Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____
Кафедра _Комп'ютерного моделювання та інтелектуальних технологій_

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти _____другий(магістерський)_____
_____Дослідження методів автоматизованої обробки звернень та_____
___генерації відповідей у страхових інформаційних системах на основі___
_____аналізу даних клієнта_____
(тема)

Виконав:
здобувач _2_ року навчання,
групи _СПРм-24-1__
_Дмитро ТЕЛЮКОВ_____
(власне ім'я, прізвище)
Спеціальність_122 Комп'ютерні науки_
(код і повна назва спеціальності)
Тип програми _освітньо-наукова_____
(освітньо-професійна або освітньо-наукова)
Освітня програма Системне проектування
(повна назва освітньої програми)
Керівник _доц. каф. КМІТ Зульфія ІМАНГУЛОВА_
(посада, власне ім'я, прізвище)

Допускається до захисту
Завідувач кафедри КМІТ _____ ___Ігор ГРЕБЕННІК_
(підпис) (власне ім'я, прізвище)

2026 р

Я як здобувач вищої освіти ХНУРЕ розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

12.05.2026  Дмитро ТЕЛЮКОВ

Кваліфікаційна робота не містить відомостей заборонених до відкритого опублікування.

Кваліфікаційна робота виконана у відповідності до стандартів, що діють в Україні.

Попередній захист проведено 12.05.2026 р.

Керівник кваліфікаційної роботи  Зульфія ІМАНГУЛОВА

Харківський національний університет радіоелектроніки

Факультет комп'ютерних наук
Кафедра комп'ютерного моделювання та інтелектуальних технологій
Рівень вищої освіти: другий (магістерський)
Спеціальність: 122 Комп'ютерні науки
(код і повна назва)
Тип програми: освітньо-наукова
(освітньо-професійна або освітньо-наукова)
Освітня програма: Системне проектування
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____»_____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Телюкову Дмитру Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи: «Дослідження методів автоматизованої обробки звернень та генерації відповідей у страхових інформаційних системах на основі аналізу даних клієнта»
затверджена наказом університету від 06 квітня 2026 р. № 268СТ
2. Термін подання здобувачем роботи до екзаменаційної комісії 15 травня 2026 р.
3. Вихідні дані до роботи: тема кваліфікаційної роботи; матеріали науково-дослідної практики; приклади німецькомовної страхової кореспонденції та PDF-вкладень; опис бізнес-процесів обробки звернень у страхових інформаційних системах; методи rule-based класифікації, машинного навчання, NLP, NER, RAG та LLM; локальний програмний прототип системи; тестовий набір тестових кейсів різної складності; результати експериментального порівняння Traditional та LLM+RAG; метрики Intent Accuracy, Intent Macro F1, Intent Micro F1, Field Micro F1, PDF Field Hit, latency p50, latency p90 та latency mean; вимоги до on-premise розгортання, конфіденційності та контрольованості обробки персональних даних.
4. Перелік питань, що потрібно опрацювати в роботі: аналіз предметної області страхових інформаційних систем та особливостей страхової кореспонденції; аналіз існуючих методів автоматизованої обробки звернень; проєктування архітектури системи обробки страхової кореспонденції; реалізація Traditional-підходу на основі правил, ML, NER, regex та шаблонів; реалізація LLM+RAG-підходу з використанням бази знань; організація експериментального дослідження та вибір метрик оцінювання; порівняльний аналіз результатів Traditional та LLM+RAG; обґрунтування гібридної каскадної моделі; визначення обмежень, практичної цінності та напрямів подальшого розвитку системи.
5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій: загальна архітектура системи автоматизованої обробки страхової кореспонденції; схема основних функціональних компонентів системи; pipeline обробки вхідного страхового звернення; схема алгоритму верифікації договору та пов'язаних документів; ER-схема бази даних системи; діаграма логіки

Traditional-підходу; діаграма логіки LLM+RAG-підходу; діаграма каскадної логіки гібридної моделі; діаграма компонентів гібридного підходу; таблиці результатів експериментального порівняння; каскадна логіка обробки звернення гібридним підходом; діаграма схеми гібридного підходу.

6. Консультанти розділів роботи (

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Розділи спеціальної частини	*доц. Імангулова З.І.*		12.05.2026

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Уточнення теми, мети, об'єкта, предмета та завдань кваліфікаційної роботи	06.04.2026	викон.
2	Аналіз предметної області страхових інформаційних систем та страхової кореспонденції	10.04.2026	викон.
3	Аналіз існуючих методів автоматизованої обробки звернень, NLP, ML, LLM та RAG	13.04.2026	викон.
4	Формування вимог до системи та проєктування загальної архітектури рішення	15.04.2026	викон.
5	Реалізація Traditional-підходу: rules, ML-класифікація, extraction, шаблонна відповідь	17.04.2026	викон.
6	Реалізація LLM+RAG-підходу та локальної бази знань	19.04.2026	викон.
7	Підготовка тестового набору кейсів і сценаріїв експериментального дослідження	22.04.2026	викон.
8	Проведення експериментального порівняння Traditional та LLM+RAG	24.04.2026	викон.
9	Аналіз метрик, результатів класифікації, вилучення полів, PDF Field Hit та latency	27.04.2026	викон.
10	Обґрунтування гібридної каскадної моделі	30.04.2026	викон.
11	Оформлення пояснювальної записки, рисунків, таблиць і переліку джерел	02.05.2026	викон.
12	Представлення до рецензування	1.05.2026	викон.
13	Представлення кваліфікаційної роботи	15.05.2026	викон.

Дата видачі завдання 06 04 2026 р.

Здобувач _____ Дмитро ТЕЛЮКОВ
(підпис)

Керівник роботи _____ доц. Зульфія ІМАНГУЛОВА
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 108 сторінок, 10 рисунків, 3 таблиці, 25 джерел, 2 додатка.

АВТОМАТИЗОВАНА ОБРОБКА ЗВЕРНЕНЬ, СТРАХОВА КОРЕСПОНДЕНЦІЯ, СТРАХОВА ІНФОРМАЦІЙНА СИСТЕМА, NLP, LLM, RAG, КЛАСИФІКАЦІЯ ЗВЕРНЕНЬ, ВИЛУЧЕННЯ ДАНИХ, ГЕНЕРАЦІЯ ВІДПОВІДЕЙ, ГІБРИДНА МОДЕЛЬ.

Об’єктом дослідження є процес обробки клієнтських звернень та страхової кореспонденції в інформаційних системах страхових організацій.

Предметом дослідження є моделі, методи та інструментальні засоби автоматизованого аналізу текстових звернень, вилучення з них значущої інформації та генерації відповідей.

Метою роботи є дослідження методів автоматизованої обробки звернень та генерації відповідей у страхових інформаційних системах на основі аналізу даних клієнта, а також проєктування прототипу системи для обробки німецькомовної страхової кореспонденції.

Для розв’язання поставлених задач використано методи аналізу предметної області, класифікації текстів, вилучення структурованих полів із неструктурованих текстів і PDF-документів, retrieval-augmented generation та верифікації даних за договірною базою.

За результатами експериментального дослідження встановлено, що детермінований підхід Traditional (Rules + ML) забезпечує високу швидкодію та якісне вилучення структурованих полів, тоді як підхід LLM+RAG краще працює зі складними зверненнями та зверненнями, що містять важливі дані у PDF-вкладеннях. Обґрунтовано доцільність застосування гібридної каскадної моделі з перевагами обох підходів.

Отримані результати можуть бути використані для створення та вдосконалення внутрішніх систем автоматизованої обробки страхової кореспонденції.

ABSTRACT

Explanatory note of the qualification work: 108 pages, 10 figures, 3 tables, 25 sources, 2 appendixes.

INSURANCE CORRESPONDENCE PROCESSING, INSURANCE INFORMATION SYSTEM, NATURAL LANGUAGE PROCESSING, LARGE LANGUAGE MODELS, RETRIEVAL-AUGMENTED GENERATION, INTENT CLASSIFICATION, DATA EXTRACTION, RESPONSE GENERATION, HYBRID MODEL

The object of the study is the process of processing client requests and insurance correspondence in insurance information systems. The subject of the study is models, methods, and software tools for automated analysis of text requests, extraction of significant information from them, and response generation.

The purpose of the work is to study methods for automated request processing and response generation in insurance information systems based on client data analysis, as well as to design a prototype system for processing German-language insurance correspondence.

The research methods include domain analysis, text classification, extraction of structured fields from unstructured text and PDF documents, retrieval-augmented generation, and verification of extracted data using the contract database. Experimental evaluation showed that the Traditional approach based on Rules and Machine Learning provides high speed and reliable extraction of structured fields, while the LLM+RAG approach performs better on paraphrased requests and requests containing important information in PDF attachments. Based on the obtained results, the feasibility of using a hybrid cascade model was substantiated, which used advantages of both approaches.

The practical significance of the work lies in the possibility of applying the obtained results to the development and improvement of internal systems for automated insurance correspondence.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК ТА ТЕРМІНІВ

Audit trail – журнал обробки або слід аудиту; послідовність записів про всі дії системи під час обробки кейсу, зокрема класифікацію, вилучення полів, верифікацію, генерацію відповіді та передачу на ручну перевірку;

BM25 – алгоритм ранжування документів у задачах інформаційного пошуку, що використовується для знаходження релевантних текстових фрагментів у базі знань;

Confidence – рівень упевненості системи у прийнятому рішенні, наприклад у визначеному intent або вилученому полі;

CRM – Customer Relationship Management, система керування взаємодією з клієнтами; програмне середовище для зберігання клієнтських даних, історії звернень і бізнес-процесів обслуговування;

Extraction – процес вилучення структурованих даних із неструктурованого або напівструктурованого тексту, наприклад номера договору, адреси, дати, суми або номера тікета;

Fact checking – перевірка фактів; етап контролю згенерованої відповіді, під час якого система перевіряє, чи всі твердження підтверджені вхідними даними, базою знань або договірною базою;

Field Micro F1 – метрика якості вилучення структурованих полів, що обчислюється як micro-усереднене значення F1-score для всіх вилучених реквізитів;

HITL – Human-in-the-loop, людина в контурі прийняття рішень; підхід, за якого складні або ризикові результати автоматичної обробки передаються фахівцю для ручної перевірки;

Intent – намір або тип звернення; клас, до якого система відносить вхідний лист, наприклад зміна адреси, запит щодо виплати, розірвання договору або запит щодо внеску;

Intent Accuracy – метрика точності класифікації intent, що показує частку правильно класифікованих звернень серед усіх тестових прикладів;

Intent Macro F1 – macro-усереднене значення F1-score для всіх класів intent, яке надає однакову вагу кожному класу незалежно від кількості прикладів;

Intent Micro F1 – micro-усереднене значення F1-score для задачі класифікації intent, яке враховує загальну кількість правильних і помилкових рішень за всіма класами;

Latency – затримка обробки; час, необхідний системі для виконання певного етапу або повного pipeline обробки звернення;

Latency Mean – середнє значення затримки обробки для всіх тестових кейсів;

Latency p50 – медіанне значення затримки обробки, нижче якого знаходиться 50% усіх виміряних результатів;

Latency p90 – значення затримки обробки, нижче якого знаходиться 90% усіх виміряних результатів;

LLM – Large Language Model, велика мовна модель; модель штучного інтелекту, призначена для аналізу, розуміння та генерації тексту природною мовою;

LLM+RAG – підхід, що поєднує велику мовну модель із механізмом пошуку релевантного контексту в базі знань;

ML – Machine Learning, машинне навчання; клас методів штучного інтелекту, що дозволяє будувати моделі на основі навчальних даних;

NER – Named Entity Recognition, розпізнавання іменованих сутностей, метод NLP для виявлення у тексті імен, адрес, організацій, дат, номерів договорів та інших сутностей;

NLP – Natural Language Processing, обробка природної мови; напрям штучного інтелекту, пов’язаний з аналізом, розумінням і генерацією тексту природною мовою;

Ollama – програмне середовище для локального запуску великих мовних моделей;

On-premise – локальне розгортання програмної системи у внутрішній інфраструктурі організації без передачі даних у зовнішні хмарні сервіси;

PDF Field Hit – метрика, що показує частку очікуваних полів із PDF-вкладень, які система змогла коректно знайти;

Pipeline – послідовність етапів обробки даних, через які проходить вхідне звернення: від приймання листа до формування результату;

RAG – Retrieval-Augmented Generation, генерація з доповненням пошуком; підхід, за якого мовна модель отримує додатковий релевантний контекст із бази знань перед формуванням відповіді;

Regex – regular expression, регулярний вираз; формальний шаблон для пошуку текстових фрагментів певного формату, наприклад номера договору, дати, суми або IBAN;

Rule-based підхід – підхід, заснований на правилах, ключових словах, регулярних виразах і логічних умовах;

TF-IDF – Term Frequency–Inverse Document Frequency, статистичний метод перетворення тексту на числове представлення з урахуванням важливості слів у корпусі документів;

Traditional-підхід – детермінований підхід до обробки страхової кореспонденції, що поєднує rule-based класифікацію, ML-класифікатор, регулярні вирази, NER, PDF-парсинг і шаблонну генерацію відповіді;

База знань – набір структурованих або напівструктурованих документів, що містять правила, інструкції, описи процесів і доменні відомості, які використовуються системою під час RAG-пошуку;

Страхова кореспонденція – сукупність електронних листів, повідомлень, вкладених документів і службових сповіщень, пов’язаних із супроводом страхових договорів і обслуговуванням клієнтів.

