕННЯ МАНІПУЛЯТИВНОГО КОНТЕНТУ

2.1. Концептуальні засади побудови моделі виявлення маніпулятивного контенту

Підчас розробки застосунку основна увага приділялась створенню моделі, яка може аналізувати не тільки зміст повідомлення, але й особливості його поширення у середовищі соціальної мережі. Такий підхід враховує, що маніпулятивний контент не завжди можна визначити тільки за допомогою текстових ознак, оскільки на практиці значна частина інформаційних загроз має прихований характер – повідомлення може виглядати нейтральним, однак спосіб його розповсюдження може вказувати на координовану активність або штучне підсилення [30].

У більшості з систем виявлення дезінформації основний акцент робиться на методах обробки природної мови – для аналізу використовуються трансформерні моделі або ж алгоритми текстової класифікації [31]. Подібні підходи дозволяють визначати емоційно забарвлені конструкції(речення), клікбейтні заголовки, токсичність або ознаки маніпулятивного впливу. Разом із тим аналіз лише текстового вмісту має певні обмеження – наприклад, однакові повідомлення можуть мати різний рівень ризику залежно від того, яким способом воно поширюється та які групи користувачів беруть участь у його розповсюдженні.

Спираючись на вищезазначене у розробленій системі вирішено використовувати інтегрований підхід, що поєднує два напрями аналізу:

- семантичний аналіз повідомлення;
- аналіз поведінкових та структурних характеристик поширення.

Семантичний компонент в моделі відповідає за аналіз та дослідження текстового вмісту повідомлення – у межах цього етапу система виконує попередню обробку тексту, нормалізацію даних, побудову ембедінгів, а також

аналіз емоційного забарвлення, рівня токсичності та наявності маніпулятивних мовних конструкцій.

Результатом роботи семантичного модуля є формування вектора ознак:

$$S = \{s_1, s_2, \dots, s_n\} \quad (2.1)$$

де s_i – параметри семантичного аналізу повідомлення.

Окремо система виконує аналіз поведінкових характеристик поширення повідомлення: взаємодія між користувачами подається у вигляді графової структури, де вузли відповідають користувачам або інформаційним джерелам, а зв'язки відображають поширення повідомлень, репости, коментарі чи інші форми взаємодії. Подібне представлення широко використовується для аналізу інформаційних каскадів та дослідження поширення дезінформації в соціальних мережах [32]. На основі отриманих даних система оцінює інтенсивність поширення повідомлення, рівень активності користувачів та характер взаємодій між акаунтами.

$$G = \{g_1, g_2, \dots, g_m\} \quad (2.2)$$

де g_i – параметри структурного аналізу.

Таке моделювання базується на положенні, що будь-які соціальні платформи та мережі в своїй основі є дискретними структурами, а інструменти теорії графів є математичним фундаментом для побудови алгоритмів кібербезпеки та аналізу трафіку [33].

Ключовою ідеєю є об'єднання результатів семантичного та структурного аналізу в одну систему оцінювання ризику. Інтегральна оцінка повідомлення визначається такою функцією:

$$T(m) = F(S, G) \quad (2.3)$$

де S – семантичний вектор ознак;

G – структурний вектор характеристик поширення;

$T(m)$ – інтегральна оцінка ризику повідомлення.

Такий підхід зменшує вплив обмежень використання поокремості методів

аналізу, якщо текстове повідомлення не містить явних ознак маніпуляції, система може врахувати нетипову поведінку поширення або підозрілу активність взаємодій. Також наприклад: емоційний характер повідомлення не є достатньою підставою для класифікації посту або повідомлення як маніпулятивного без додаткового аналізу поведінкових та інших характеристик.

Важливим етапом побудови моделі було розділення процесу аналізу на окремі функціональні етапи: обробку тексту, аналіз поведінкових характеристик та формування підсумкової оцінки ризику. Архітектура вебзастосунку зроблена за модульним принципом – основною метою такої структури є забезпечення незалежної роботи окремих компонентів, можливості масштабування та підтримки подальшого розширення функціоналу без зміни загальної логіки застосунку. Можна виділити декілька основних компонентів архітектури:

- модуль збору даних;
- backend-компонент;
- модуль виявлення маніпулятивності;
- модуль аналізу ботоподібної активності;
- модуль аналізу поширення;
- модуль оцінювання ризику;
- frontend-компонент візуалізації результатів.

Загальну структуру взаємодії компонентів системи наведено на рис 2.1. Першим етапом роботи застосунку є отримання даних із зовнішніх джерел. Для цього використовується scraper-модуль, реалізований із використанням бібліотек Playwright та BeautifulSoup. Модуль дозволяє отримати текстовий контент, часові мітоки, реакцій користувачів та службові метадані з соціальної мережі. У системі передбачено декілька режимів збору інформації, що дозволяє працювати як із повними даними сторінки, так і з обмеженим набором метаданих. Після того як отриманні данні – інформація передається до backend-компоненту, реалізованого за допомогою FastAPI [34]. Backend-сервер відповідає за координацію роботи окремих модулів системи, обробку API-запитів, маршрутизацію даних та для формування підсумкових результатів аналізу. Для

обміну інформацією між компонентами є JSON-структури та REST API.

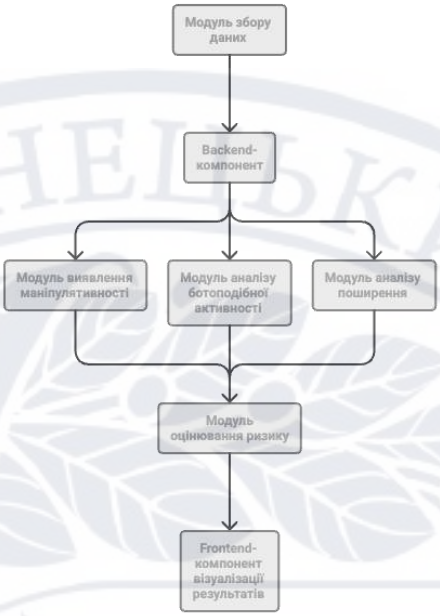

Рисунок 2.1 – Архітектура застосунку

Важливим компонентом є модуль виявлення маніпулятивності, що аналізує текст повідомлення, у його межах використовуються NLP-методи, ембендінгс та трансформерні моделі для формування контекстного представлення тексту [31]. В результаті модуль формує оцінки маніпулятивності повідомлення та додаткові характеристики.

Окремо аналізується ботоподібна активність акаунтів, для цього використовуються характеристики профілів, часові особливості активності та статистичні параметри взаємодій, якщо цю інформацію можна отримати. В результаті цей модуль дозволяє визначати ознаки автоматизованого поширення інформації та враховувати їх під час подальшої оцінки ризику.

Модуль аналізу поширення працює з графом взаємодій між користувачами для формування характеристик динаміки розповсюдження повідомлення. На цьому етапі оцінюється активність поширення, інтенсивність взаємодій та наявність нетипових шаблонів поведінки.

Після завершення аналізу результати передаються до модуля оцінювання ризику, який виконує інтеграцію всіх отриманих характеристик та формує підсумкову оцінку рівня потенційної інформаційної загрози. Додатково система

враховує рівень достовірності та повноти отриманих даних, що дозволяє підвищити надійність результатів аналізу.

Frontend-компонент реалізується за допогою React та Vite. Інтерфейс користувача забезпечує відображення результатів аналізу, візуалізацію характеристик поширення повідомлення, відображення оцінки ризику та формування аналітичних звітів.

2.2. Модуль семантичного аналізу тексту повідомлення

Один із головних етапів це попередня обробка текстового вмісту повідомлень, через те що повідомлення та інформація взята з соціальних мереж, зазвичай, містить велику кількість шумових даних та допоміжних елементів, що можуть негативно впливати на результати подальшого аналізу [35]. На практиці повідомлення часто містять URL-посилання, смайли, спеціальні символи, скорочення – тобто елементи, що не несуть суттєвого смислового навантаження для задачі виявлення маніпулятивного контенту. Загальну послідовність етапів попередньої обробки тексту наведено на рисунку 2.2.

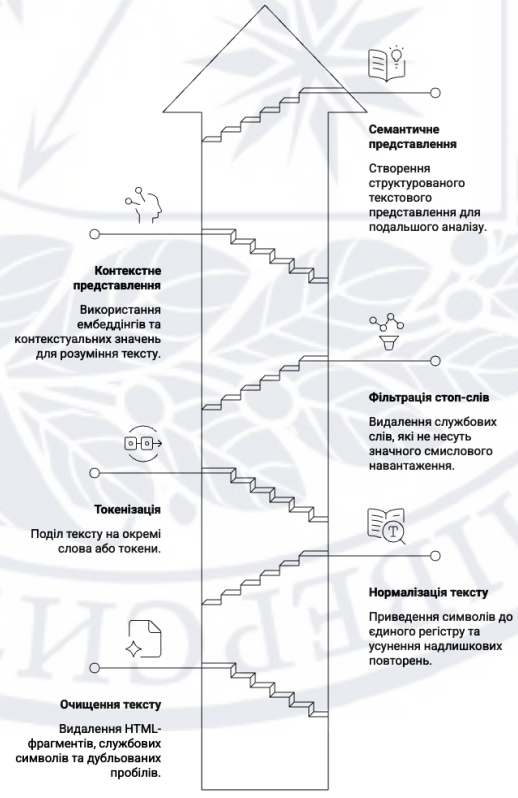

Рисунок 2.2 – Етапи попередньої обробки тексту

Попередня обробка виконується, як окремий етап перед формуванням текстових ознак та застосуванням моделей семантичного аналізу, мета цього етапу це приведення тексту до уніфікованого вигляду, зменшення впливу шумових даних та підготовка повідомлення до подальшого аналізу.

Першим етапом обробки є очищення тексту від службових елементів – з повідомлення видаляються HTML-фрагменти, службові символи, дубльовані пробіли та технічні елементи форматування, окремо виконується обробка URL-посилань, згадок користувачів та хештегів. У межах системи такі елементи можуть або повністю видалятися, або замінюватися спеціальними токенами залежно від режиму аналізу. Подібний підхід дозволяє зменшити кількість шумових ознак та зберегти структуру повідомлення [36].

Після очищення текст переходить на другий етап – етап нормалізації. Під час нього символи приводяться до єдиного регістру, усуваються надлишкові повторення символів та виконується базова уніфікація тексту. Необхідність такої обробки пояснюється тим, що користувачі часто використовують нестандартні написання слів, велику кількість емоційних повторів або комбінації символів для підсилення емоційного забарвлення повідомлення.

Третім етапом є токенізація – процес поділу тексту на окремі елементи або токени. Токенізація використовується для формування структурованого представлення повідомлення, яке може бути використане під час подальшого аналізу. Після цього додатково виконується фільтрація стоп-слів – службових слів, які зазвичай не несуть значного смислового навантаження та можуть негативно впливати на ефективність аналізу тексту [37].