ВСТУП

Сучасні страхові організації та страхові посередники працюють в умовах постійного зростання обсягів цифрової комунікації з клієнтами, партнерами та страховими компаніями. Значна частина взаємодії здійснюється через електронну пошту, вебформи та вкладені документи, у яких містяться запити на зміну персональних даних, уточнення умов договору, повідомлення про страхові події, розірвання договорів та інші типи кореспонденції.

Однією з ключових проблем функціонування страхових інформаційних систем є автоматизована обробка вхідної кореспонденції, коли кількість листів і вкладень є значною, а зміст повідомлень може істотно відрізнятися за структурою, повнотою та стилем викладення. У традиційних підходах аналіз таких звернень виконується співробітниками вручну або з використанням простих шаблонних механізмів, що створює ризики затримок, суб’єктивності, перевантаження персоналу та помилок під час класифікації або підготовки відповіді. Крім того, значна частина корисної інформації міститься у неструктурованому тексті або вкладених PDF-документах, що ускладнює її формалізацію і подальше використання класичними засобами обробки даних. Тому виникає потреба у впровадженні інтелектуальних методів підтримки обробки звернень, здатних аналізувати зміст листів, вилучати значущі реквізити, враховувати контекст і формувати релевантні відповіді.

Метою дослідження є аналіз, реалізація та експериментальне порівняння методів автоматизованої обробки звернень і генерації відповідей у страхових інформаційних системах на основі аналізу даних клієнта, а також обґрунтування гібридної каскадної моделі для обробки німецькомовної страхової кореспонденції. Досягнення цієї мети передбачає послідовне вирішення низки взаємопов’язаних завдань, пов’язаних з аналізом предметної області страхових інформаційних систем, визначенням особливостей страхової кореспонденції як об’єкта автоматизації, дослідженням rule-based, ML, NLP, NER, LLM та RAG підходів, формуванням вимог до системи, проєктуванням її

архітектури, реалізацією базових підходів Traditional та LLM+RAG, проведенням експериментального порівняння за обраними метриками та подальшим обґрунтуванням гібридної моделі на основі отриманих результатів.

У межах роботи передбачається не лише теоретичний аналіз існуючих методів, а й практична перевірка їхньої придатності до страхової предметної області. Для цього реалізується Traditional-підхід, що поєднує правила, ML-класифікацію, регулярні вирази, NER, PDF-парсинг і шаблонну генерацію відповіді, а також LLM+RAG-підхід, який використовує локальну базу знань, пошук релевантного контексту та контрольовану генерацію відповіді мовною моделлю. Порівняння цих підходів виконується за метриками якості класифікації, вилучення структурованих полів, роботи з PDF-вкладеннями та часових характеристик обробки. На основі результатів такого порівняння обґрунтовується доцільність побудови гібридної каскадної моделі, у якій швидкий і контрольований deterministic/ML-контур використовується для типових звернень, а LLM+RAG залучається лише для складних, перефразованих або неоднозначних випадків.

Сфера застосування запропонованого рішення охоплює страхові компанії, страхових брокерів, страхових посередників та інші організації, діяльність яких пов’язана з обробкою значної кількості клієнтських звернень, службової кореспонденції та вкладених документів у цифровому середовищі. Запропонована система може використовуватися для попередньої класифікації звернень, вилучення структурованих реквізитів, перевірки даних за внутрішніми джерелами, підготовки чернеток відповідей і передачі складних або ризикових випадків на ручну перевірку. Практична цінність такого рішення полягає в можливості зменшення навантаження на співробітників, скорочення часу первинної обробки звернень, підвищення якості структуризації даних і стандартизації комунікації з клієнтами та страховиками.

Сфера застосування запропонованого рішення – страхові компанії, страхові брокери, страхові посередники та інші організації, діяльність яких

пов’язана з обробкою великої кількості клієнтських звернень і документів у цифровому середовищі.

Апробацію результатів роботи здійснено шляхом реалізації програмного прототипу системи, проведення експериментального порівняння Traditional та LLM+RAG-підходів на тестовому наборі страхових кейсів, а також оприлюднення окремих результатів дослідження в наукових публікаціях. Зокрема, результати щодо гібридної організації моделей для автоматизованої обробки страхової кореспонденції було представлено на 30-му Міжнародному молодіжному форумі «Радіоелектроніка та молодь у XXI столітті» у межах конференції «Інформаційні інтелектуальні системи», а також опубліковано у журналі «Grail of Science» за матеріалами VI International Scientific and Practical Conference “Open Science Nowadays: Main Mission, Trends and Instruments, Path and its Development” [1, 2].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз предметної області страхових інформаційних систем

Сучасні страхові інформаційні системи є складними організаційно-технічними комплексами, призначеними для підтримки основних бізнес-процесів страховика, страхового посередника або брокера. До таких процесів належать укладання та супровід договорів страхування, обробка звернень клієнтів, ведення історії взаємодії, розрахунок страхових платежів, врегулювання страхових випадків, підготовка документів, а також комунікація з клієнтами, партнерами та державними структурами. У межах цифровізації страхового ринку роль інформаційних систем постійно зростає, оскільки саме вони забезпечують швидкість обробки інформації, зменшення кількості ручних операцій, підвищення якості сервісу та зниження ймовірності помилок.

Особливе місце у структурі страхових інформаційних систем займає підсистема роботи зі зверненнями клієнтів і страховою кореспонденцією. У практичній діяльності страхових компаній та брокерів значна частина важливої інформації надходить саме через електронну пошту, вебформи, повідомлення у внутрішніх кабінетах користувачів, а також у вигляді вкладених документів. Такі звернення можуть містити запити на зміну персональних даних, уточнення умов договору, заяви про настання страхового випадку, запити на розірвання договору, прохання про надання довідок, копій полісів, рахунків або іншої документації. Усе це формує значний потік неструктурованої або напівструктурованої інформації, яка повинна бути оперативно проаналізована, класифікована та передана на подальше опрацювання.

На відміну від класичних транзакційних операцій, де інформація вводиться в чітко визначені поля бази даних, страхова кореспонденція має вільну текстову форму. Один і той самий запит може бути сформульований різними способами, з використанням різної термінології, скорочень, розмовних конструкцій або професійного жаргону. Крім того, частина важливої інформації

може міститися не в самому тексті листа, а у вкладених документах, наприклад у сканованих заявах, договорах, довідках, рахунках або підтвердженнях платежу. Це істотно ускладнює автоматизовану обробку таких звернень і вимагає використання спеціалізованих методів аналізу тексту, класифікації повідомлень, вилучення сутностей та генерації відповідей.

Ще однією важливою особливістю страхових інформаційних систем є високий рівень вимог до точності та надійності обробки даних. Помилки у трактуванні клієнтського звернення можуть призвести не лише до затримок у сервісі, але й до фінансових втрат, юридичних ризиків, порушення регуляторних вимог або погіршення репутації компанії. Саме тому автоматизація у цій сфері не може обмежуватися простим пошуком ключових слів. Вона повинна враховувати зміст повідомлення, контекст договору, наявність супровідних документів, історію попередньої взаємодії та правила обробки конкретного типу звернення.

Таким чином, страхові інформаційні системи сьогодні потребують інтелектуальних засобів підтримки обробки звернень, які поєднують можливості з аналізу тексту, структуризації вхідної інформації, прийняття рішень щодо подальшого маршруту обробки та формування коректної відповіді клієнту. Це робить задачу дослідження методів автоматизованої обробки страхових звернень актуальною як з наукової, так і з практичної точки зору.

1.2 Особливості страхової кореспонденції як об’єкта автоматизації

Страхова кореспонденція є специфічним видом інформаційного ресурсу, який має низку характеристик, що ускладнюють його автоматизовану обробку. Насамперед вона має переважно неструктурований або напівструктурований характер. Це означає, що повідомлення надходять у довільній текстовій формі, не мають жорстко визначеної структури та можуть істотно відрізнятися за обсягом, стилем, повнотою і якістю викладення інформації. Один клієнт може

коротко написати лише суть проблеми, інший – детально описати ситуацію на кілька абзаців, а ще інший – надати лише вкладення без пояснювального тексту.

Другою важливою особливістю є наявність професійної термінології та доменно-залежних формулювань. У страхуванні активно використовуються назви продуктів, типів договорів, страхових подій, внутрішніх процесів, фінансових показників, ролей учасників страхового процесу. Часто в листах можуть згадуватися номери договорів, полісів, заяв, рахунків, дані застрахованих осіб, реквізити роботодавців або посередників. Автоматизована система повинна коректно розпізнавати такі сутності та відрізняти їх від загального текстового наповнення.

Наступною ознакою є багатоканальність і різнорідність вхідних даних. Страхова кореспонденція може надходити не лише як простий текст електронного листа, а й у вигляді вкладень у форматах PDF, DOCX, JPG, PNG або сканованих документів. Часто вирішальна інформація міститься саме у вкладенні, а текст листа виконує лише роль короткого супроводу. У такому випадку система повинна не тільки аналізувати сам лист, а й уміти працювати з прикріпленими файлами, вилучати з них текстову інформацію, знаходити релевантні реквізити та враховувати їх у загальному рішенні.

Також важливо враховувати, що страхові звернення мають різний ступінь критичності. Частина з них є типовими інформаційними запитами, які можна обробити за шаблоном. Інші стосуються фінансових змін, юридично значущих дій або страхових випадків, де необхідна підвищена точність, перевірка повноти документів і, можливо, участь фахівця. У зв’язку з цим автоматизована система повинна не лише класифікувати звернення, а й визначати рівень впевненості у результаті та, за потреби, передавати складні випадки на ручне опрацювання.

Окрему складність становить мовна варіативність. У страховій практиці одне й те саме звернення може бути виражене різними словами, фразами або граматичними конструкціями. Наприклад, клієнт може повідомити про зміну адреси прямо, а може описати це непрямо, через пояснення переїзду чи

необхідність оновлення контактних даних. Тому система автоматизації повинна працювати не лише на рівні окремих ключових слів, але й на рівні смислового аналізу тексту.

Ще однією принциповою характеристикою є конфіденційність даних. Страхова кореспонденція часто містить персональні та фінансові відомості, інколи інформацію про стан здоров'я, майнові обставини або інші чутливі дані. Відповідно, під час автоматизації такої обробки необхідно враховувати вимоги щодо захисту персональних даних, обмеження на передачу інформації зовнішнім сервісам і необхідність побудови безпечної архітектури рішення.

Отже, страхова кореспонденція як об'єкт автоматизації характеризується неструктурованістю, доменною специфікою, наявністю вкладених документів, мовною варіативністю, різним рівнем критичності та високими вимогами до конфіденційності. Саме сукупність цих характеристик обумовлює потребу в застосуванні сучасних методів NLP, машинного навчання та гібридних підходів до обробки звернень у страхових інформаційних системах.