Важливою особливістю є використання контекстного представлення тексту. Замість класичних підходів на основі лише частоти слів у повідомленні застосовуються ембеддінги та контекстуальні значення. Це особливо важливо для виявлення маніпуляцій, оскільки однакові слова або фрази можуть мати різне смислове навантаження залежно від контексту їх використання. Для формування контекстного представлення тексту використовуються трансформерні NLP-моделі, здатні аналізувати взаємозв’язки між словами у повідомленні та

враховувати контекст сусідніх фрагментів тексту [31]. Подібний підхід дозволяє покращити якість подальшого аналізу емоційного забарвлення, токсичності та маніпулятивних мовних конструкцій.

Результатом виконання попередньої обробки є структуроване текстове представлення повідомлення, яке дозволяє зменшити вплив шумових даних, уніфікувати текстовий вміст повідомлень та придатне для подальшого формування семантичних ознак та роботи модулів класифікації.

Після завершення етапу попередньої обробки можна переходити до аналізу семантичних характеристик повідомлення – для виявлення мовних факторів, що можуть вказувати на наявність маніпулятивного впливу, емоційного тиску або інших форм впливу на користувача. На відміну від класичних підходів, що базуються лише на пошуку окремих ключових слів, аналізується загальний контекст повідомлення та взаємозв’язок між окремими текстовими конструкціями [36]. До основних семантичних ознак, що враховуються на цьому етапі, належать:

- емоційне забарвлення повідомлення;
- рівень токсичності;
- наявність clickbait-конструкцій;
- ознаки маніпулятивного впливу;
- контекстні семантичні характеристики тексту.

Основні компоненти семантичного аналізу наведено на рисунку 2.3.

Рисунок 2.3 – Семантичний аналіз характеристик повідомлення

Одним із базових компонентів семантичного аналізу є визначення емоційного забарвлення повідомлення (sentiment analysis). На цьому кроці система оцінює загальну емоційну спрямованість тексту, а також наявність виражених негативних або провокативних елементів. Необхідність такого аналізу знову ж таки пов’язана з тим, що маніпулятивний контент часто орієнтований на виклик емоційної реакції користувача. Подібні повідомлення зазвичай характеризуються підвищеною емоційністю та використанням мовних конструкцій із сильним емоційним навантаженням [38].

Окремо потрібно виконати аналіз токсичності тексту – визначити ознаки агресивної, образливої або конфліктної комунікації. Аналіз токсичності не використовується як самостійний критерій виявлення маніпулятивності, однак дозволяє враховувати додаткові характеристики повідомлення під час формування загальної оцінки ризику. На практиці маніпулятивні інформаційні кампанії часто супроводжуються використанням агресивної риторики або навмисним провокуванням конфліктних дискусій у коментарях та публікаціях.

Ще одною частиною аналізу є виявлення клікбейт-конструкцій та елементів емоційного підсилення, для яких характерне використання перебільшень, провокативних заголовків, або створення відчуття терміновості. Подібні мовні конструкції використовуються для збільшення залученості користувачів та штучного підвищення активності поширення контенту [39]. Також аналізуються маніпулятивні мовні шаблони та когнітивні тригери. Аналіз таких конструкцій дозволяє враховувати не лише окремі слова або фрази, але й загальну логіку побудови повідомлення. Для підвищення якості аналізу система використовує контекстне представлення тексту, сформоване на попередньому етапі обробки. Це дозволяє враховувати сенсові залежності між окремими фрагментами повідомлення та зменшити кількість помилкових спрацювань під час класифікації. Особливо важливим це є у випадках, коли окремі слова можуть мати різне значення залежно від контексту використання.

Таким чином за допомогою модулю семантичного аналізу отримується набір характеристик, які відображають змістовні та емоційні особливості

повідомлення.

Після семантичного аналізу результати обробки тексту потрібно привести до уніфікованого вигляду, для подальшого використання у модулі оцінювання ризику. Для цього отримані характеристики агрегуються у семантичний вектор ознак S, загальний вигляд якого було наведено у підрозділі 2.1. Формування семантичного вектора дозволяє представити результати аналізу повідомлення у вигляді набору параметрів, що характеризують змістовні, емоційні та поведінкові особливості тексту. Подібний підхід спрощує подальшу інтеграцію результатів в межах єдиної системи оцінювання ризику.

До складу семантичного вектора входять результати аналізу характеристик, розглянутих на попередньому етапі, зокрема параметри емоційного забарвлення, токсичності, маніпулятивних мовних конструкцій, клікбейт-ознак та контекстного представлення тексту.

Для деталізації структури семантичного вектора, загальний вигляд якого було наведено у формулі (2.1), процес його формування можна подати у вигляді:

$$S = f(E, T, C, M, K) \tag{2.4}$$

де E – характеристики емоційного забарвлення повідомлення;

T – параметри токсичності тексту;

C – показники наявності клікбейт-конструкцій;

M – характеристики маніпулятивних мовних шаблонів;

K – контекстні семантичні характеристики повідомлення.

Після виконання аналізу окремі параметри приводяться до нормалізованого вигляду. Нормалізація дозволяє усунути відмінності між діапазонами значень різних характеристик та забезпечити коректне функціонування алгоритмів машинного навчання [37]. Окремі характеристики можуть додатково зважуватись залежно від режиму аналізу та повноти доступних вхідних даних. Наприклад, у випадку недостатнього обсягу текстового контенту окремі значення можуть мати менший вплив на підсумковий результат оцінювання, подібний підхід дозволяє підвищити

стійкість системи до неповних даних.

Семантичний вектор не використовується як самостійний критерій класифікації повідомлення – отримані характеристики передаються до модуля інтегрованого оцінювання ризику, де поєднуються зі структурними та поведінковими характеристиками поширення контенту.

2.3. Аналіз поведінкових характеристик поширення повідомлення

Наступним ключовим етапом є аналіз особливостей поширення повідомлення у соціальній мережі. Для задачі виявлення маніпулятивного контенту значення має не лише зміст повідомлення, а й характер взаємодій навколо нього: швидкість появи реакцій, інтенсивність поширення, активність акаунтів, можливі ознаки штучного підсилення. Саме тому у розроблюваній системі поведінковий аналіз використовується як додатковий рівень оцінювання ризику.

У реальних умовах поведінкові характеристики дають змогу уточнити результат семантичного аналізу або виявити критичні моменти, які за допомогою нього складно було б виявити, навіть за відсутності явних маніпулятивних мовних конструкцій нетипова активність акаунтів або аномальні шаблони взаємодії можуть бути додатковими сигналами для оцінювання ризику [40].

Взаємодії між користувачами подаються у вигляді мережевої структури, де акаунти відповідають вершинам графа, а інформаційні взаємодії між ними – ребрам. До таких взаємодій можуть належати репости, коментарі, реакції, або інші форми взаємодії користувачів, які сприяють поширенню повідомлення. На основі отриманих даних можливо сформувати поведінкові характеристики поширення та оцінює динаміку інформаційної активності.

Одним із основних параметрів аналізу – інтенсивність взаємодій навколо повідомлення. У звичайній ситуації поширення інформації відбувається поступово та залежить від інтересу аудиторії, але для маніпулятивних інформаційних кампаній можуть бути характерними різкі сплески активності, велика кількість однотипних реакцій або надмірна концентрація взаємодій у

короткому часовому інтервалі. Подібні особливості можуть свідчити про штучне підсилення видимості контенту або використання автоматизованих механізмів поширення.

Важливу роль у межах поведінкового аналізу відіграють часові характеристики активності – система враховує швидкість появи перших реакцій після публікації повідомлення, нерівномірність поширення інформації та зміну інтенсивності взаємодій у часі. Для безпосереднього дослідження структури утворених каскадів та детекції кількості шляхів поширення між обліковими записами використовуються базові графові алгоритми обходу, такі як пошук у глибину (DFS) та пошук у ширину (BFS), що мають часову складність O(V+E) та забезпечують ефективне обчислення метрик зв'язності на великих обсягах даних [41]. Якщо повідомлення за короткий проміжок часу отримує велику кількість однотипних реакцій, це може бути додатковим сигналом підозрілої активності, особливо коли така поведінка не відповідає типовим характеристикам поширення.

Окремо аналізується характер взаємодій між акаунтами, якщо повідомлення поширюється переважно через невелику групу профілів із подібними часовими шаблонами активності або повторюваними сценаріями поведінки, система може розглядати це як одну з ознак аномальної групової активності. При цьому сама наявність активного поширення не є достатньою підставою для класифікації контенту як маніпулятивного – поведінкові характеристики використовуються лише у поєднанні з результатами семантичного аналізу та іншими параметрами системи.

Поведінкові характеристики поширення повідомлення агрегуються у вектор G, загальний вигляд якого було наведено у підрозділі 2.1. Для уточнення структури поведінкового вектора та деталізації параметрів, що використовуються під час аналізу поширення повідомлення, процес його формування можна подати у вигляді:

$$G = f(I, A, R, C, T) \tag{2.5}$$

де I – інтенсивність поширення повідомлення;

A – рівень активності акаунтів;

R – швидкість появи реакцій;

C – характеристики взаємодій між користувачами;

T – часові параметри поширення повідомлення.

Після формування поведінкових характеристик окремі параметри також приводяться до уніфікованого вигляду для подальшого використання у модулі оцінювання ризику. Що знову ж таки дозволяє поєднати результати семантичного та поведінкового аналізу у межах єдиної інтегрованої системи оцінювання інформаційних загроз.

Таким чином, аналіз поведінкових характеристик поширення дозволяє враховувати не лише зміст повідомлення, але й особливості його розповсюдження у соціальній мережі. Використання поведінкових сигналів у поєднанні із семантичним аналізом дозволяє підвищити надійність виявлення потенційно маніпулятивного контенту та зменшити вплив обмежень використання окремих методів аналізу.

Після формування базових поведінкових характеристик окрему увагу слід приділити аналізу нетипової активності акаунтів та ознакам автоматизованого поширення інформації. У межах соціальних мереж маніпулятивні кампанії часто супроводжуються використанням великої кількості профілів із подібними сценаріями поведінки, синхронною активністю або неприродно високою інтенсивністю взаємодій [42]. Саме тому додатково враховуються характеристики, що можуть вказувати на ботоподібну або аномальну поведінку. Достовірне визначення автоматизованих акаунтів є складною задачею, оскільки частина даних соціальнних мереж може бути недоступною через технічні обмеження платформи або інші налаштування приватності. Тому аналізуються не окремі акаунти як такі, а поведінкові сигнали, що можуть свідчити про нетипові шаблони активності.

При проведенні аналізу враховуються часові особливості взаємодій, повторюваність поведінкових сценаріїв акаунтів, інтенсивність публікацій та

характер реакцій на повідомлення. Наприклад – занадто висока швидкість поширення контенту, регулярні повторювані сценарії активності або синхронна взаємодія груп акаунтів можуть розглядатися як додаткові фактори ризику. Подібні характеристики самі по собі не є достатньою підставою для класифікації активності як маніпулятивної, однак у поєднанні з іншими параметрами дозволяють підвищити достовірність оцінювання.

Окремо враховується стабільність поведінки акаунтів у часі –для звичайного користувача активність характерезується нерівномірністю взаємодій та зміною інтенсивності використання соціальної мережі, натомість – автоматизовані або напівавтоматизовані сценарії поширення інформації можуть супроводжуватись надмірною регулярністю взаємодій, повторюваними часовими інтервалами активності або узгодженими шаблонами поведінки акаунтів.

Характеристики ботоподібної активності використовуються, як допоміжний компонент аналізу та не розглядаються ізольовано від інших параметрів – це дозволяє зменшити кількість помилкових спрацювань, оскільки окремі ознаки нетипової поведінки можуть бути характерними і для звичайних користувачів, особливо у випадках активного обговорення популярних інформаційних подій.

Для узагальнення характеристик аномальної активності формується набір додаткових поведінкових параметрів, які надалі використовуються у модулі оцінювання ризику. Процес формування набору характеристик ботоподібної активності можна подати у вигляді:

$$B = f(F, S, P, R) \tag{2.6}$$

де F – частота та інтенсивність взаємодій;

S – синхронність активності акаунтів;

P – повторюваність поведінкових шаблонів;

R – регулярність часових інтервалів активності.

Сформовані параметри використовуються як додаткові поведінкові

сигнали під час комплексного оцінювання рівня ризику повідомлення – це дозволяє враховувати не лише зміст інформації, а й особливості активності акаунтів, що беруть участь у її поширенні. Таким чином, аналіз ботоподібної та аномальної активності дозволяє доповнити результати семантичного аналізу та підвищити надійність виявлення потенційно маніпулятивного контенту в умовах неповноти або обмеженості вхідних даних.

Характеристики поширення повідомлення та ознаки ботоподібної активності аналізуються як взаємопов’язані поведінкові компоненти системи, що дозволяє враховувати не лише загальну динаміку поширення інформації, а й особливості активності акаунтів, що беруть участь у взаємодіях із постом.

Вектор G, структура якого була наведена у формулі (2.5), використовується для опису параметрів поширення повідомлення: інтенсивності взаємодій, швидкості появи реакцій, активності аудиторії та часових характеристик інформаційного потоку. Окремо сформований набір параметрів B, поданий у формулі (2.6), який характеризує ознаки нетипової або ботоподібної поведінки акаунтів.

Розділення цих груп характеристик дозволяє уникнути ситуації, коли активне поширення повідомлення автоматично інтерпретується, як ознака маніпулятивної активності. Наприклад – висока інтенсивність взаємодій може бути характерною як для нормального поширення популярного контенту, так і для координованого інформаційного впливу, тому характеристики поширення повідомлення та ознаки аномальної активності аналізуються у взаємозв’язку між собою.

Розділення поведінкових характеристик на окремі групи також підвищує стійкість системи до неповноти вхідних даних. На практиці частина інформації про взаємодії або активність акаунтів може бути недоступною через різні обмеження соціальних мереж або приватності користувачів. У такому випадку оцінювання ризику не повинно залежати лише від одного типу поведінкових ознак.

У результаті параметри поширення повідомлення та ознаки ботоподібної

активності використовуються спільно під час подальшого інтегрованого оцінювання рівня потенційної інформаційної загрози.

2.4. Алгоритм оцінювання ризику повідомлення

Після отримання результатів семантичного та поведінкового аналізу система може переходити до комплексного оцінювання рівня ризику повідомлення. На цьому етапі система переходить від окремого аналізу ознак до їх спільної інтерпретації в межах єдиної оцінки ризику.

У підрозділі 2.1 інтегральну оцінку повідомлення було подано у загальному вигляді як функцію об'єднання семантичних та поведінкових характеристик. Після деталізації окремих компонентів моделі структуру оцінювання можна уточнити таким чином:

$$T(m) = F(S, G, B, Q) \tag{2.7}$$

де S – семантичні характеристики повідомлення;

G – параметри поширення повідомлення;

B – характеристики ботоподібної або аномальної активності акаунтів;

Q – показник повноти та достовірності отриманих даних.

Подібна структура моделі відповідає логіці роботи, у якій результати окремих модулів аналізу поєднуються під час формування фінальної оцінки ризику [43]. Семантичний компонент використовується для аналізу змісту повідомлення, поведінковий – для оцінювання особливостей поширення інформації, а параметри ботоподібної активності враховують нетипові шаблони взаємодії акаунтів [40].

Додатково враховується показник Q, який характеризує рівень повноти доступних даних. Його використання дозволяє коригувати підсумкову оцінку ризику залежно від обсягу та надійності характеристик, отриманих під час аналізу повідомлення.

Формування інтегральної оцінки виконується на основі узгодженого

аналізу декількох груп характеристик повідомлення – під час оцінювання система враховує результати семантичного аналізу, параметри поширення інформації, ознаки аномальної активності акаунтів та рівень повноти доступних даних. Подібний підхід дозволяє формувати підсумкову оцінку ризику з урахуванням взаємозв'язків між окремими характеристиками повідомлення та особливостями його поширення.

Інтегрована модель оцінювання забезпечує узгодження результатів різних модулів аналізу та формування єдиного показника ризику. Завдяки цьому підсумкове рішення системи базується не на окремій ознаці, а на сукупності семантичних, поведінкових та ботоподібних характеристик повідомлення.

Для оцінювання якості навченої моделі класифікації використовуються метрики справедливості (Fairness Metrics), що дозволяють перевірити рівномірність розподілу прогнозів між класами. Зокрема, статистичний паритет вимірює, чи рівномірно розподілені передбачення між групами, і в результаті фінальна оцінка ризику формується на основі сукупного аналізу змісту повідомлення, характеристик його поширення та особливостей поведінки акаунтів, що взаємодіють із контентом [44].

Для практичного використання інтегрованої моделі необхідно визначити порядок формування числового показника ризику на основі результатів роботи окремих модулів аналізу. Результати семантичного, лінгвістичного та поведінкового аналізу приводяться до узгодженого формату та використовуються під час обчислення підсумкової оцінки повідомлення.

У загальному вигляді проміжний показник ризику можна подати так:

$$R_{raw} = w_1 P_{nlp} + w_2 P_{ling} + w_3 P_{net} \quad (2.8)$$

де P_{nlp} – оцінка, отримана за допомогою NLP-модуля;

P_{ling} – лінгвістична оцінка маніпулятивних ознак тексту;

P_{net} – поведінковий сигнал поширення;

w_1, w_2, w_3 – вагові коефіцієнти відповідних компонентів.

Така схема дає змогу враховувати декілька груп ознак під час формування

фінальної оцінки. NLP-компонент відображає контекстне представлення повідомлення, лінгвістичний компонент фіксує характерні маніпулятивні конструкції, а поведінковий сигнал враховує особливості поширення інформації та активності акаунтів.

У випадках, коли NLP-модуль недоступний або частина даних відсутня, система може використовувати спрощений режим оцінювання, що базується на доступних лінгвістичних і поведінкових параметрах:

$$R_{raw} = F(P_{ling}, P_{net}) \tag{2.9}$$

Наявність такого режиму забезпечує працездатність системи за умов неповного набору вхідних характеристик або обмежень середовища виконання.

Після обчислення проміжного значення результат приводиться до уніфікованого діапазону:

$$0 \leq R_{raw} \leq 1 \tag{2.10}$$

Це забезпечує єдиний формат представлення ризику та спрощує подальшу інтерпретацію результатів у межах застосунку.

Оскільки ознаки ботоподібної активності у підрозділі 2.3 були представлені як окрема група параметрів B, у межах оцінювання вони можуть бути зведені до окремого числового показника P_{bot}:

$$P_{bot} = \sigma(M, P) \tag{2.11}$$

де M – оцінка аномальності поведінки акаунта;

P – профільна оцінка акаунта;

σ – функція нелінійного перетворення, що приводить результат до інтерпретованого ймовірнісного вигляду.

Оцінка ботоподібної активності не замінює загальний показник ризику повідомлення, а використовується як додатковий сигнал під час комплексного аналізу. Це дає змогу враховувати ситуації, коли підозрілою є не лише семантика повідомлення, а й характер активності акаунтів, що беруть участь у його

поширенні [40].

Отже, формалізація алгоритму оцінювання ризику визначає порядок переходу від часткових результатів аналізу до єдиного числового показника. Такий показник надалі може використовуватись для класифікації повідомлення та подання результатів користувачу у застосунку.

Побудова алгоритму виявлення маніпулятивного контенту базується на послідовній обробці вхідного повідомлення та поступовому уточненні рівня ризику. Основна задача алгоритму полягає не лише в обчисленні числового показника, а й у визначенні того, які саме групи ознак доступні для аналізу та можуть бути використані під час оцінювання.

На початковому етапі система отримує текст повідомлення, доступні метадані та інформацію про взаємодії з контентом. Далі виконується перевірка повноти вхідних даних: якщо доступний лише текст, основна увага приділяється семантичному та лінгвістичному аналізу; якщо також наявні дані про поширення, до оцінювання додаються поведінкові характеристики; за наявності профільних або часових параметрів акаунтів враховуються ознаки ботоподібної активності.

Після цього система формує часткові оцінки для доступних груп ознак. Для текстового вмісту визначаються семантичні та лінгвістичні показники, для поширення – поведінковий сигнал, для акаунтів – оцінка аномальної або ботоподібної активності. Якщо певний компонент недоступний, алгоритм не припиняє роботу, а переходить до спрощеного режиму оцінювання на основі наявних параметрів.

На наступному етапі виконується об’єднання часткових результатів у проміжний показник ризику. Після цього значення приводиться до єдиного діапазону та передається до блоку інтерпретації. Важливо, що алгоритм не формує висновок лише за однією ознакою: підвищений рівень ризику визначається на основі поєднання декількох факторів, а не через окремий показник.

В алгоритмі окремі модулі аналізу не функціонують ізольовано, а використовуються як взаємопов’язані компоненти єдиної системи оцінювання. Це дозволяє враховувати взаємозв’язок між змістом повідомлення, характеристиками його поширення та особливостями активності акаунтів під час формування підсумкового результату.

Загальну послідовність роботи алгоритму можна подати так:

1. отримання тексту повідомлення та доступних метаданих;
2. перевірка наявності текстових, поведінкових і профільних характеристик;
3. формування часткових оцінок для доступних груп ознак;
4. вибір повного або спрощеного режиму оцінювання;
5. обчислення проміжного показника ризику;
6. нормалізація результату до єдиного діапазону;
7. передача оцінки до модуля інтерпретації та frontend-компонента.

У результаті алгоритм забезпечує гнучке оцінювання повідомлення залежно від доступності вхідних даних – така структура відповідає модульній архітектурі системи та дозволяє зберігати працездатність аналізу навіть за неповного набору характеристик. Загальну схему роботи алгоритму оцінювання ризику повідомлення наведено на рисунку 2.4.