1.3 Актуальність теми дослідження

Актуальність теми дослідження визначається кількома взаємопов'язаними факторами. По-перше, у сучасних умовах страхові компанії та брокери працюють із великими обсягами цифрової комунікації, що постійно зростає. Значна кількість клієнтських звернень надходить у вигляді електронних листів та вкладених документів, а їх ручне опрацювання потребує суттєвих витрат часу та людських ресурсів. У разі збільшення навантаження це призводить до затримок у відповідях, зростання кількості помилок і зниження якості обслуговування.

По-друге, автоматизація стандартних операцій стає необхідною умовою підвищення ефективності діяльності страхових організацій. Велика частина звернень має повторюваний характер: зміна даних клієнта, уточнення статусу договору, надсилання документів, запити на довідки, стандартні консультації.

Такі звернення можуть бути частково або повністю автоматизовані за умови коректного визначення їх типу, вилучення ключових даних і формування релевантної відповіді.

По-третє, розвиток сучасних методів обробки природної мови, машинного навчання та великих мовних моделей відкриває нові можливості для побудови інтелектуальних систем підтримки прийняття рішень у страховій сфері. Якщо раніше автоматизація обмежувалася жорсткими правилами та шаблонами, то сьогодні з’являється можливість аналізувати зміст звернення глибше, ураховувати контекст, працювати з документами та створювати більш гнучкі механізми відповіді. Водночас ці технології потребують дослідження з точки зору їх придатності саме до страхового домену, де важливими є точність, пояснюваність результатів і контрольованість рішень.

По-четверте, актуальність теми підсилюється вимогами до захисту даних і правової відповідності інформаційних систем. У страховій сфері працюють із чутливою персональною інформацією, тому використання інтелектуальних методів має супроводжуватися дотриманням принципів безпеки, конфіденційності та мінімізації ризиків витоку даних. Це робить доцільним дослідження таких підходів, які можна інтегрувати у внутрішню інфраструктуру організації або адаптувати до обмежень конкретного середовища.

Крім того, тема має наукову актуальність, оскільки перебуває на перетині системного проєктування, аналізу даних, обробки природної мови та прикладних інформаційних систем. Дослідження методів автоматизованої обробки звернень дає змогу не лише створити практично корисне рішення, але й сформувати підхід до поєднання rule-based механізмів, методів машинного навчання та сучасних генеративних моделей у рамках єдиної системи.

Отже, актуальність теми дослідження полягає в об’єктивній потребі страхових організацій у підвищенні швидкості та якості обробки звернень, у можливості використання сучасних інтелектуальних методів для вирішення цієї задачі, а також у необхідності розроблення таких рішень, які поєднують

ефективність, надійність, безпечність і практичну придатність до умов реальної експлуатації.

1.4 Об’єкт, предмет, мета та завдання дослідження

Об’єктом дослідження є процеси обробки клієнтських звернень та страхової кореспонденції в інформаційних системах страхових організацій.

Предметом дослідження є моделі, методи та інструментальні засоби автоматизованого аналізу текстових звернень, вилучення з них значущої інформації та генерації відповідей у страхових інформаційних системах.

Метою дослідження є аналіз і обґрунтування підходів до автоматизованої обробки страхових звернень на основі сучасних методів аналізу тексту, машинного навчання та інтелектуальної обробки даних, а також проєктування рішення, придатного для використання в страхових інформаційних системах.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- дослідити особливості страхової кореспонденції як специфічного об’єкта автоматизованої обробки;
- проаналізувати існуючі підходи до класифікації звернень, вилучення сутностей і формування відповідей;
- визначити переваги та обмеження rule-based, ML-орієнтованих і LLM-орієнтованих підходів у межах страхової предметної області;
- сформувати вимоги до системи автоматизованої обробки звернень;
- запропонувати архітектуру та логіку функціонування відповідного рішення;
- розглянути можливість практичної реалізації прототипу системи та оцінити її прикладну цінність.

Практичне значення дослідження полягає в тому, що його результати можуть бути використані для створення або вдосконалення програмних засобів підтримки роботи з клієнтськими зверненнями, зменшення навантаження на

співробітників, скорочення часу відповіді на типові запити та підвищення загальної якості обслуговування у страховій сфері [3].

2 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ АВТОМАТИЗОВАНОЇ ОБРОБКИ ЗВЕРНЕНЬ

2.1 Традиційні підходи до обробки звернень у страхових інформаційних системах

У більшості страхових організацій обробка звернень клієнтів історично будувалася на основі поєднання ручної роботи співробітників, внутрішніх регламентів і стандартних інформаційних систем для реєстрації подій. Такий підхід передбачає, що вхідне повідомлення спочатку переглядається оператором, після чого визначається його тип, перевіряється наявність необхідних реквізитів, встановлюється відповідальний підрозділ і приймається рішення щодо подальших дій. За необхідності співробітник формує відповідь клієнту вручну або використовує один із підготовлених шаблонів. Подібна схема є зрозумілою з організаційної точки зору, але у випадку великого потоку звернень вона створює значне навантаження на персонал і вимагає суттєвих часових витрат.

Традиційні підходи мають певні переваги. Передусім вони є достатньо контрольованими, оскільки рішення приймає людина, яка може врахувати контекст, особливості клієнта, наявність документів і внутрішні правила компанії. Крім того, ручна обробка знижує ймовірність грубих помилок у складних або нестандартних випадках, особливо коли йдеться про юридично значущі дії, фінансові питання або страхові події. Проте ці переваги супроводжуються низкою суттєвих недоліків.

Основним недоліком традиційного ручного підходу є його низька масштабованість. Зі зростанням кількості клієнтів та цифрових каналів комунікації збільшується і число звернень, які потрібно опрацювати. У результаті виникають черги, затримки у відповіді, перевантаження працівників і підвищення ризику пропуску важливих деталей. Крім того, результати ручної обробки значною мірою залежать від досвіду конкретного співробітника, його

уважності та знання внутрішніх процедур, що може спричиняти нерівномірність якості роботи.

Іншим поширеним традиційним підходом є використання шаблонізованої обробки. У цьому випадку для найбільш типових звернень формуються заздалегідь підготовлені сценарії: наприклад, відповіді на запити щодо зміни адреси, повторного надсилання документів, уточнення умов договору або статусу обробки справи. Такий підхід дозволяє зменшити час на підготовку відповідей і частково стандартизувати комунікацію з клієнтом. Однак ефективність шаблонної обробки безпосередньо залежить від того, наскільки правильно працівник зможе віднести звернення до конкретного сценарію. За відсутності інтелектуальної підтримки навіть робота з шаблонами все одно вимагає участі людини на етапі аналізу та маршрутизації звернення [4].

Таким чином, традиційні методи обробки звернень забезпечують базовий рівень надійності та керованості процесу, проте вже не повною мірою відповідають вимогам сучасних страхових організацій, які працюють із великим обсягом цифрової кореспонденції. Це створює передумови для переходу до автоматизованих або напівавтоматизованих підходів, здатних поєднати продуктивність із достатнім рівнем точності та контролю.

2.2 Rule-based підхід у задачах класифікації та маршрутизації звернень

Одним із найпростіших і водночас найбільш практичних підходів до автоматизованої обробки звернень є rule-based підхід, тобто підхід, заснований на правилах [5]. Його суть полягає у використанні заздалегідь визначених умов, шаблонів, ключових слів, регулярних виразів та логічних конструкцій, за допомогою яких система аналізує текст звернення й приймає рішення щодо його категорії, пріоритету або подальшої маршрутизації.

У страховій сфері rule-based підхід може бути ефективним для виявлення типових сценаріїв, що мають відносно стабільні мовні ознаки. Наприклад, якщо в листі містяться формулювання, пов’язані зі зміною адреси, банківських

реквізитів або запитом на повторне надсилання полісу, така кореспонденція може бути віднесена до відповідного класу на основі словників і шаблонів. Подібним чином можуть визначатися заяви про розірвання договору, повідомлення про страхову подію, запити на довідки або звернення щодо виплат. Для цього застосовуються не лише окремі ключові слова, а й комбінації слів, мовні конструкції, наявність певних реквізитів, вкладень чи номерів договорів.

Перевагою rule-based підходу є його прозорість і пояснюваність. Кожне рішення системи можна обґрунтувати конкретним правилом, що спрацювало під час аналізу тексту. Це особливо важливо в прикладних системах, де результати автоматизації повинні бути контрольованими та зрозумілими для користувача або адміністратора. Крім того, правила легко узгоджувати з внутрішніми бізнес-процесами, корпоративними регламентами та вимогами безпеки. Ще однією перевагою є відносно невисокі обчислювальні витрати: rule-based система не потребує тривалого навчання, складної інфраструктури або великого обсягу розмічених даних.

Разом з тим, підхід на основі правил має суттєві обмеження. Передусім він погано працює в умовах мовної варіативності, коли одна й та сама думка може бути виражена різними словами або непрямими формулюваннями. Якщо повідомлення не містить передбачених ключових ознак, система може не розпізнати його правильно. Іншою проблемою є складність підтримки великої кількості правил. Зі зростанням числа сценаріїв обробки система стає громіздкою, важкою в супроводі та схильною до конфліктів між правилами. Окремі правила можуть частково перекривати одне одного, що ускладнює логіку прийняття рішень.

У задачах маршрутизації звернень rule-based підхід часто є доцільним як перший рівень фільтрації. Він дозволяє швидко виявляти очевидні та добре формалізовані випадки, а складніші або неоднозначні повідомлення передавати на наступний рівень аналізу. Саме тому в сучасних системах автоматизації

обробки звернень правила доцільно розглядати не як універсальне рішення, а як один із компонентів багаторівневої інтелектуальної архітектури.

2.3 Методи машинного навчання для аналізу текстових звернень

Подальшим кроком у розвитку автоматизованої обробки звернень стало застосування методів машинного навчання. На відміну від rule-based систем, які спираються на явно задані правила, методи машинного навчання будують модель на основі прикладів і навчаються виявляти закономірності в текстових даних. Це дозволяє краще враховувати мовну різноманітність і підвищувати якість класифікації в тих випадках, коли ручне описання всіх можливих правил є складним або практично неможливим [6].

У задачах аналізу страхових звернень методи машинного навчання можуть застосовуватися для класифікації повідомлень за типами, визначення пріоритету, виявлення наміру користувача, вилучення окремих сутностей, а також для виявлення подібності між зверненнями. Найчастіше у практичних системах використовуються алгоритми текстової класифікації, які працюють із векторними поданнями тексту. На базовому рівні це можуть бути статистичні моделі, що використовують частотні ознаки слів, n-грам або TF-IDF-подання тексту. Такі моделі порівняно прості в реалізації, швидко навчаються та можуть показувати достатньо добрі результати на задачах із чітко визначеними класами звернень.

Перевага методів машинного навчання полягає в тому, що вони здатні узагальнювати знання на нові приклади. Якщо система навчена на достатній кількості репрезентативних звернень, вона може коректно розпізнавати навіть ті повідомлення, які не містять точних збігів із попередньо визначеними шаблонами. Це особливо корисно в умовах, коли користувачі формулюють запити по-різному, використовують синонімічні вирази, орфографічні варіанти або неповні речення.

Однак ефективність методів машинного навчання суттєво залежить від якості навчальних даних. Для побудови надійної моделі потрібен достатній обсяг розмічених прикладів, що відображають реальну різноманітність страхових звернень. У страховій сфері збирання та підготовка такого корпусу може бути проблематичною через конфіденційність даних, наявність персональної інформації та різнорідність вхідних документів. Крім того, навчена модель може втрачати актуальність у разі зміни типових сценаріїв звернень, появи нових продуктів або змін у внутрішніх процесах компанії.

Ще одним важливим обмеженням є пояснюваність результатів. На відміну від rule-based систем, де можна чітко вказати, яке правило спрацювало, у випадку статистичної моделі пояснення часто є менш очевидним. Для прикладних страхових систем це може створювати труднощі під час аудиту, перевірки якості або погодження автоматизованих рішень із бізнес-користувачами.

Отже, методи машинного навчання є більш гнучкими порівняно з підходом на основі правил і дозволяють підвищити якість аналізу текстових звернень. Водночас їх впровадження потребує наявності даних, налаштування процесів навчання та оцінювання, а також врахування вимог до надійності й інтерпретованості результатів.

2.4 Використання NLP та LLM для обробки страхових листів

Сучасний етап розвитку інтелектуальних інформаційних систем пов’язаний із широким застосуванням методів обробки природної мови та великих мовних моделей. На відміну від класичних статистичних підходів, сучасні NLP-методи здатні враховувати контекст слів у реченні, семантичні зв’язки між фрагментами тексту та загальний зміст повідомлення. Це відкриває нові можливості для аналізу страхових листів, у яких важлива інформація часто розподілена по всьому тексту, виражена непрямо або супроводжується вкладеними документами [5, 7].

Методи NLP можуть використовуватися для токенізації тексту, нормалізації словоформ, виявлення іменованих сутностей, синтаксичного та семантичного аналізу, визначення намірів користувача, а також для пошуку релевантної інформації в документах. У страховому контексті це дозволяє автоматично виявляти номери договорів, дати, адреси, персональні дані, типи страхових продуктів, згадки про зміни умов договору чи настання події. У поєднанні з аналізом вкладень такі методи можуть формувати більш повне уявлення про звернення та покращувати якість подальшої обробки.

Окремий інтерес становлять великі мовні моделі, які здатні не лише аналізувати текст, але й генерувати змістовні відповіді природною мовою. Для задачі автоматизованої обробки страхових звернень це означає можливість створення систем, які не просто визначають тип звернення, а й формують попередню відповідь клієнту, ставлять уточнювальні питання, резюмують зміст листа або готують структуроване подання отриманої інформації для співробітника. Такі можливості особливо цінні у випадках, коли звернення є складним, містить кілька намірів або вимагає узгодженого поєднання даних із листа, вкладень та внутрішніх баз знань.

Водночас використання LLM у страхових інформаційних системах пов’язане з низкою ризиків. По-перше, великі мовні моделі можуть генерувати неточну або вигадану інформацію, якщо не мають достатнього контексту або працюють без зовнішньої перевірки джерел. У страхуванні така поведінка є небажаною, оскільки відповіді клієнтам повинні бути коректними, юридично безпечними та узгодженими з фактичними даними договору. По-друге, застосування зовнішніх хмарних сервісів для обробки листів може суперечити вимогам щодо конфіденційності та захисту персональних даних. По-третє, великі моделі мають вищі вимоги до обчислювальних ресурсів, що може ускладнювати їх локальне використання в корпоративному середовищі.

У зв’язку з цим найбільш перспективним є не безумовне заміщення попередніх підходів LLM-моделями, а їх контрольоване використання у складі комбінованих рішень. Великі мовні моделі можуть застосовуватися там, де

потрібен глибший аналіз контексту, узагальнення змісту або генерація тексту, але з обов’язковим обмеженням через правила, зовнішні джерела знань, перевірку даних і механізми контролю впевненості.

2.5 Порівняльний аналіз підходів та обґрунтування вибору напряму дослідження

Розглянуті підходи до автоматизованої обробки звернень мають різні властивості, переваги та обмеження. Традиційна ручна обробка забезпечує високий рівень контролю й можливість врахування нюансів конкретної ситуації, однак є повільною, дорогою з точки зору трудових ресурсів і малоефективною при масштабуванні. Rule-based підхід дозволяє швидко автоматизувати типові сценарії, є прозорим і добре підходить для суворо регламентованих бізнес-процесів, але погано адаптується до мовної різноманітності та нестандартних формулювань. Методи машинного навчання забезпечують вищу гнучкість у задачах класифікації текстів, проте потребують навчальних даних і не завжди достатньо пояснювані. Сучасні NLP-методи та великі мовні моделі мають найширший функціональний потенціал, але потребують особливо обережного використання через ризики неточностей, складність контролю і підвищені вимоги до ресурсів та захисту даних.

Для страхових інформаційних систем критично важливими є не лише точність, а й надійність, керованість, відповідність внутрішнім правилам та можливість практичного впровадження [8]. Саме тому жоден із розглянутих підходів у чистому вигляді не є універсальним. Найбільш доцільним виглядає поєднання їх сильних сторін у межах єдиної системи. Наприклад, правила можуть використовуватися для первинного виявлення очевидних сценаріїв і контролю критичних умов, моделі машинного навчання – для класифікації текстів зі змінною мовною формою, а методи NLP та LLM – для складного аналізу контексту, вилучення інформації з неструктурованих фрагментів і формування проектів відповідей.

Таким чином, аналіз існуючих методів показує доцільність дослідження саме комбінованого або гібридного підходу до автоматизованої обробки страхових звернень. Такий напрям дозволяє зберегти переваги прозорості та контрольованості там, де це критично, і водночас використати можливості сучасних інтелектуальних методів там, де традиційні засоби виявляються недостатніми. Саме гібридний підхід доцільно покласти в основу подальшого проєктування системи, оскільки він найкраще відповідає специфіці страхової предметної області та вимогам практичного використання [9].

3 ПРОЄКТУВАННЯ АРХІТЕКТУРИ

3.1 Архітектура запропонованої системи

У межах дослідження запропоновано архітектуру системи автоматизованої обробки страхової кореспонденції, орієнтовану на використання у внутрішньому контурі страхової організації або брокерської компанії. Однією з базових вимог до такої системи є можливість локального розгортання без передачі даних у зовнішні сервіси, що особливо важливо з огляду на конфіденційність страхових даних, наявність персональної інформації клієнтів і потребу дотримання вимог безпеки. Саме тому система реалізована як on-premise рішення, здатне працювати у внутрішній інфраструктурі організації.

Архітектурно система побудована за шаблоном Ports & Adapters, або гексагональною архітектурою [10]. Такий підхід дозволяє чітко відокремити предметну логіку, прикладні сценарії, зовнішні джерела даних і технічні механізми інтеграції. У центральній частині системи розміщується доменний і прикладний рівень, де визначено основні сутності, сценарії обробки та правила бізнес-логіки. Зовнішні компоненти, такі як база даних, файлове сховище, модулі машинного навчання, мовні моделі, модулі анонімізації та модулі вилучення тексту з документів, підключаються через стандартизовані порти й адаптери. Завдяки цьому можлива заміна окремих технічних реалізацій без зміни прикладної логіки системи. Наприклад, локальне файлове сховище може бути надалі замінене на об’єктне сховище, а тестовий LLM-клієнт – на реальну локальну модель.

На рівні прикладної логіки система підтримує кілька основних сценаріїв:

- приймання нового звернення;
- обробку кейсу;
- отримання структурованих результатів;
- генерацію чернетки відповіді;

- переіндексацію бази знань;
- повторне навчання ML-моделі та ручну роботу з випадками низької впевненості.

Усі ці сценарії реалізуються через окремі use case-компоненти, що забезпечує модульність і придатність архітектури до подальшого розвитку.

Важливою перевагою запропонованої архітектури є також її тестованість. Оскільки предметна логіка не залежить безпосередньо від конкретних адаптерів, окремі компоненти можуть перевірятися ізольовано. Це спрощує модульне тестування, інтеграційне тестування та проведення експериментів над різними варіантами реалізації системи. Загальна архітектура системи автоматизованої обробки страхової кореспонденції зображена на рисунку 3.1.

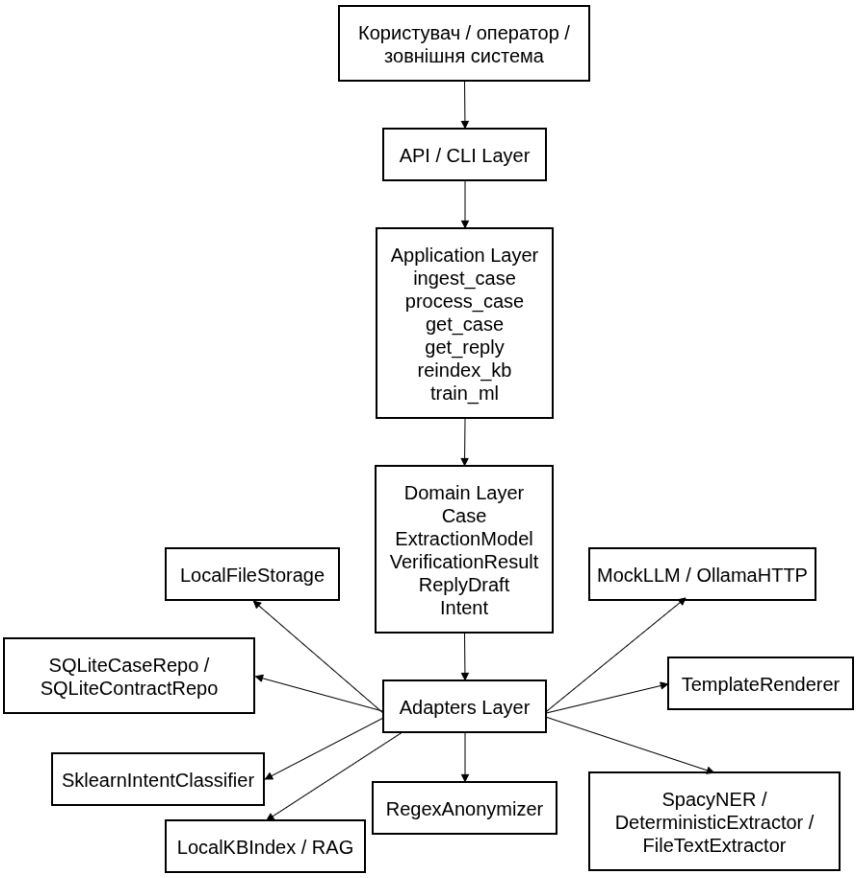

Рисунок 3.1 – Загальна архітектура системи автоматизованої обробки страхової кореспонденції

Отже, обрана архітектура забезпечує поєднання гнучкості, керованості, ізоляції доменної логіки та практичної зручності розгортання, що є доцільним для системи автоматизованої обробки страхової кореспонденції.

3.2 Основні компоненти системи

Функціонально система, наведена на рисунку 3.2, складається з кількох взаємопов’язаних компонентів, кожен з яких виконує окрему роль у загальному процесі обробки звернення.

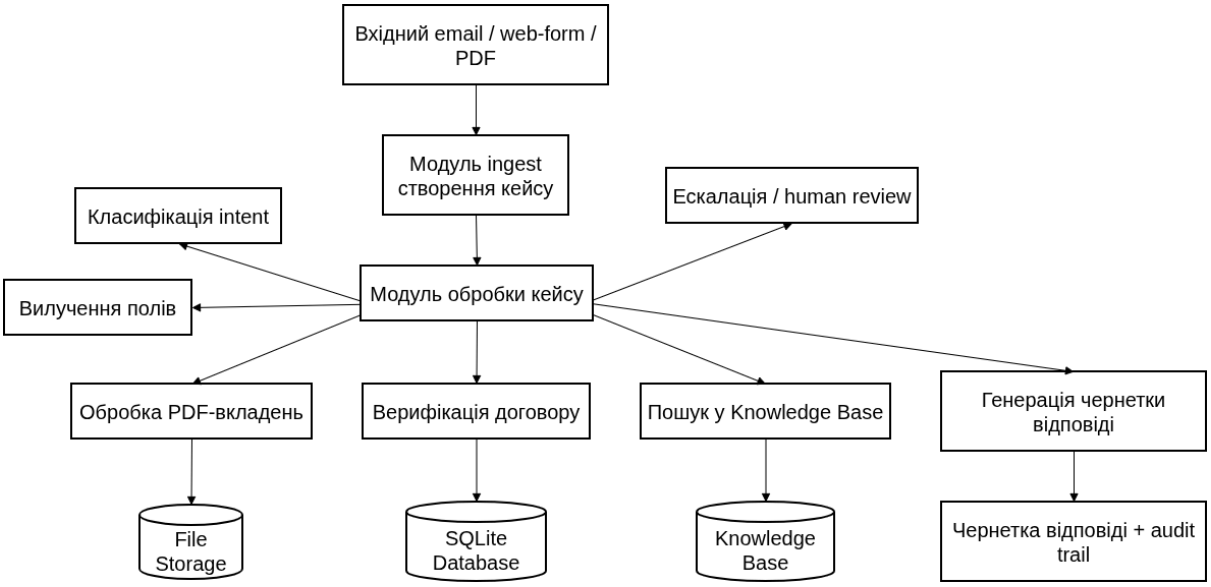

Рисунок 3.2 – Основні функціональні компоненти системи

Центральним елементом є модуль керування кейсами, який відповідає за створення кейсу на основі вхідного листа, збереження теми, відправника, тексту повідомлення, часу створення, вкладень, а також подальших результатів обробки. У моделі даних кейс виступає основною сутністю, з якою пов’язані визначений intent, рівень впевненості класифікації, результати вилучення полів, згенерована відповідь і повний журнал проходження pipeline.