Після обчислення інтегрального показника ризику проводиться інтерпретація результатів. Основна задача полягає у перетворенні числової оцінки на зрозуміле для користувача представлення рівня потенційної інформаційної загрози.

У межах застосунку підсумкова оцінка використовується не лише для класифікації повідомлення, а й для формування короткого пояснення щодо результатів аналізу. Під час інтерпретації враховується, які саме групи ознак мали найбільший вплив на фінальний результат: семантичні характеристики тексту, поведінкові параметри поширення або ознаки аномальної активності акаунтів.

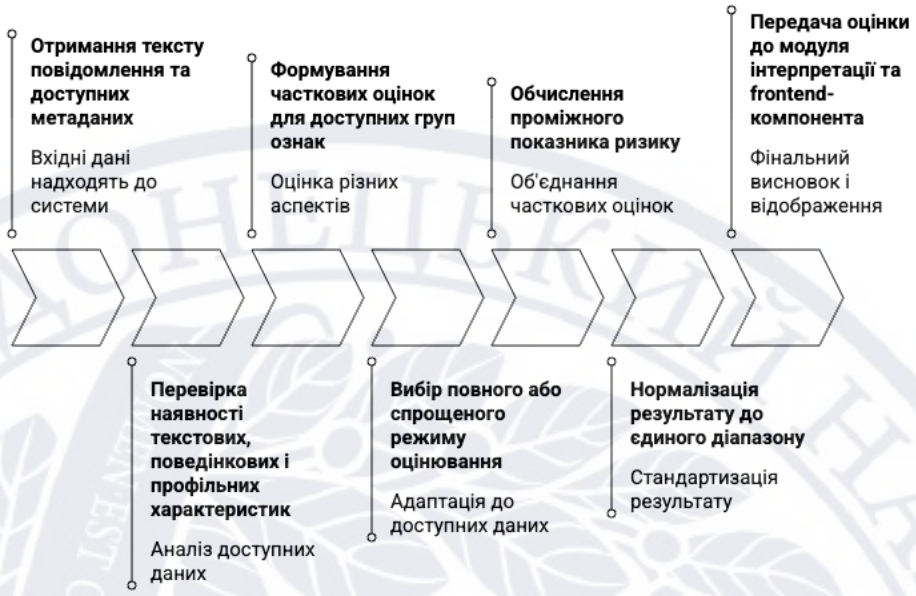

Рисунок 2.4 – Послідовність роботи алгоритму оцінювання ризику повідомлення

Для спрощення представлення результатів інтегральний показник ризику може бути поділений на декілька умовних рівнів:

- низький рівень ризику;
- середній рівень ризику;
- високий рівень ризику.

Низький рівень ризику відповідає відсутності виражених ознак маніпулятивного впливу або аномальної активності. Середній рівень може свідчити про наявність окремих підозрілих характеристик повідомлення чи його поширення. Високий рівень ризику визначається у випадках, коли система одночасно фіксує декілька груп потенційно небезпечних ознак.

Окремо під час інтерпретації результатів враховується повнота доступних вхідних даних. У випадку обмеженого набору характеристик система може додатково позначати результат аналізу, що дозволяє коректніше оцінювати рівень достовірності отриманої оцінки.

У фронтенд-компоненті результати аналізу відображаються у вигляді інтегральної оцінки ризику, короткого пояснення та додаткових характеристик

повідомлення. Це забезпечує зрозуміле подання результатів аналізу без необхідності безпосереднього перегляду внутрішніх параметрів моделі.

Оцінка функціонування запропонованої моделі визначається поєднанням кількох груп ознак, що описують повідомлення з різних сторін. На відміну від підходів, які враховують лише текстовий зміст, розроблена модель додатково використовує поведінкові характеристики поширення та ознаки аномальної активності акаунтів. Це дозволяє виявляти ризикові повідомлення навіть у випадках, коли маніпулятивний вплив не має очевидного мовного вираження.

Особливістю моделі є інтегроване оцінювання ризику, у якому результати окремих модулів не розглядаються ізольовано. Підсумкова оцінка формується на основі взаємозв’язку між змістом повідомлення, динамікою його поширення та характером активності акаунтів. Завдяки цьому зменшується ймовірність помилкової інтерпретації окремих ознак, наприклад, високої емоційності тексту або активного поширення популярного повідомлення. Окремі компоненти можуть працювати незалежно, а результати їх аналізу приводяться до спільного формату для подальшого оцінювання ризику. Така побудова спрощує розширення системи, дає змогу змінювати або доповнювати окремі модулі без повної перебудови застосунку та забезпечує роботу за умов неповного набору вхідних характеристик.

Отже, розроблена модель забезпечує комплексний підхід до виявлення маніпулятивного контенту в соціальних мережах. Вона поєднує аналіз змісту, поведінкових характеристик і ознак аномальної активності, формує інтегральний показник ризику та подає результат у зрозумілому для користувача вигляді. Це створює основу для практичної реалізації застосунку та подальшої перевірки його роботи на даних соціальних мереж.

РОЗДІЛ 3
РЕАЛІЗАЦІЯ ЗАСТОСУНКУ ВИЯВЛЕННЯ МАНІПУЛЯТИВНОГО КОНТЕНТУ

3.1. Архітектура та технології розробки застосунку

Основною мовою програмування для реалізації застосунку було обрано Python, що зумовлено широкими можливостями мови у сфері обробки природної мови, машинного навчання та аналізу даних. Python має значну кількість бібліотек для роботи з текстовою інформацією та інтеграції сучасних моделей штучного інтелекту, що робить його одним із найпоширеніших інструментів розробки інтелектуальних систем аналізу контенту.

Серед сучасних підходів до розробки серверної частини застосунків особливої популярності набуває асинхронна архітектура, що дозволяє розробникам зосередитися на написанні логіки, не обтяжуючись управлінням серверами. Використання асинхронних фреймворків значно спрощує процес масштабування та знижує витрати на інфраструктуру [28]. Тому для реалізації серверної частини застосунку використовується FastAPI. Даний фреймворк підтримує асинхронну обробку запитів, що буде корисно у разі якщо додаток перейде на багатокористувацьку або багатозадачну системи, автоматичну генерацію документації API та забезпечує високу продуктивність взастосунку. Це дозволяє реалізувати централізовану взаємодію між інтерфейсом користувача та модулями аналізу контенту.

Для формалізації структури даних використовується бібліотека Pydantic. Вона забезпечує валідацію інформації, автоматичне перетворення типів даних та підтримання єдиної моделі обміну даними між компонентами системи.

Особливу роль у системі відіграють засоби аналізу природної мови. Для семантичного аналізу тексту використовуються попередньо навчені трансформерні моделі та алгоритми обробки текстових даних. Це дозволяє оцінювати зміст повідомлень не лише на рівні окремих слів, а й з урахуванням контексту, емоційного забарвлення та характерних мовних конструкцій.

При виборі системи зберігання даних для застосунку враховувалися переваги реляційних СУБД, вони демонструють стабільну продуктивність у сценаріях зі структурованими даними та складними взаємозв’язками, що вимагають гарантій цілісності [29]. Тому для зберігання результатів аналізу використовується SQLite завдяки простоті розгортання, відсутності потреби у виділеному сервері БД та зручності використання на етапах розробки й тестування системи. База даних забезпечує накопичення історії перевірок, збереження результатів оцінювання ризику та повторне використання отриманих результатів. У разі переходу до багатокористувацького режиму експлуатації або необхідності обробки великої кількості одночасних запитів доцільним є використання повноцінної серверної СУБД, зокрема PostgreSQL, яка забезпечує підтримку паралельних транзакцій, масштабування та підвищену продуктивність.

Користувацький інтерфейс реалізовано за допомогою React та Vite. Компонентна архітектура React дозволяє створити гнучкий інтерфейс відображення результатів аналізу, а Vite забезпечує швидке складання та запуск клієнтської частини застосунку.

Для забезпеченя стабільності розгортання використовується Docker. Контейнеризація дозволяє запускати систему в однаковому програмному середовищі незалежно від апаратної платформи та операційної системи.

Таким чином, обраний стек технологій орієнтований насамперед на реалізацію інтелектуального аналізу текстового контенту, оцінювання ризику маніпулятивного впливу та створення застосунку для взаємодії користувача з результатами аналізу.

Додаток для виявлення маніпулятивного контенту реалізовано за допомогою клієнт-серверної архітектури. Клієнтська частина забезпечує введення посилання, запуск аналізу та відображення результатів, а серверна частина відповідає за обробку запиту, формування аналітичних показників і передачу структурованої відповіді до інтерфейсу користувача. Загальну структуру взаємодії між компонентами застосунку наведено на рисунку 3.1.

Рисунок 3.1 – Архітектура застосунку виявлення маніпулятивного контенту

Frontend-частина реалізована за допомогою React та Vite. Основний файл застосунку App.jsx містить набір вкладок, через які користувач може переглядати різні результати аналізу: загальний огляд, граф каскаду, поширення, маніпулятивні ознаки, підозрілі акаунти, хмару слів, звіт та історію. Фрагмент структури вкладок наведено у лістингу:

```
const TABS = [
  { id: "overview",     label: "Огляд" },
  { id: "graph",        label: "Граф каскаду" },
  { id: "ic",           label: "Поширення" },
  { id: "manipulation", label: "Маніпуляції" },
  { id: "bots",         label: "Підозрілі акаунти" },
  { id: "wordcloud",    label: "Хмара слів" },
  { id: "report",       label: "Звіт" },
  { id: "history",      label: "Історія" },
]
```

Для запуску аналізу користувач вводить посилання на публікацію, після чого frontend формує POST-запит до REST API backend-сервера. Відповідний

фрагмент реалізації наведено у лістингу формування запиту на аналіз публікації:

```
const response = await fetch("/api/v1/analyze", {
  method: "POST",
  headers: { "Content-Type": "application/json" },
  body: JSON.stringify({ url: inputUrl, depth: 2, max_nodes: 300 }),
    })
```

Backend-частина реалізована на основі FastAPI. Основні маршрути зосереджені у файлі `backend/api/routes.py`. Для запуску аналізу використовується `endpoint /api/v1/analyze`, який приймає URL, перевіряє його коректність і формує результат аналізу. Фрагмент реалізації endpoint наведено у лістингу реалізації endpoint запуску аналізу:

```
@router.post("/analyze", response_model=AnalysisResult)
async def analyze_post(req: AnalyzeRequest):
    url = req.url.strip()
    if not url:
        raise HTTPException(422, "Посилання не може бути порожнім")

    result = build_analysis(url)
    _save_to_history(result)
    return result
```

З фрагменту видно, що backend приймає запит від frontend-частини, запускає внутрішній pipeline аналізу повідомлення та повертає результат у форматі AnalysisResult. У межах pipeline формуються оцінка маніпулятивності, результати аналізу ботоподібної активності, метрики каскаду, IC-модель поширення, оцінка якості даних та хмара ключових слів.

Окремо реалізовано endpoint для формування хмари ключових слів. Він приймає текст повідомлення та повертає список термінів із вагами, як показано у лістингу Endpoint формування хмари ключових слів:

```
@router.post("/word-cloud", response_model=List[WordCloudTerm])
```

```
async def word_cloud(req: WordCloudRequest):
    if not req.text.strip():
        return []
    return make_word_cloud(req.text, limit=max(8, min(req.limit,
80)))
```

Для передачі результатів між backend і frontend використовуються Pydantic-моделі даних. Основною моделлю є AnalysisResult, яка об'єднує всі дані, необхідні для відображення результатів у інтерфейсі. Її структура наведена у лістингу результатів аналізу:

```
class AnalysisResult(BaseModel):
    url: str
    post: PostData
    manipulation: ManipulationResult
    bots: List[BotScore]
    cascade: CascadeMetrics
    ic_simulation: ICSimulation
    data_quality: DataQuality
    word_cloud: List[WordCloudTerm] = []
    scenario_id: str = ""
     analyzed_at: datetime = Field(default_factory=datetime.utcnow)
```

Модель `AnalysisResult` дозволяє передати до frontend-частини не окремі непов'язані значення, а повний структурований результат аналізу. Завдяки цьому компоненти `DashboardOverview`, `ManipulationPanel`, `BotPanel`, `CascadeGraph`, `ICSimulationPanel`, `WordCloudPanel`, `ReportPanel` та `HistoryPanel` отримують дані в єдиному форматі.

Для розгортання використовується Docker Compose. У його конфігурації передбачено окремі сервіси для backend та frontend частин, що показано у лістингу фрагменту Docker Compose конфігурації:

```
services:
  backend:
```

```
    build: ./backend
  frontend:
    build: ./frontend
```

Контейнеризація спрощує запуск застосунку, ізоляцію залежностей середовища виконання та дозволяє окремо підтримувати серверну й клієнтську частини.

Таким чином, архітектура застосунку побудована навколо FastAPI backend, React/Vite frontend та Pydantic-моделей даних. Backend формує повний результат аналізу, а frontend відображає його у вигляді окремих аналітичних панелей, що забезпечує зрозуміле представлення результатів для користувача.

3.2. Реалізація модулів аналізу маніпулятивного контенту

Коли backend-система отримує повідомлення – вона виконує формування набору характеристик, які використовуються для оцінювання потенційної маніпулятивності контенту. На відміну від спрощених підходів, де результат формується лише у вигляді одного числового значення, аналіз поділяється на декілька окремих компонентів: лінгвістичні сигнали, NLP-оцінки, текстові характеристики повідомлення.

Основні структури результату аналізу описані у файлі backend/core/models.py. Для представлення окремих сигналів маніпулятивності використовується модель ManipulationSignals Окремі сигнали формуються на основі аналізу текстових конструкцій повідомлення. Наприклад, сигнал emotional_trigger враховує наявність емоційно забарвлених фраз, підсилювальних слів, надмірного використання знаків оклику та інших конструкцій, що можуть вказувати на емоційний тиск або спробу привернення уваги користувача. Після обчислення проміжних характеристик отримані значення нормалізуються та приводяться до узгодженого діапазону, фрагмент якої наведено у лістингу структури сигналів маніпулятивності:

```
class ManipulationSignals(BaseModel):
    emotional_trigger: float = Field(default=0.0, ge=0, le=1)
```

```
    false_urgency: float = Field(default=0.0, ge=0, le=1)
    echo_chamber: float = Field(default=0.0, ge=0, le=1)
    repetition_pattern: float = Field(default=0.0, ge=0, le=1)
    bot_amplification: float = Field(default=0.0, ge=0, le=1)
    coordinated_action: float = Field(default=0.0, ge=0, le=1)
    rhetoric_manipulation: float = Field(default=0.0, ge=0, le=1)
    caps_abuse: float = Field(default=0.0, ge=0, le=1)
```

У лістингу структури сигналів маніпулятивності показано, що кожна характеристика подається не як булеве значення, а як числовий параметр у діапазоні від 0 до 1. Для приведення різних ознак до єдиного масштабу використовується обмеження значення у допустимому інтервалі. Фрагмент відповідної реалізації наведено у лістингу обмежень значення сигналу в діапазоні [0;1]:

```
def clamp_score(value: float) -> float:
    return max(0.0, min(value, 1.0))
```

Такий підхід дозволяє уникати ситуацій, коли окрема характеристика має непропорційно великий вплив на підсумкову оцінку ризику.

Після нормалізації сигналів система формує узагальнений результат аналізу. Для цього використовується модель ManipulationResult, фрагмент якої наведено у лістингу структури результату аналізу маніпулятивності:

```
class ManipulationResult(BaseModel):
    overall_score: float = Field(ge=0, le=1)
    label: str
    signals: ManipulationSignals
    nlp_ensemble_score: float = Field(default=0.0, ge=0, le=1)
    linguistic_score: float = Field(default=0.0, ge=0, le=1)
    network_score: float = Field(default=0.0, ge=0, le=1)
    bayesian_calibrated: float = Field(ge=0, le=1)
    models_loaded: List[str] = []
```

Структура ManipulationResult об’єднує результати декількох компонентів аналізу: NLP-оцінки повідомлення, лінгвістичні сигнали та поведінкові характеристики поширення. Окремо зберігається скоригована оцінка bayesian_calibrated, що використовується для додаткового калібрування

результату.

Окремо у системі формується лінгвістична оцінка повідомлення. Вона базується на групі текстових сигналів, які описують емоційний тиск, штучну терміновість, риторичні маніпуляції, повторюваність конструкцій та надмірне використання великих літер. Для цього окремі сигнали агрегуються у спільний показник linguistic_score. У лістингу формування лінгвістичної оцінки повідомлення показано, що лінгвістична оцінка формується як зважена сума декількох текстових сигналів:

```
linguistic_score = clamp_score(
    0.25 * signals.emotional_trigger +
    0.20 * signals.false_urgency +
    0.20 * signals.rhetoric_manipulation +
    0.15 * signals.repetition_pattern +
    0.10 * signals.echo_chamber +
    0.10 * signals.caps_abuse
)
```

Найбільшу вагу мають ознаки емоційного впливу, штучної терміновості та риторичної маніпуляції, оскільки саме вони найбільш характерні для маніпулятивного контенту. Після формування лінгвістичної оцінки система виконує комбінування результатів окремих компонентів аналізу. У межах backend-реалізації для цього використовується зважене агрегування NLP-оцінки, лінгвістичних характеристик та поведінкового сигналу поширення. Фрагмент реалізації наведено у лістингу комбінування результатів аналізу:

```
raw_score = (
    0.45 * nlp_score +
    0.35 * linguistic_score +
    0.20 * network_score
     )
```

У наведеному фрагменті найбільший вплив має NLP-компонент, оскільки саме він формує контекстне представлення повідомлення. Лінгвістичні характеристики використовуються для уточнення маніпулятивних мовних конструкцій, а поведінковий сигнал враховує особливості взаємодії користувачів

із контентом.

Після формування сирої оцінки система додатково виконує її калібрування. Необхідність цього етапу пояснюється тим, що окремі модулі можуть формувати занадто впевнені результати навіть за неповного набору характеристик. Для зменшення ефекту overconfidence використовується додаткове калібрування оцінки ризику. Фрагмент реалізації наведено у лістингу калібрування оцінки ризику:

```
calibrated = calibrator.calibrate(raw_score)
final_score = max(0.0, min(calibrated, 1.0))
```

Після калібрування результат додатково приводиться до діапазону від 0 до 1, що дозволяє використовувати єдиний формат представлення оцінки ризику у frontend-компонентах системи.

Після завершення аналізу система додатково виконує підготовку текстових даних для побудови хмари слів. На відміну від використання готових бібліотечних прикладів, у межах застосунку попередньо реалізовано очищення тексту та фільтрацію службових елементів. Фрагмент реалізації наведено у лістингу попередньої обробки тексту для побудови хмари слів:

```
tokens = re.findall(
    r"[#@]?[A-Za-zA-Яа-яІіЇїЄєҐґ0-9_]{3,}",
    text.lower()
)
normalized = []
for token in tokens:
    token = token.strip("_.,:;!?()[]{}\"'`")
    if not token:
        continue
    if token in STOPWORDS:
        continue
    if token.startswith("@"):
        continue
    normalized.append(token)
```

У наведеному фрагменті система виконує:

- токенізацію тексту;

- приведення символів до нижнього регістру;
- очищення службових символів;
- фільтрацію stop-слів;
- видалення згадок акаунтів.

Така обробка дозволяє зменшити кількість шумових елементів та залишити лише важливі характеристики повідомлення. Після очищення тексту система виконує підрахунок частоти появи термінів та нормалізацію їх ваг. Фрагмент реалізації наведено у лістингу нормалізації ваг термінів:

```
counts = Counter(normalized)
max_count = max(counts.values())
for term, count in counts.most_common(limit):
    terms.append(
        WordCloudTerm(
            text=term,
            value=count,
            weight=round(count / max_count, 3),
        )
)
```

У лістингу показано, що кожен термін отримує абсолютну частоту появи та нормалізовану вагу. Нормалізація виконується відносно найбільш уживаного терміна, що дозволяє коректно масштабувати елементи під час побудови хмари слів у frontend-компоненті WordCloudPanel.

Таким чином, модулі аналізу маніпулятивного контенту реалізують не лише використання NLP-компонентів, а й власну логіку агрегування, нормалізації та калібрування результатів. Система поєднує семантичний аналіз тексту, поведінкові характеристики поширення, оцінювання ботоподібної активності та додаткову обробку текстових даних, що дозволяє формувати більш інформативний та інтерпретований результат аналізу повідомлення.

3.3. Реалізація модуля виявлення ботоподібної активності

Окремим компонентом backend-системи є модуль виявлення ботоподібної активності. Його задача полягає у визначенні нетипових шаблонів поведінки

користувачів, що можуть свідчити про автоматизоване або координоване поширення інформації. Реалізація модуля зосереджена у файлі backend/ml/bot_detector.py.

Основою аналізу є дослідження часових інтервалів між взаємодіями акаунта. Для цього система формує послідовність активності користувача та аналізує регулярність поведінки. У звичайних користувачів часові інтервали активності є нерівномірними, тоді як автоматизовані акаунти часто демонструють надто стабільні або повторювані шаблони взаємодій.

Для аналізу часових характеристик система обчислює інтервали між взаємодіями акаунта. Фрагмент реалізації наведено у лістингу аналізу часових інтервалів активності акаунта:

```
intervals = np.diff(activity_timestamps)
avg_interval = np.mean(intervals)
std_interval = np.std(intervals)
    regularity_score = 1.0 / (1.0 + std_interval)
```

У наведеному фрагменті система визначає:

- середній часовий інтервал активності;
- відхилення між інтервалами;
- рівень регулярності поведінки акаунта.