Наступним важливим компонентом є модуль класифікації звернень. Він використовується для визначення типу запиту клієнта, наприклад зміни адреси,

запиту щодо виплати, зміни внесків, розірвання договору або іншого сценарію. У системі передбачено десять цільових типів звернень, серед яких також є службовий клас UNKNOWN для невизначених випадків. Для розпізнавання intent застосовуються три можливі механізми: правила, модель машинного навчання та LLM-класифікація.

Модуль вилучення даних відповідає за перетворення неструктурованого тексту листа та вкладених документів у структурований набір полів. Серед таких полів можуть бути номер договору, адреса, дата, сума, IBAN, роль відправника, номер тікета, перелік відсутніх даних та додаткова службова інформація. Для цього в системі поєднуються кілька методів: детерміноване вилучення за регулярними виразами, використання NER-модуля для пошуку іменованих сутностей та аналіз тексту, вилученого з PDF-вкладень. Об’єднання результатів із різних джерел дозволяє підвищити повноту та точність сформованої структури даних.

Окреме місце займає модуль верифікації договорів і документів. Після вилучення номера договору система звертається до репозиторію договорів, перевіряє факт існування контракту, його статус, наявність пов’язаного клієнта, а в окремих сценаріях – наявність необхідних документів у файловому сховищі. Результат цієї перевірки оформлюється як окрема структура верифікації, що далі враховується під час маршрутизації кейсу, ескалації до ручного опрацювання та формування відповіді клієнту. Такий підхід забезпечує важливу перевагу: генерація відповіді спирається не лише на текст листа, а й на перевірені факти з договірної бази.

Для роботи з неструктурованими доменними знаннями використовується база знань і модуль RAG [11]. У ролі джерел знань виступають markdown-документи, присвячені типовим страховим процесам, наприклад зміні адреси, запитам на виплати, зміні внесків чи поновленню договору. Пошук у базі знань реалізується як гібридне поєднання BM25 та векторного пошуку [12, 13]. Крім того, для такої теми дослідження як обробка страхових звернень застосовується політика доступу до чутливих знань, де пошук

обмежується наміром звернення та рівнем допустимої чутливості. Це зменшує ризик використання нерелевантних фрагментів під час генерації відповіді.

Компонент генерації відповіді працює на основі шаблонів Jinja2 та, за потреби, LLM-поліпшення тексту. Для кожного типу запиту передбачено окремий шаблон, який враховує як результати вилучення полів, так і контекст верифікації договору. Наприклад, якщо договір розірваний, текст відповіді буде відрізнятися від стандартного сценарію для активного договору. Якщо ж система виявила брак обов'язкового документа, відповідь буде містити прохання надати відповідні матеріали.

Додатково в системі присутні модулі анонімізації персональних даних, журналювання, черги на ручну перевірку, активного навчання моделі та REST API для зовнішньої взаємодії. У сукупності ці компоненти формують повноцінне прикладне рішення для автоматизованої обробки страхової кореспонденції.

3.3 Сценарій обробки вхідного звернення

Pipeline обробки вхідного страхового звернення зображений на рисунку 3.3 Процес обробки вхідного звернення починається зі створення кейсу на основі листа, що надійшов у систему. Під час цього етапу зберігаються тема, адреса відправника, текст повідомлення, дата створення та вкладення. Якщо до листа прикріплено файли, їх байтове представлення розміщується у файловому сховищі, а в базі даних зберігаються лише метадані та посилання на розміщення файлу. Такий підхід спрощує подальшу роботу з документами та не перевантажує основну таблицю кейсів великими двійковими даними.

Після створення кейсу запускається сценарій обробки даних. На першому етапі система аналізує текст через набір правил і шаблонів, намагаючись визначити intent за ключовими словами, регулярними виразами та іншими детермінованими ознаками. Якщо впевненість rules-модуля є достатньою, intent приймається без переходу до наступних джерел класифікації. Якщо ж

rules-рівень не забезпечує надійного результату, система переходить до ML-класифікатора. Якщо і цього недостатньо, використовується LLM-класифікація як фінальний резервний механізм. Така логіка дозволяє поєднати швидкість і передбачуваність rules/ML-підходів із гнучкістю мовної моделі для нестандартних випадків.

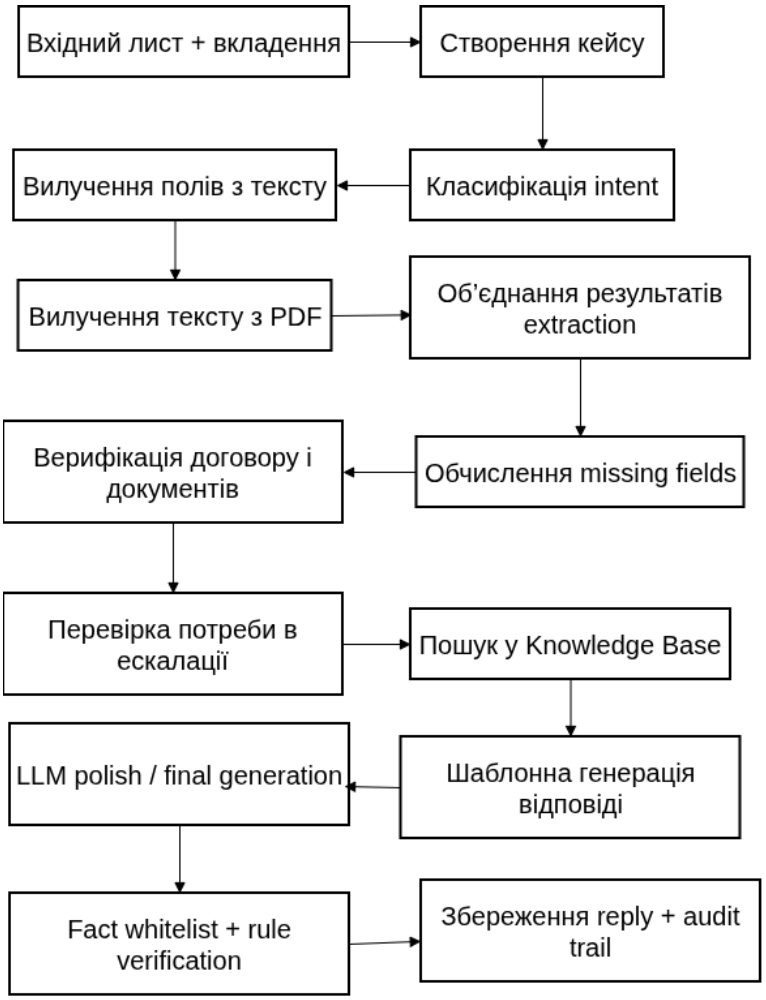

Рисунок 3.3 – Pipeline обробки вхідного страхового звернення

Після визначення типу звернення виконується вилучення значущих полів. На цьому етапі текст листа проходить через NER-модуль і детерміновані екстрактори, а вкладення у форматі PDF – через модуль вилучення тексту. Отримані результати зливаються в єдину структуру, після чого для конкретного intent обчислюється перелік відсутніх обов'язкових полів. Якщо система не має

упевненості, можлива додаткова спроба заповнення частини пропущених полів за допомогою LLM, але вже після первинної детермінованої обробки.

Далі система виконує верифікацію за договірною базою, алгоритм якої зображений на рисунку 3.4. За номером договору здійснюється пошук контракту, перевірка його статусу, пошук клієнта та, якщо того вимагає конкретний тип звернення, перевірка наявності документів відповідного типу. У випадку, коли номер договору відсутній, контракт не знайдено, договір розірвано або потрібного документа немає у файловому сховищі, система не просто фіксує це як технічний результат, а використовує як фактор для подальшого рішення: чи можна сформувати відповідь автоматично, чи слід поставити уточнювальне запитання, чи потрібно ескалювати кейс для ручного опрацювання.

На наступному етапі за потреби виконується пошук у базі знань. Результати пошуку передаються в модуль генерації відповіді як контекст, який доповнює шаблон та перевірені факти з договорної бази. Це дозволяє сформувати більш змістовну й професійну відповідь без виходу за межі контрольованого доменного контексту.

Завершальним етапом є формування чернетки відповіді. Спочатку система рендерить шаблон для відповідного intent із підстановкою вилучених полів і результатів верифікації. Далі мовна модель виконує стилістичне редагування тексту. Після цього відповідь проходить перевірку fact whitelist і rule-based верифікацію, які гарантують, що у згенерованому тексті не з’явилися нові неперевірені факти, а всі потрібні уточнювальні запитання дійсно поставлені. Якщо рівень впевненості занадто низький або верифікація виявила критичні проблеми, кейс потрапляє до черги ручної перевірки. Усі кроки pipeline записуються до audit trail, що дає можливість простежити хід прийняття рішення по кожному кейсу.

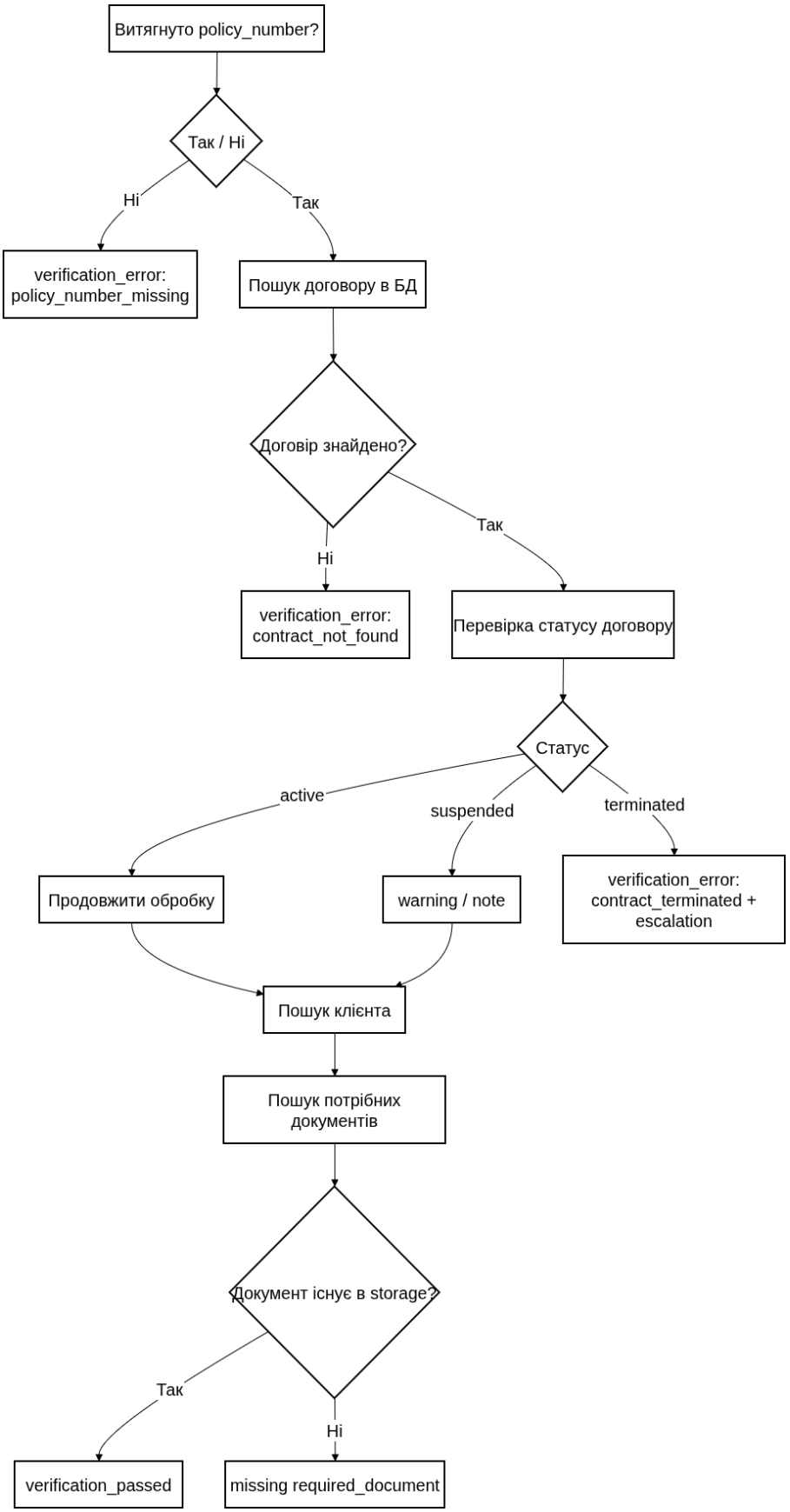

Рисунок 3.4 – Схема алгоритму верифікації договору та пов’язаних документів

3.4 Реалізація Traditional-підходу

Traditional-підхід у межах прототипу реалізовано як базовий детермінований контур обробки страхової кореспонденції, який поєднує правила, модель машинного навчання, регулярні вирази, NER-модуль, PDF-парсер і шаблонну генерацію відповіді. Основна ідея цього підходу полягає в тому, що найбільш типові страхові звернення можуть бути оброблені без залучення великих мовних моделей, оскільки вони мають повторювану структуру, характерні доменні формулювання та формалізовані реквізити.

У Traditional-підході процес обробки починається з аналізу тексту листа за допомогою rule-based-модуля. Цей модуль містить набір правил, ключових слів, регулярних виразів і доменних шаблонів, які дозволяють визначити ймовірний тип звернення. Наприклад, наявність у тексті формулювань, пов’язаних зі зміною адреси, розірванням договору, додатковим внеском, виплатою або запитом щодо страхового внеску, може бути використана як ознака відповідного intent. Результатом роботи rule-based-рівня є попередньо визначений клас звернення, рівень упевненості та перелік причин, які пояснюють прийняте рішення.

Якщо правило дає достатньо надійний результат, система може прийняти визначений intent без подальшого ускладнення обробки. Якщо ж звернення не має чітких ключових ознак або правила повертають недостатній рівень упевненості, використовується ML-класифікатор. У прототипі для цього застосовується модель текстової класифікації на основі TF-IDF-представлення тексту та алгоритму Logistic Regression. TF-IDF дозволяє перетворити текст листа на числовий вектор, що відображає важливість окремих слів і фрагментів у межах корпусу, а Logistic Regression виконує класифікацію звернення за одним із наперед визначених класів.

Використання ML-класифікатора дозволяє частково подолати обмеження rule-based-підходу. Якщо користувач формулює звернення не точно за

очікуваним шаблоном, але зберігає загальний семантичний зв’язок із певним типом запиту, модель може правильно визначити intent на основі статистичних ознак тексту. Водночас така модель залишається достатньо легкою з точки зору обчислювальних ресурсів і може працювати локально без використання зовнішніх сервісів.

Після визначення intent виконується вилучення структурованих полів. Для цього Traditional-підхід використовує комбінацію регулярних виразів, NER-модуля та аналізу PDF-вкладень. Регулярні вирази застосовуються для пошуку реквізитів, які мають чітко виражений формат: номерів договорів, номерів тікетів, сум, дат, IBAN, числових значень внесків та інших формалізованих даних. Такий спосіб є особливо ефективним для страхових документів, де значна частина важливої інформації подається у стандартизованому вигляді.

NER-модуль використовується для виявлення іменованих сутностей, зокрема імен осіб, адрес, організацій або інших текстових елементів, які важче описати простими регулярними виразами. У поєднанні з deterministic extraction це дозволяє системі сформувати більш повну структуру даних щодо звернення. Якщо до листа додано PDF-документ, система спочатку вилучає з нього текст, після чого застосовує до цього тексту ті самі механізми аналізу, що й до основного тіла повідомлення.

Важливою частиною Traditional-підходу є визначення відсутніх обов’язкових полів. Для кожного intent задається перелік реквізитів, необхідних для подальшої автоматизованої обробки. Наприклад, для звернення щодо зміни адреси важливими можуть бути номер договору та нова адреса, для звернення щодо виплати – номер договору та дата або тип виплати, для звернення щодо додаткового внеску – номер договору й сума. Якщо частина обов’язкових полів відсутня, система фіксує це в результаті обробки та може сформувати відповідь із проханням надати додаткові дані.

Після класифікації та вилучення полів виконується перевірка даних за договірною базою. Якщо в тексті знайдено номер договору, система звертається

до відповідного репозиторію, перевіряє наявність договору, його статус, пов’язаного клієнта та, за потреби, наявність документів у файловому сховищі. Така верифікація є важливою, оскільки дозволяє будувати відповідь не лише на основі тексту листа, а й на основі перевірених структурованих даних.

Генерація відповіді у Traditional-підході виконується за допомогою шаблонів. Для кожного intent передбачено окремий шаблон відповіді, у який підставляються вилучені поля, результати верифікації та інформація про відсутні дані. Такий підхід забезпечує контрольованість, передбачуваність і стилістичну стабільність відповіді. Він також зменшує ризик появи неперевірених фактів, оскільки система не генерує відповідь вільно, а формує її на основі заздалегідь визначеної структури.

Узагальнена логіка Traditional-підходу подана на рисунку 3.5. Перевагою Traditional-підходу є висока швидкодія, низькі вимоги до ресурсів, відтворюваність результатів і надійне вилучення формалізованих полів. Він добре підходить для типових страхових звернень, де структура запиту достатньо передбачувана, а критично важливі реквізити мають формальний вигляд. Водночас цей підхід має обмеження у випадках перефразованих, неповних або семантично складних звернень, де зміст не можна надійно визначити лише за ключовими словами, статистичними ознаками або регулярними виразами. Саме тому для порівняння в межах дослідження було також реалізовано LLM+RAG-підхід.

3.5 Реалізація LLM+RAG-підходу

LLM+RAG-підхід у межах прототипу реалізовано як альтернативний інтелектуальний контур обробки страхової кореспонденції, орієнтований на роботу зі складнішими, менш структурованими або перефразованими зверненнями. На відміну від Traditional-підходу, де основна логіка базується на правилах, ML-класифікаторі та шаблонах, LLM+RAG використовує велику мовну модель у поєднанні з пошуком релевантної інформації в базі знань.

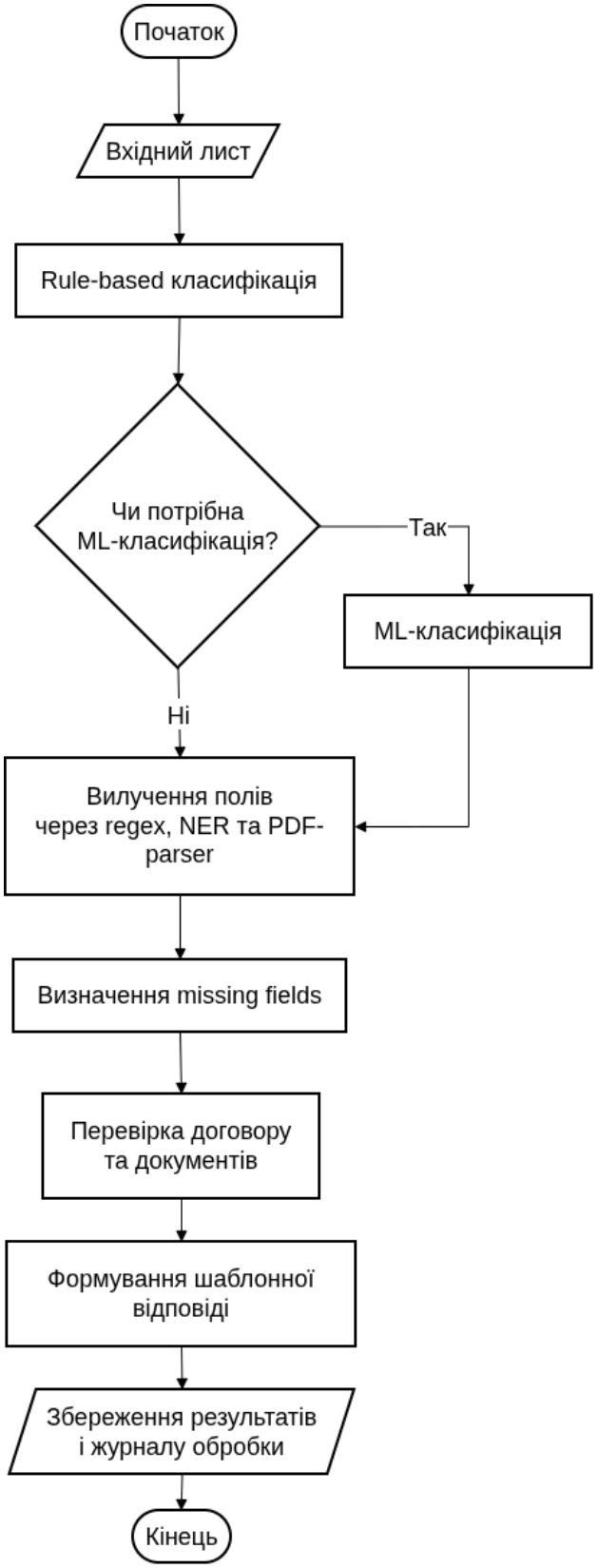

Рисунок 3.5 – Діаграма узагальненої логіки традиційного підходу

Застосування RAG, тобто retrieval-augmented generation, обумовлене тим, що велика мовна модель сама по собі не має гарантованого доступу до актуальних внутрішніх правил, процесів і доменних інструкцій конкретної страхової організації. Тому перед генерацією відповіді система спочатку виконує пошук релевантних фрагментів у локальній базі знань, а вже потім передає ці фрагменти у prompt мовної моделі. Це дозволяє обмежити генерацію відповідей доменним контекстом і зменшити ризик формування довільних або неперевірених тверджень.

База знань у прототипі організована у вигляді набору markdown-документів, кожен із яких описує окремий страховий процес або типовий сценарій обробки. Наприклад, окремі документи можуть містити правила щодо зміни адреси, запитів на виплату, додаткових внесків, поновлення договору, розірвання договору або обробки службових повідомлень від страховика. Перед використанням ці документи індексуються, після чого система може виконувати пошук релевантного контексту для конкретного вхідного звернення.

Пошук у базі знань може поєднувати лексичні та векторні методи. Лексичний пошук дозволяє знаходити документи за збігом ключових слів і доменних термінів, а векторний або семантичний пошук дає змогу знаходити релевантні фрагменти навіть тоді, коли формулювання у зверненні відрізняється від формулювань у базі знань. У межах досліджуваного прототипу такий підхід використовується для підготовки контексту, який передається мовній моделі разом із текстом листа та структурованими даними.

Обробка звернення в LLM+RAG-підході починається з підготовки вхідного контексту. Система отримує тему листа, текст повідомлення, текст із вкладених PDF-документів, якщо такі є, а також доступні метадані кейсу. За потреби виконується попередня анонімізація персональних або чутливих даних, щоб зменшити ризики обробки конфіденційної інформації мовною моделлю. Після цього формується пошуковий запит до бази знань, за яким знаходяться релевантні фрагменти доменної інформації.

На наступному етапі формується prompt для LLM. Він містить інструкцію щодо ролі моделі, текст звернення, релевантний контекст із бази знань, очікувану схему відповіді та обмеження щодо використання фактів. Для задачі класифікації prompt вимагає повернути intent із переліку допустимих класів. Для задачі вилучення даних prompt задає структуру очікуваних полів, наприклад номер договору, дату, суму, адресу, номер тікета або перелік відсутніх реквізитів. Для задачі генерації відповіді prompt вимагає сформувати ділову відповідь німецькою мовою на основі наданого контексту.