Якщо часові проміжки між взаємодіями майже не змінюються, значення regularity_score збільшується, що може свідчити про автоматизований характер активності. Для аналізу зміни поведінкових станів використовується Markov-модель. Її задача полягає у відстеженні переходів між різними режимами активності акаунта залежно від часових інтервалів між взаємодіями. Фрагмент реалізації наведено у лістингу формування поведінкових переходів акаунта:

```
states = []
for delta in intervals:
    if delta < 60:
        states.append("BURST")
    elif delta < 3600:
        states.append("ACTIVE")
```

```
elif delta < 86400:
    states.append("IDLE")
else:
        states.append("SILENT")
```

У межах моделі:

- BURST відповідає різким серіям активності;
- ACTIVE – регулярній активності;
- IDLE – тимчасовій неактивності;
- SILENT – тривалим періодам відсутності взаємодій.

Після формування послідовності станів система будує матрицю переходів між ними, яка показує частоту зміни режимів активності акаунта. Наприклад, визначається, як часто після стану ACTIVE виникає BURST або IDLE. Для оцінювання поведінки акаунта отримана матриця порівнюється з двома базовими шаблонами:

- шаблоном звичайної поведінки користувача;
- шаблоном ботоподібної активності.

У межах реалізації такі шаблони задаються у вигляді еталонних матриць переходів між станами. Для оцінювання відмінності між матрицями використовується KL-дивергенція:

$$D_{(KL)}(P \parallel Q) = \sum_{i,j} P(i,j) \log \frac{P(i,j)}{Q(i,j)} \quad 3.1$$

де $P(i,j)$ – імовірності переходів між станами для аналізованого акаунта, $Q(i,j)$ – еталонна матриця поведінки.

Якщо поведінка акаунта є ближчою до ботоподібного шаблону, значення поведінкової оцінки збільшується. Фрагмент реалізації наведено у лістингу порівняння поведінкових шаблонів:

```
kl_human = kl_divergence(user_matrix, HUMAN_T)
kl_bot = kl_divergence(user_matrix, BOT_T)
markov_score = kl_human / (kl_human + kl_bot)
```

У наведеному фрагменті значення markov_score збільшується, якщо поведінка акаунта є ближчою до ботоподібного шаблону. Окремо враховуються характеристики профілю акаунта: співвідношення підписників, інтенсивність публікацій, вік акаунта та інші параметри. Після цього поведінкові та профільні характеристики агрегуються у спільний показник. Фрагмент реалізації наведено у лістингу формування оцінки ботоподібної активності:

```
raw_score = (
    0.70 * markov_score +
    0.30 * profile_score
)
```

Основний вплив має markov_score, оскільки саме він враховує часові та поведінкові особливості взаємодій акаунта. Профільна оцінка використовується як додатковий фактор уточнення результату. Для приведення результату до ймовірнісного вигляду використовується sigmoid-нормалізація. Фрагмент реалізації наведено у лістингу нормалізації ймовірності ботоподібної активності:

```
bot_probability = sigmoid(
    5 * (raw_score - 0.40)
)
```

Після обчислення значення система формує одну з трьох категорій: людина, підозрілий, бот.

Отже, модуль виявлення ботоподібної активності поєднує аналіз часових характеристик, Markov-модель поведінкових переходів та профільні параметри акаунтів. Це дозволяє оцінювати не лише окремі характеристики профілю, а й загальний характер поведінки користувачів під час поширення інформації.

3.4. Реалізація інтерфейсу користувача та візуалізація результатів

Frontend-частина застосунку реалізована за допомогою React та Vite у вигляді набору окремих аналітичних панелей. Інтерфейс дозволяє переглядати результати аналізу повідомлення, характеристики поширення, поведінкові сигнали акаунтів та інтегральну оцінку ризику. Основними вкладками системи

є: огляд, граф каскаду, поширення, маніпуляції, підозрілі акаунти, хмара слів та звіт. Загальний вигляд інтерфейсу наведено на рисунку 3.2.

Рисунок 3.2 – Головна панель застосунку

Для відображення результатів аналізу маніпулятивності використовується компонент ManipulationPanel. У ньому відображаються сигнали ризику, текстові характеристики повідомлення та інтегральна оцінка маніпулятивності. Окремо система показує пояснення щодо факторів, які найбільше вплинули на фінальний результат аналізу. Приклад відображення панелі наведено на рисунку 3.3.

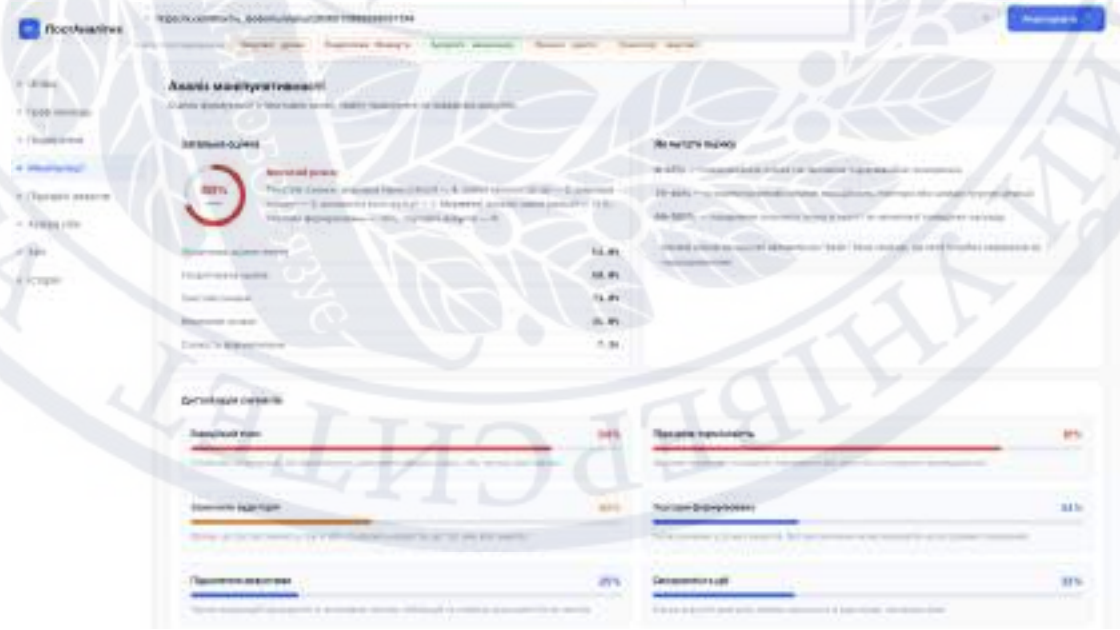

Рисунок 3.3 – Панель аналізу маніпулятивності

Для аналізу поведінки акаунтів реалізовано компонент BotPanel, який відображає ймовірність ботоподібної активності, anomaly score та поведінкові характеристики акаунтів. Приклад роботи панелі наведено на рисунку 3.4.

Для візуалізації ключових термінів повідомлення використовується компонент WordCloudPanel. Розмір слова залежить від частоти його появи у каскаді. Фрагмент реалізації наведено у лістингу відображення хмари слів:

```
{terms.map(term => (
  <span
    key={term.text}
    style={{
      fontSize: `${16 + term.weight * 42}px`
    }}
  >
    {term.text}
  </span>
))}
```

На рисунку 3.4 наведено панель аналізу підозрілих акаунтів.

Рисунок 3.4 – Панель аналізу підозрілих акаунтів

Приклад побудови хмари слів наведено на рисунку 3.5.

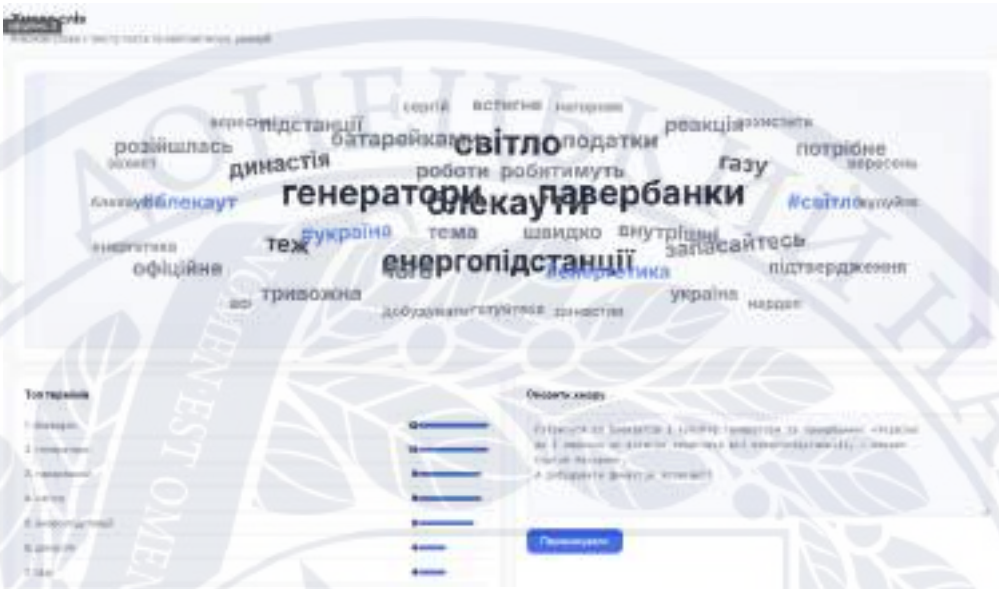

Рисунок 3.5 – Візуалізація хмари ключових слів

Таким чином забезпечується візуалізація всіх результатів аналізу у вигляді окремих інформаційних панелей, що спрощує інтерпретацію результатів роботи застосунку [45].

Після реалізації основних компонентів системи було виконано перевірку роботи frontend частини застосунку. Основна увага приділялась коректності взаємодії між компонентами, стабільності відображення результатів аналізу та зручності роботи користувача з інтерфейсом системи.

Окремо перевірялась адаптивність інтерфейсу. Frontend-застосунок коректно відображає інформаційні панелі на різних розмірах екрана, а структура компонентів автоматично перебудовується залежно від області відображення. Це дозволяє зберігати зручність роботи як на великих моніторах, так і на ноутбуках або мобільних пристроях. Приклад адаптивного відображення інтерфейсу наведено на рисунку 3.7.