Особливістю LLM+RAG-підходу є те, що одна й та сама мовна модель може виконувати кілька функцій: класифікацію intent, вилучення полів, узагальнення змісту листа та підготовку чернетки відповіді. Це робить підхід більш гнучким порівняно з Traditional-контуром. Наприклад, якщо звернення сформульоване непрямо або не містить типових ключових слів, LLM може визначити його зміст на основі семантики тексту. Якщо важлива інформація міститься у PDF-вкладенні, модель може врахувати її разом із текстом листа.

Однак така гнучкість супроводжується ризиками. Мовна модель може пропускати формалізовані реквізити, некоректно нормалізувати значення або генерувати зайві твердження, яких не було у вхідних даних чи базі знань. Тому в прототипі результат LLM-обробки розглядається не як безумовно правильне рішення, а як результат, що потребує подальшої перевірки. Для цього використовується верифікація за структурованими даними договору, перевірка наявності обов’язкових полів і контроль того, щоб у відповіді не з’являлися непідтверджені факти.

Після отримання відповіді від LLM система намагається привести результат до структурованого формату. Для цього очікується, що модель повертає intent, confidence, вилучені поля, missing fields і draft reply у визначеній структурі. Якщо відповідь не відповідає очікуваному формату або містить недостатньо даних, кейс може бути позначений як такий, що потребує ручної перевірки. Така логіка є важливою для страхового домену, де некоректна

автоматична відповідь може призвести до помилок у комунікації з клієнтом або страховиком.

Узагальнена логіка LLM+RAG-підходу подана на рисунку 3.6. Перевагою LLM+RAG-підходу є здатність працювати з більш гнучкими мовними формулюваннями, враховувати ширший контекст звернення та формувати більш природні текстові відповіді. Він є особливо корисним для перефразованих листів, складних звернень і випадків, де зміст залежить від інформації у вкладених документах.

Водночас цей підхід має істотні обмеження: вищу затримку обробки, більші вимоги до ресурсів, нижчу стабільність точного вилучення формалізованих полів і потребу в додатковому контролі результатів. Саме тому в межах експериментального дослідження LLM+RAG розглядається як альтернативний підхід для порівняння з Traditional, а не як безумовна заміна детермінованого контуру.

3.6 Практична реалізація прототипу

Практична реалізація системи виконана як багатокомпонентний програмний прототип із REST API, CLI-інтерфейсом, модульною структурою коду, SQLite-базою даних, локальним файловим сховищем, базою знань у вигляді markdown-документів, ML-моделлю та LLM-клієнтом. Згідно з документацією, API-частина реалізована на базі FastAPI і підтримує ендпоінти для завантаження кейсів, їх обробки, перегляду чернеток відповідей, ручної верифікації договорів, перегляду черги низької впевненості, переіндексації knowledge base та донавчання моделі [14–17]. Це свідчить, що система спроєктована не як ізольований алгоритм, а як повноцінний прикладний сервіс, придатний до інтеграції в операційний контур.

Рисунок 3.6 – Діаграма узагальненої логіки LLM+RAG підходу

Зберігання даних організовано у SQLite з WAL-режимом. У схемі бази даних передбачені таблиці для кейсів, договорів, клієнтів, документів, виправлень оператором, черги низької впевненості та метаданих вкладень. Така структура дає можливість не лише обробляти поточні звернення, а й накопичувати історію кейсів, результати класифікації, журнали pipeline та ground truth-виправлення для подальшого active learning. Вкладення фізично розміщуються в окремому файловому сховищі, а в БД зберігаються лише їх ключі та метадані [16]. Це рішення є практичним для on-premise розгортання і відповідає вимогам до контрольованого локального зберігання документів. ER-схема бази даних системи зображена на рисунку 3.7.

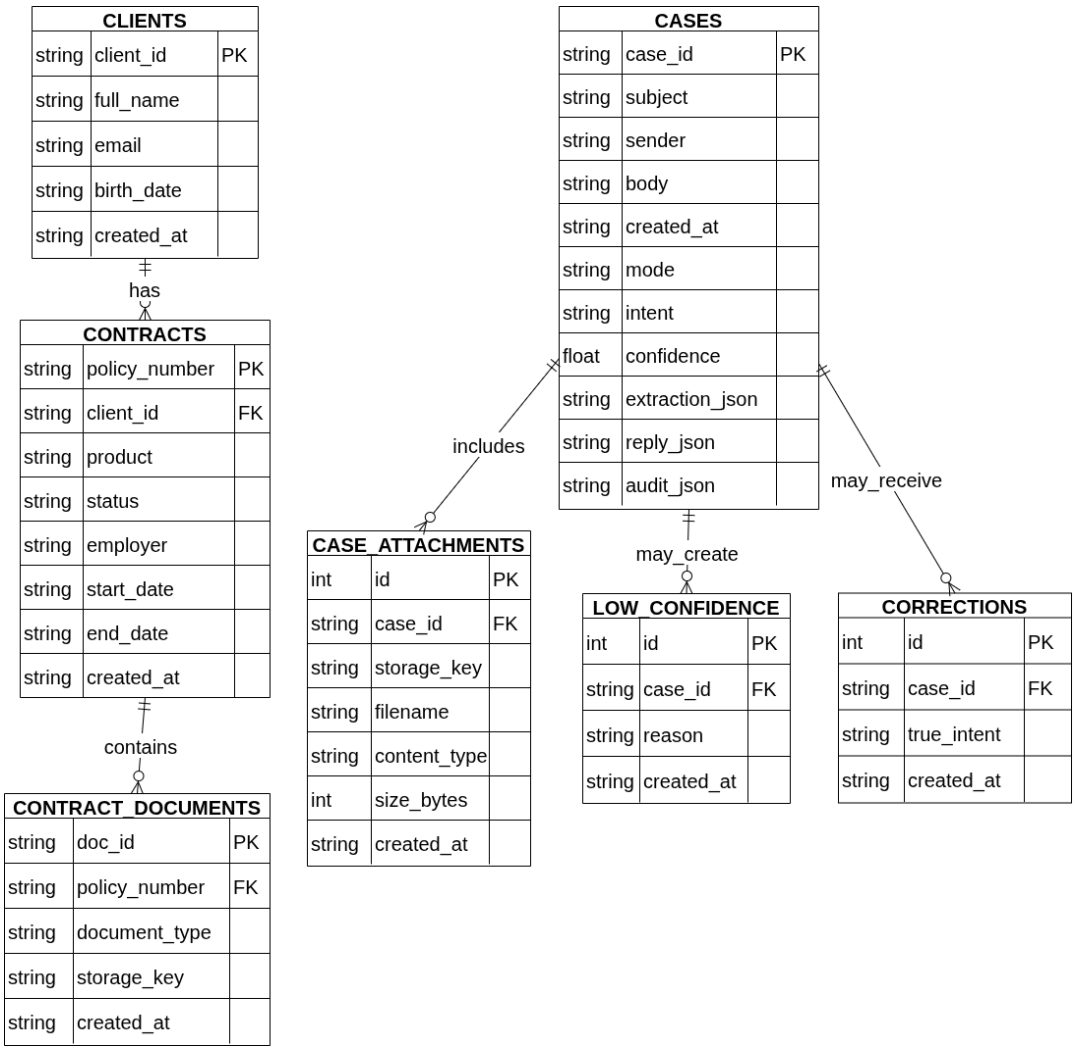

Рисунок 3.7 – ER-схема бази даних системи

На рівні інфраструктури передбачено як локальний запуск, так і контейнеризацію через Docker Compose. Окремо описано можливість використання локальної LLM через Ollama або тестового MockLLM для середовищ, де робота з реальною моделлю не потрібна або обмежена ресурсами.

Отже, практична реалізація прототипу підтверджує технічну здійсненність запропонованої архітектури та показує, що система може бути побудована як локальний сервіс із повним циклом обробки: від приймання листа до формування контрольованої чернетки відповіді й накопичення даних для подальшого вдосконалення моделі.

4 РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ ТА ЇХ АНАЛІЗ

4.1 Організація експериментального дослідження

Для оцінювання запропонованих підходів було проведено експериментальне порівняння двох базових стратегій обробки страхових звернень: детермінованого підходу Traditional (Rules + ML) та інтелектуального підходу LLM+RAG. Метою експерименту було визначити, який із підходів краще підходить для задачі автоматизованої обробки страхової кореспонденції з урахуванням точності класифікації, якості вилучення полів, здатності працювати з PDF-вкладеннями та часових характеристик обробки. Експериментальний набір містив 20 тестових кейсів, поділених на чотири групи складності. Група A включала прості листи з явними ключовими словами. Група B містила звернення середньої складності, де частина полів була відсутня або задавалася неявно. Група C включала перефразовані листи, у яких стандартні ключові слова були відсутні або замінені нетиповими формулюваннями. Група D складалася з кейсів, у яких суттєва інформація містилася у PDF-вкладеннях. Така структура набору дозволила оцінити системи не лише на тривіальних випадках, але й у більш складних сценаріях, наближених до практичних умов.

Для побудови тестового набору використовувалися типові доменні сценарії, характерні для страхової кореспонденції. Серед них були: зміна адреси клієнта, запит щодо виплати або дати настання виплати, автоматична відповідь страховика з номером тікета, запит щодо розміру внеску, звернення невизначеного типу, об’єднання договорів, питання щодо поновлення договору та повідомлення про додатковий внесок. В окремих кейсах очікуваними результатами були не лише тип intent, а й конкретні поля, наприклад policy_number, address тощо. Це забезпечило можливість оцінити не лише класифікацію, але й якість структуризації даних.

У межах оцінювання використовувалися такі показники: Intent Accuracy, Intent Macro F1, Intent Micro F1, Field Micro F1, PDF Field Hit, а також часові метрики latency p50, latency p90 і latency mean. Такий набір метрик є достатнім для оцінки системи з двох ключових точок зору: по-перше, наскільки правильно вона визначає тип звернення, а по-друге, наскільки коректно вилучає потрібні поля для подальшої автоматизації. Окремий інтерес становить метрика PDF Field Hit, оскільки в реальному страхуванні значна частина релевантних даних часто розміщується у вкладених документах, а не в тілі листа.

Під час аналізу результатів слід враховувати, що в матеріалах наявні кілька проміжних запусків із mock-LLM, які відображають етапи налагодження системи. Через це показники LLM+RAG у mock-режимі суттєво змінювалися – від досить високих до практично неприйнятних. Саме тому основний підсумковий висновок доцільно робити на основі фінального звіту з локальною моделлю через Ollama, а mock-запуски використовувати як підтвердження того, що LLM-контур є чутливим до конкретної реалізації та налаштувань.

4.2 Метрики оцінювання результатів експерименту

Для кількісного оцінювання результатів експериментального дослідження було використано набір метрик, що дозволяє проаналізувати якість роботи системи з кількох точок зору: правильність класифікації типу звернення, якість вилучення структурованих полів, здатність системи працювати з PDF-вкладеннями та часові характеристики обробки. Такий підхід є доцільним, оскільки задача автоматизованої обробки страхової кореспонденції не зводиться лише до визначення класу листа. Для практичного використання системи важливо не тільки правильно встановити intent звернення, але й коректно знайти реквізити договору, дату, суму, номер тікета, адресу або інші поля, необхідні для подальшої автоматизації бізнес-процесу.

Базовою метрикою для оцінювання класифікації intent була Accuracy, або частка правильно класифікованих звернень серед усіх тестових прикладів. Вона обчислюється за формулою:

$$Accuracy = \frac{N_{correct}}{N_{total}}, \quad (4.1)$$

де $N_{correct}$ – кількість звернень, для яких система правильно визначила intent;

N_{total} – загальна кількість звернень у тестовому наборі.

Метрика Accuracy є зрозумілою та зручною для загального порівняння підходів, однак вона не завжди достатньо повно відображає якість роботи класифікатора, особливо якщо класи представлені нерівномірно. Наприклад, якщо більшість листів належить до одного або кількох найпоширеніших типів звернень, система може мати високе значення Accuracy, але при цьому погано розпізнавати рідкісні класи. Саме тому додатково використовувалися метрики Precision, Recall та F1-score, які є стандартними для задач класифікації текстів і машинного навчання [18, 19].