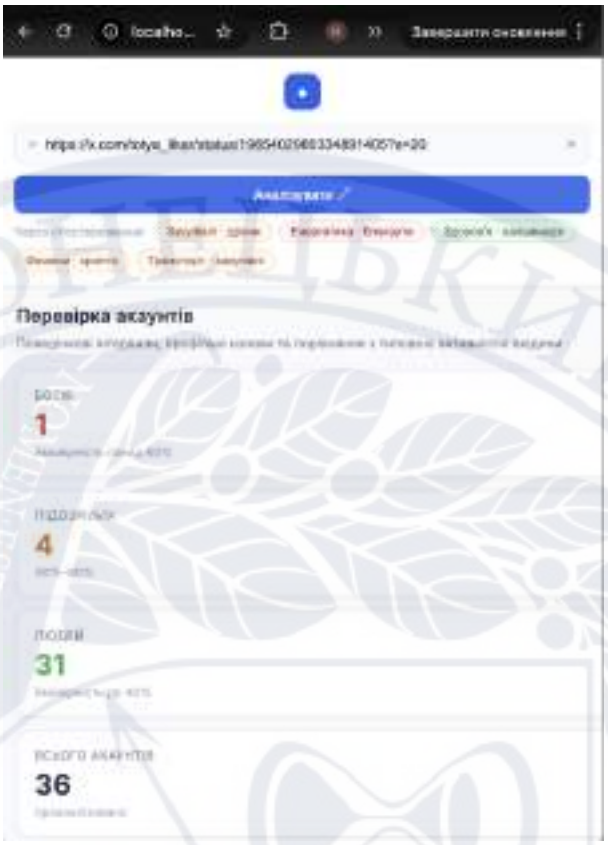

Рисунок 3.7 – Адаптивне відображення інтерфейсу застосунку

Також при зменшені екрану бокове меню, для зручності відкривається при натиску на логотип застосунку, що наведено на рисунку 3.8.

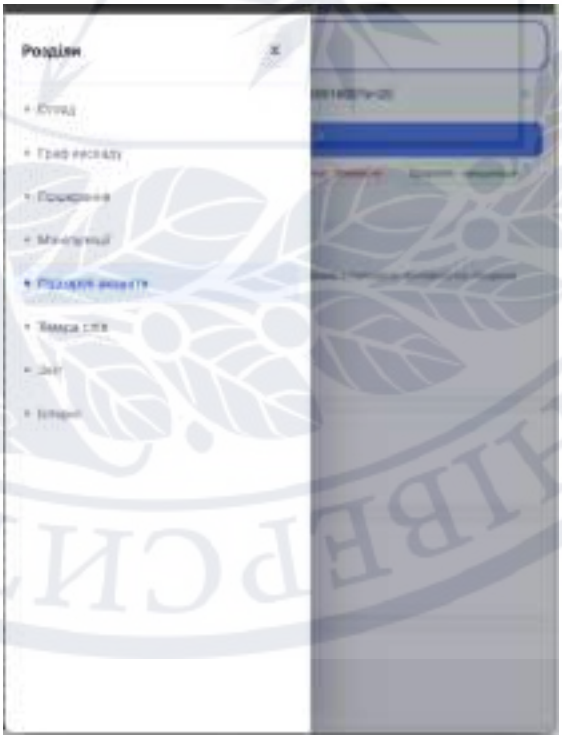

Рисунок 3.7 – Відображення блоку меню на малих екранах

Також оцінювалась організація інформаційних блоків інтерфейсу. Результати аналізу згруповані у вигляді окремих панелей, що дозволяє розділити характеристики – подібна структура спрощує сприйняття результатів та зменшує перевантаження інтерфейсу великою кількістю інформації. Також перевірялась обробка не правильних посилань. Перед запуском система виконує перевірку коректності URL-адреси та підтримуваної платформи. Приклад обробки невалідного запиту наведено на рисунку 3.9.

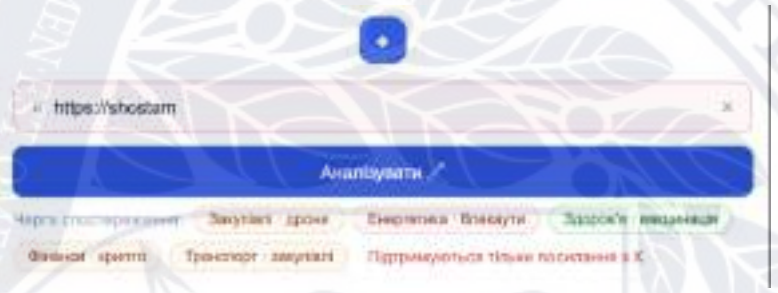

Рисунок 3.9 – Обробка некоректного посилання у застосунку

Перевірка роботи системи показала коректність роботи frontend компонентів, стабільність відображення результатів аналізу та працездатність основних елементів користувацького інтерфейсу.

ВИСНОВКИ

Бакалаврська робота було була присвячена вирішенню питань проєктування та реалізації застосунку для виявлення маніпулятивного контенту в соціальних мережах. В роботі проведено аналіз теоретичних основ інформаційних загроз, що несе маніпулятивний контент, обґрунтовано вибір методів аналізу тексту. Реалізовано застосунок у вигляді програмної системи, яка дозволяє оцінювати потенційні ознаки маніпулятивного впливу в публікаціях соціальних мереж. За результатами роботи були отримані наступні результати та висновки:

1. У процесі виконання роботи було досліджено теоретичні аспекти маніпулятивного впливу в соціальних мережах. Проаналізовано сучасні підходи до виявлення маніпулятивного контенту, дезінформації та пропаганди. Встановлено, що існуючі рішення мають низку обмежень: частина з них орієнтована переважно на ручний фактчекінг, інші оцінюють достовірність джерела, але не аналізують конкретний текст повідомлення. Це підтвердило доцільність розробки окремого застосунку, орієнтованого саме на аналіз повідомлень соціальних мереж.

Розкрито теоретичні аспекти маніпулятивного впливу в цифровому середовищі. У роботі розглянуто основні типи інформаційних загроз: дезінформацію, міс-інформацію, пропаганду, клікбейт, емоційну маніпуляцію, deepfake-контент, бот-активність та скоординовану неавтентичну поведінку. Окремо визначено лінгвістичні ознаки маніпулятивного контенту, зокрема емоційний, тиск, штучну терміновість риторичні маніпуляції, повторювані конструкції, надмірне використання великих літер та формування ехо-камер.

2. Обґрунтовано вибір методів аналізу тексту та підходів до візуалізації результатів. У роботі розглянуто методи попередньої обробки тексту, токенізацію, нормалізацію, формування семантичних ознак, використання NLP-моделей і лінгвістичних сигналів. Встановлено, що аналіз лише текстового змісту є недостатнім, оскільки маніпулятивний вплив може проявлятися також

через поведінкові та структурні особливості поширення повідомлення. Розроблено модель оцінювання ризику маніпулятивності, яка поєднує семантичні характеристики повідомлення, лінгвістичні сигнали, поведінкові параметри поширення та ознаки ботоподібної активності. Підсумкова оцінка формується на основі декількох груп ознак, що дозволяє уникнути залежності від одного показника та забезпечує більш обґрунтовану інтерпретацію результатів аналізу.

3. Спроєктовано архітектуру застосунку на основі клієнт-серверної моделі. Backend-частина реалізована з використанням FastAPI та відповідає за обробку запитів, запуск pipeline аналізу, формування результатів і передачу їх до frontend. Frontend-частина реалізована за допомогою React та Vite і забезпечує введення посилання, запуск аналізу, перегляд оцінки ризику, панелі маніпулятивних ознак, підозрілих акаунтів, хмари слів, звіту та історії.

4. Програмно реалізовано застосунок, який включає основні модулі: модуль аналізу маніпулятивності, модуль виявлення ботоподібної активності, модуль формування хмари ключових слів, модуль обробки результатів і frontend-компоненти для їх візуалізації. Для обміну даними між backend і frontend використано структуровані Pydantic-моделі.

5. Проведено перевірку роботи застосунку. Було перевірено коректність формування результату аналізу, роботу панелі маніпулятивності, панелі підозрілих акаунтів, хмари слів, адаптивність інтерфейсу та обробку некоректних посилань. Результати перевірки підтвердили працездатність основних компонентів та коректність відображення результатів користувачу. Розроблений застосунок може використовуватись як інструмент попереднього аналізу контенту соціальних мереж.

6. Подальший розвиток системи може бути пов’язаний із розширенням підтримки українськомовних і мультимовних NLP-моделей, додаванням нових типів маніпулятивних ознак, покращенням аналізу поведінкових сигналів, інтеграцією мультимедійного аналізу та переходом до більш масштабованої серверної архітектури для стабільної роботи з паралельними запитам.

СПИСОК ВИКОРИСТАНИХ ПОСИЛАНЬ

1. Джерела інформації українців у 2023 році. URL: https://uchoose.uacrisis.org/dzherela-informatsiyi-ukrayintsiv-u-2023-rotsi/ (дата звернення: 09.05.2026)

2. Dwi Surjatmodjo, Andi Alimuddin Unde, Hafied Cangara, Alem Febri Sonni. Information Pandemic: A Critical Review of Disinformation Spread on Social Media and Its Implications for State Resilience. 9 серпня 2024 р. URL: https://doi.org/10.3390/socsci13080418

3. Key Statistics on Fake News & Misinformation in Media in 2024. URL: https://redline.digital/fake-news-statistics/ (дата звернення: 09.05.2026)

4. ‘Think before sharing,’ Giorgia Meloni says as AI-made lingerie image of her goes viral. URL: https://www.theguardian.com/world/2026/may/05/giorgia-meloni-ai-generated-lingerie-image-deepfake (дата звернення: 11.05.2026)

5. Пропаганда та дезінформація: що найбільше впливає на український інфопростір. URL: https://nsju.org/novini/propaganda-ta-dezinformacziya-shho-najbilshe-vplyvaye-na-ukrayinskyj-infoprostir/ (дата звернення: 11.05.2026)

6. Лаптєва М. А., Баценко Т. В. Психолінгвістичні ознаки маніпулятивного контенту в соціальних мережах. Міжнародна наукова інтернет-конференція на тему: "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення", випуск 110 (м. Тернопіль, 7-8 травня 2026 року).

7. EEAS Report on Foreign Information Manipulation and Interference Threats. URL: https://www.eeas.europa.eu/eeas/2nd-eeas-report-foreign-information-manipulation-and-interference-threats_en (дата звернення: 11.05.2026)

8. Javier Pastor, Pantaleone Nespoli, José A. Ruipérez, David Camacho. Influence Operations in Social Networks. 17 лютого, 2025. URL: https://arxiv.org/html/2502.11827v1

9. Потапова Н. А., Суліма В. К., Лаптєва М. А., Павлюк О.А. Концептуальні засади комплексної оцінки інформаційних загроз у соціальних мережах: теоретико-математична модель. Наука і техніка сьогодні. 2026. №5. С.

5642-5653.

10. Дезінформація, пропаганда і все, що між ними: як розпізнати і захиститися. URL: https://detector.media/withoutsection/article/201754/2022-08-10-dezinformatsiya-propaganda-i-vse-shcho-mizh-nymy-yak-rozpiznaty-i-zakhystytysya/ (дата звернення: 12.05.2026)

11. Jurafsky D., Martin J. H. Speech and Language Processing. 3rd ed. Stanford University, 9 січня, 2026. URL: https://web.stanford.edu/~jurafsky/slp3/ed3book_jan26.pdf

12. Jurafsky D., Martin J. H. Speech and Language Processing. 3rd ed. Draft. Stanford University, 2024. URL: https://web.stanford.edu/~jurafsky/slp3/

13. Pymorphy3 Documentation. Morphological analyzer for Russian and Ukrainian languages. 2024. URL: https://pymorphy3.readthedocs.io/ (дата звернення: 12.05.2026)

14. Stanza – A Python NLP Package for Many Human Languages. URL: https://stanfordnlp.github.io/stanza/ (дата звернення: 15.05.2026)

15. Sudhakar M., Kaliyamurthie K.P. Detection of fake news from social media using support vector machine learning algorithms, 2024 p. URL: https://www.sciencedirect.com/science/article/pii/S2665917424000047?via%3Dihub

16. Ayankemi O. O., Oluwaferanmi I. R., Abdullai B. A. Fake News Detection System Using Logistic Regression, Decision Tree and Random Forest, 2024. URL: https://www.researchgate.net/publication/380693392_Fake_News_Detection_System_Using_Logistic_Regression_Decision_Tree_and_Random_Forest

17. Herun Wan, Minnan Luo, Zihan Ma, Guang Dai, Xiang Zhao – How Do Social Bots Participate in Misinformation Spread? A Comprehensive Dataset and Analysis, 2025. URL: https://aclanthology.org/2025.emnlp-main.1604.pdf

18. DSA Transparency Report - October 2024. URL: https://transparency.x.com/dsa-transparency-report.html (дата звернення: 18.05.2026)

19. X transparency report reveals harmful content data. URL: https://www.mmm-online.com/home/channel/x-transparency-report-reveals-harmful-

content-data/ (дата звернення: 18.05.2026)

20. Integrity Reports, Third Quarter 2025. URL: https://transparency.meta.com/reports/integrity-reports-q3-2025 (дата звернення: 18.05.2026)

21. YouTube's New Standards for Inauthentic Content and Creator Likeness. URL: https://influencermarketinghub.com/youtube-inauthentic-content/ (дата звернення: 18.05.2026)

22. Morgan Wack, Kayla Duskin. Political Fact-Checking Efforts are Constrained by Deficiencies in Coverage, Speed, and Reach, 2024. URL: https://www.researchgate.net/publication/387184370_Political_Fact-Checking_Efforts_are_Constrained_by_Deficiencies_in_Coverage_Speed_and_Reach (дата звернення: 18.05.2026)

23. Bonfire Leadership Solutions. Navigating Truth in a Post-Truth World: Top Fact-Checking Blogs for Combating Disinformation in 2025. URL: https://bonfireleadershipsolutions.com/ (дата звернення: 19.05.2026)

24. Broniatowski D. A. et al. Twitter and Facebook posts about COVID-19 are less likely to spread misinformation compared to other health topics. PLOS ONE. 2022. Vol. 17, No. 1. Article e0261768. DOI: 10.1371/journal.pone.0261768

25. Aslett K. et al. News credibility labels have limited average effects on news diet quality and fail to reduce misperceptions. Science Advances. 2022. Vol. 8, No. 18. Article eabl3844. DOI: 10.1126/sciadv.abl3844

26. Automateed. AI Content Detection Tools 2025: The Best Accuracy & Trends. URL: https://www.automateed.com/ai-content-detection-tools-2025 (дата звернення: 19.05.2026)

27. Потапова Н. А., Веселовська Н. Р., Лаптєва М. А., Суліма В. К. Проєкт та архітектура системи аналізу інформаційних потоків у соціальних мережах на основі інтеграції NLP та графових методів. Наука і техніка сьогодні. 2026. № 5. С. 5628-5641.

28. Лаптєва М. А., Зелінська О. В. Сучасні тенденції backend мобільних додатків. Збірник тез V Всеукраїнської науково-практичної конференції

здобувачів, аспірантів та молодих вчених «Прикладні інформаційні технології» (м. Вінниця, 20 лютого 2025 року). Вінниця: ДонНУ імені Василя Стуса, 2025. С. 85-87.

29. Лаптєва М. А., Гончар В. М. Різниця між SQL і NoSQL: переваги та обмеження реляційних баз даних. Збірник тез IV Всеукраїнської науково-практичної конференції «Комп'ютерні технології обробки даних» - 2023 (м. Вінниця, 23 вересня, 2024 року). Вінниця: ДонНУ імені Василя Стуса, 2024 С. 206-208.

30. Vosoughi S., Roy D., Aral S. The spread of true and false news online. Science. 2018. Vol. 359, № 6380. P. 1146–1151. DOI: https://doi.org/10.1126/science.aap9559

31. Devlin J., Chang M. W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // Proceedings of NAACL-HLT. 2019. P. 4171–4186. DOI: https://doi.org/10.48550/arXiv.1810.04805

32. Zhou X., Zafarani R. A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities. ACM Computing Surveys. 2020. Vol. 53(5). Article 109.

33. Лаптєва М. А., Ніколюк П. К. Використання дискретної математики у програмуванні. Збірник тез III Всеукраїнської науково-практичної конференції «Комп'ютерні технології обробки даних» - 2022 (м. Вінниця, 16 січня, 2023 року). Вінниця: ДонНУ імені Василя Стуса, 2023 С. 286-287.

34. FastAPI Documentation. URL: https://fastapi.tiangolo.com/

35. Aggarwal C. C. Neural Networks and Deep Learning. Cham : Springer, 2018. 497 p.

36. Jurafsky D., Martin J. H. Speech and Language Processing. 3rd ed. Stanford University, 2023. URL: https://web.stanford.edu/~jurafsky/slp3/

37. Bird S., Klein E., Loper E. Natural Language Processing with Python. Updated for Python 3. O'Reilly Media, 2023.

38. Birjali M., Kasri M., Beni-Hssane A. A comprehensive survey on sentiment analysis: Approaches, challenges and trends. Knowledge-Based Systems. 2021. Vol.

226. Article 107134. DOI: 10.1016/j.knosys.2021.107134

39. Chakraborty A., Paranjape B., Kakarla S., Ganguly N. Stop Clickbait: Detecting and Preventing Clickbaits in Online News Media. Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. 2016. P. 9–16.

40. Ferrara E., Varol O., Davis C., Menczer F., Flammini A. The rise of social bots. Communications of the ACM. 2016. Vol. 59, № 7. P. 96–104. DOI: https://doi.org/10.1145/2818717

41. Лаптєва М.А., Гончар В.М. Алгоритми знаходження кількості шляхів між вершинами в графах» Збірник тез IV Всеукраїнської науково-практичної конференції здобувачів, аспірантів та молодих вчених «Прикладні інформаційні технології» (м. Вінниця, 18 липня 2023 року). Вінниця: ДонНУ імені Василя Стуса, 2023 С. 140-142.

42. Cresci S., Di Pietro R., Petrocchi M., Spognardi A., Tesconi M. The paradigm-shift of social spambots. Proceedings of the 26th International Conference on World Wide Web Companion. 2017. P. 963–972. DOI: https://doi.org/10.1145/3041021.3055135.

43. Zhou X., Zafarani R. Fake News: A Survey of Research, Detection Methods, and Opportunities. ACM Computing Surveys. 2020. Vol. 53, № 5. DOI: https://doi.org/10.1145/3395046 .

44. Лаптєва М. А., Хмелівський Ю. С. Виявлення упереджень у ШІ за допомогою статистичного навчання. Збірник тез V Всеукраїнської науково-практичної конференції «Комп'ютерні технології обробки даних» - 2024 (м. Вінниця, 31 жовтня, 2025 року). Вінниця: ДонНУ імені Василя Стуса, 2025 С. 26-28.

45. Li W., Li S., Liu C., Lu L., Shi Z., Wen S. Span Identification and Technique Classification of Propaganda in News Articles. Complex & Intelligent Systems. 2021. DOI: https://doi.org/10.1007/s40747-021-00393-y.

ДОДАТКИ

ДОДАТОК А

Класифікація інформаційних загроз у соціальних мережах

Тип загрози	Визначення	Приклади	Намір	Рівень шкоди
Дезінформація	Свідоме поширення завідомо неправдивої інформації з метою введення аудиторії в оману	Фейкові новини, маніпульовані фото/відео, сфабриковані цитати	Умисний	Високий
Міс-інформація	Поширення хибних відомостей без свідомого наміру ввести в оману	Перепублікація застарілих новин, помилки в цифрах, чутки	Ненавмисний	Середній
Пропаганда	Систематичне поширення упередженої інформації для зміни системи цінностей і поведінки	Проросійські наративи, ідеологічні кампанії, ІПСО	Умисний	Високий
Маніпулятивний контент	Інформація, що використовує психологічні техніки для прихованого впливу на думку і поведінку	Клікбейт, емоційні заголовки, техніки "страху" і "терміновості"	Умисний	Високий
Deepfakes / синтетичний медіаконтент	Сфабриковані фото, відео або аудіо, створені за допомогою ШІ для імітації реальних людей	Підроблені відео політиків, синтетичні голоси, ШІ-генеровані зображення	Умисний	Високий
Ботові мережі	Автоматизовані акаунти, що масово поширюють контент для штучного збільшення охоплення	Координовані кампанії хештегів, масові лайки, штучний тренд	Умисний	Високий
Мова ворожнечі	Агресивні висловлювання з метою дискримінації або підбурювання до насильства щодо груп	Расистські дописи, ксенофобія, кібербулінг	Умисний	Високий
Ехо-камери / фільтр-бульбашки	Алгоритмічне замикання користувача в інформаційному просторі, де панує одна точка зору	Персоналізовані стрічки новин, рекомендаційні алгоритми	Структурний	Середній

ДОДАТОК Б

Функціональні вимоги до застосунку виявлення маніпулятивного контенту

Функціональна вимога	**Опис**
Введення URL публікації	Користувач повинен мати можливість ввести посилання на публікацію соціальної мережі для аналізу.
Збір даних про публікацію	Система повинна отримувати текст повідомлення, метадані та доступні взаємодії користувачів.
Аналіз маніпулятивних ознак	Система повинна виконувати автоматичне виявлення лінгвістичних та поведінкових ознак маніпулятивного контенту.
Визначення рівня ризику	Система повинна класифікувати контент за рівнями ризику (низький, середній, високий).
Аналіз ботоподібної активності	Система повинна оцінювати ймовірність ботоподібної поведінки акаунтів, які беруть участь у поширенні контенту.
Побудова графа поширення	Система повинна формувати граф взаємодій між користувачами на основі доступних даних.
Обчислення метрик поширення	Система повинна визначати характеристики каскаду поширення: глибину, розгалуження, швидкість розвитку та інші показники.
Візуалізація результатів	Система повинна відображати результати аналізу у зручному графічному вигляді.
Збереження результатів аналізу	Система повинна зберігати результати перевірок для подальшого використання.
Повторне використання результатів	Система повинна використовувати механізм кешування для зменшення кількості повторних обчислень.
Перегляд історії аналізів	Користувач повинен мати можливість переглядати раніше виконані перевірки.