Для кожного класу intent можна визначити такі величини:

– TP – кількість правильно визначених прикладів певного класу;

– FP – кількість прикладів, які були помилково віднесені до цього класу;

– FN – кількість прикладів цього класу, які система помилково віднесла до інших класів.

На основі цих величин обчислюється Precision згідно формули:

$$Precision = \frac{TP}{TP + FP}. \quad (4.2)$$

Precision показує, наскільки точними є позитивні передбачення системи. У контексті страхової кореспонденції ця метрика відповідає на питання: якщо

система визначила звернення як певний тип, наскільки часто це рішення було правильним. Наприклад, якщо лист був класифікований як звернення щодо зміни адреси, Precision показує, наскільки надійним є таке рішення.

Метрика Recall обчислюється за формулою:

$$Recall = \frac{TP}{TP + FN}. \quad (4.3)$$

Recall показує, яку частку всіх фактичних звернень певного класу система змогла знайти. У страховій системі це важливо, оскільки пропуск певного типу звернення може мати практичні наслідки. Наприклад, якщо система не розпізнає звернення щодо розірвання договору або зміни реквізитів, такий кейс може бути неправильно маршрутизований або потребуватиме додаткового ручного опрацювання.

Для узагальнення Precision і Recall використовується F1-score, який є гармонійним середнім між цими двома метриками за формулою:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}. \quad (4.4)$$

F1-score є особливо корисною метрикою тоді, коли потрібно враховувати як помилкові спрацьовування, так і пропущені приклади. У межах цього дослідження вона застосовувалася для оцінювання якості класифікації intent та якості вилучення структурованих полів. Згідно з документацією scikit-learn, F1-score інтерпретується як гармонійне середнє precision та recall, де найкраще значення дорівнює 1, а найгірше – 0.

Для багатокласової класифікації intent використовувалися два варіанти усереднення F1-score: Macro F1 та Micro F1. Метрика Macro F1 обчислюється як середнє значення F1-score для всіх класів за формулою:

$$MacroF1 = \frac{1}{K}\sum_{i=1}^{K} F1_i, \quad (4.5)$$

де K – кількість класів intent;

$F1_i$ – значення F1-score для і-го класу.

Macro F1 надає однакову вагу кожному класу незалежно від кількості прикладів у ньому. Тому ця метрика є корисною для аналізу того, наскільки рівномірно система працює з усіма типами звернень, включно з менш представленими класами. У страховій предметній області це важливо, оскільки рідкісні звернення також можуть бути юридично або фінансово значущими.

Метрика Micro F1 обчислюється шляхом агрегування всіх TP, FP та FN за всіма класами за формулами:

$$MicroPrecision = \frac{\Sigma TP}{\Sigma TP + \Sigma FP}, \quad (4.6)$$

$$MicroRecall = \frac{\Sigma TP}{\Sigma TP + \Sigma FN}, \quad (4.7)$$

$$MicroF1 = 2 * \frac{MicroPrecision * MicroRecall}{MicroPrecision + MicroRecall}. \quad (4.8)$$

Micro F1 більше відображає загальну якість роботи системи на всьому тестовому наборі. Якщо Macro F1 дозволяє оцінити баланс між класами, то Micro F1 показує інтегральну результативність системи з урахуванням усіх прикладів. Саме тому в експерименті використовувалися обидві метрики: Intent Macro F1 для аналізу збалансованості класифікації та Intent Micro F1 для оцінювання загальної точності розпізнавання intent.

Окремо оцінювалася якість вилучення структурованих полів. Для цього використовувалася метрика Field Micro F1, яка розраховується за тією ж логікою, що й Micro F1, але одиницею оцінювання є не клас листа, а окреме вилучене поле. До таких полів належать, наприклад, policy_number, address,

ticket_id, date, amount_eur, delta_eur та інші реквізити. Для кожного поля результат системи порівнювався з очікуваним еталонним значенням.

У задачі вилучення полів можна визначити:

– TP_{field} – поле було вилучене правильно;

– FP_{field} – система вилучила зайве або неправильне поле;

– FN_{field} – система не вилучила поле, яке мало бути знайдене.

Тоді Field Micro F1 обчислюється за формулою:

$$FieldMicroF1 = 2 * \frac{P_{field} * F_{field}}{P_{field} + F_{field}}, \quad (4.9)$$

де

$$P_{field} = \frac{\Sigma TP_{field}}{\Sigma TP_{field} + \Sigma FP_{field}},$$

$$F_{field} = \frac{\Sigma TP_{field}}{\Sigma TP_{field} + \Sigma FN_{field}}.$$

Ця метрика є критично важливою для досліджуваної системи, оскільки правильне визначення intent без коректного вилучення реквізитів не забезпечує повноцінної автоматизації. Наприклад, якщо система правильно визначила, що лист стосується зміни адреси, але не змогла вилучити нову адресу або номер договору, такий результат усе одно потребуватиме ручного втручання.

Для оцінювання роботи з вкладеними документами використовувалася спеціальна метрика PDF Field Hit. Вона показує, яку частку очікуваних полів, що містилися саме у PDF-вкладеннях, система змогла знайти. Метрика обчислюється за формулою:

$$PDFFieldHit = \frac{N_{pdf\_found}}{N_{pdf\_expected}}, \quad (4.10)$$

де N_{pdf_found} – кількість правильно знайдених полів із PDF-документів;

$N_{pdf\_expected}$ – загальна кількість очікуваних полів, які були присутні у PDF-вкладеннях.

Використання цієї метрики обумовлене специфікою страхової кореспонденції. У реальних бізнес-процесах важлива інформація часто міститься не в тілі електронного листа, а у прикріплених заявах, рахунках, довідках, договорах або інших документах. Тому система повинна оцінюватися не лише за здатністю аналізувати текст листа, але й за здатністю працювати з вкладеннями.

Крім якісних метрик, у дослідженні використовувалися часові характеристики обробки: Latency p50, Latency p90 та Latency Mean. Вони дозволяють оцінити практичну придатність системи до використання в реальному середовищі. Середня затримка обробки визначається за формулою:

$$LatencyMean = \frac{1}{N}\sum_{i=1}^{N} t_i, \tag{4.11}$$

де t_i – час обробки і-го звернення;

N – кількість тестових звернень.

Метрика Latency p50 відповідає медіанному часу обробки, тобто такому значенню затримки, нижче якого знаходиться 50% усіх результатів. Метрика Latency p90 показує значення, нижче якого знаходиться 90% усіх виміряних затримок. На відміну від середнього значення, p50 та p90 краще відображають поведінку системи у типових і гірших сценаріях. Це особливо важливо для порівняння Traditional-підходу та LLM+RAG, оскільки мовні моделі можуть мати значно більший і менш стабільний час відповіді.

У межах дослідження часові метрики мають не менш важливе значення, ніж показники точності. Для страхової інформаційної системи модель повинна

бути не лише якісною, але й достатньо швидкою для інтеграції у робочий процес. Якщо підхід демонструє високу якість класифікації, але потребує кількох хвилин на обробку одного звернення, його безпосереднє використання як основного production-механізму є обмеженим. Саме тому порівняння latency дозволяє обґрунтувати доцільність гібридної архітектури, де швидкий deterministic/ML-контур обробляє типові звернення, а LLM+RAG використовується лише для складних або неоднозначних випадків.

Отже, використаний набір метрик дозволяє комплексно оцінити результати експерименту. Intent Accuracy, Intent Macro F1 та Intent Micro F1 характеризують якість класифікації звернень; Field Micro F1 показує здатність системи вилучати структуровані реквізити; PDF Field Hit оцінює роботу з вкладеними документами; latency-метрики відображають практичну швидкодію системи. Разом ці показники дають змогу не лише порівняти Traditional та LLM+RAG, але й обґрунтувати вибір гібридної каскадної моделі як найбільш збалансованого рішення для автоматизованої обробки страхової кореспонденції.

4.3 Порівняльні результати Traditional та LLM+RAG

За результатами фінального експерименту з Ollama, які внесені в таблицю 4.1, встановлено, що підхід LLM+RAG перевершив Traditional за загальною точністю класифікації intent: значення Intent Accuracy зросло з 80% до 90%. Аналогічно Intent Micro F1 збільшився з 80% до 90%. Це свідчить про те, що мовна модель краще справляється із задачами розпізнавання наміру звернення, особливо у випадках, коли текст не містить очевидних ключових слів або коли значуща інформація знаходиться у вкладеннях. Водночас Intent Macro F1 для LLM+RAG був лише незначно нижчим за Traditional: 83.6% проти 84.9%, що також підтверджує конкурентоспроможність LLM-підходу на рівні класифікації.

Таблиця 4.1 – Порівняння результатів Traditional та LLM+RAG

Метрика	Traditional (Rules + ML)	LLM+RAG	Різниця
Intent Accuracy	80.0%	90.0%	+10.0%
Intent Macro F1	84.9%	83.6%	-1.3%
Intent Micro F1	80.0%	90.0%	+10.0%
Field Micro F1	96.0%	32.3%	-63.7%
PDF Field Hit	85.7%	28.6%	-57.1%
Latency p50	6 мс	228154 мс	+228148 мс
Latency p90	10 мс	298393 мс	+298383 мс
Latency Mean	8 мс	220829 мс	+220821 мс

Проте за якістю вилучення структурованих полів детермінований підхід виявився значно сильнішим. Для Traditional значення Field Micro F1 становило 96.0%, тоді як для LLM+RAG – лише 32.3%. Подібна ситуація спостерігається і для PDF Field Hit: Traditional досяг 85.7%, а LLM+RAG – лише 28.6%. Це означає, що хоча мовна модель часто правильно розпізнавала загальний намір листа, вона набагато гірше справлялася з надійним вилученням конкретних реквізитів, критично важливих для практичної обробки звернення. Для страхової сфери це є суттєвим обмеженням, оскільки без коректно знайдених номерів договорів, дат, сум, ідентифікаторів та інших ключових полів автоматичне подальше опрацювання втрачає практичну цінність.

Ще більш контрастними є часові характеристики. У фінальному звіті для Traditional середня затримка становила лише 8 мс, тоді як для LLM+RAG – близько 220829 мс, тобто понад 220 секунд. Медіанний час обробки для Traditional дорівнював 6 мс, а для LLM+RAG – 228154 мс. Таким чином, навіть якщо LLM-контур забезпечує виграш у гнучкості класифікації, він вимагає на порядки більше часу на виконання. Для production-використання у внутрішній страховій системі це означає, що застосування LLM до кожного звернення

недоцільне, особливо за умов локального розгортання на обмежених обчислювальних ресурсах.

Корисно також зіставити ці результати з проміжними mock-запусками. У більш ранньому mock-експерименті Traditional показував Intent Accuracy = 95% і Field Micro F1 = 96%, тоді як LLM+RAG мав дуже слабке вилучення полів, аж до 7.4% у першому запуску. У наступних mock-запусках результати LLM+RAG покращувалися, але залишалися нестабільними, що додатково вказує на високу залежність цього контуру від конкретної конфігурації, формулювання промптів і якості інтеграції. Це ще раз підкреслює перевагу deterministic/ML-рішень у задачах, де потрібна стабільність та відтворюваність.

Отже, результати порівняння показують, що жоден із двох базових підходів не є універсальним. Traditional краще підходить для надійного структурованого вилучення даних і швидкої обробки, тоді як LLM+RAG демонструє переваги в задачі класифікації складних або нетипових звернень.

4.4 Аналіз результатів за групами складності

Детальніший розгляд результатів за групами складності дозволяє глибше зрозуміти поведінку досліджуваних підходів. Згідно з таблицею 4.2, обидва підходи однаково добре працюють на простих і середніх кейсах, однак LLM+RAG показує перевагу на перефразованих зверненнях та PDF-орієнтованих сценаріях. Це підтверджує доцільність використання LLM як допоміжного інструмента для складних випадків.

Для групи A, до якої входили прості звернення з явними ключовими словами, обидва підходи показали однаковий результат – 100% точності. Це очікувано, оскільки такі кейси мають чітко виражені мовні ознаки, які добре виявляються як правилами, так і мовною моделлю. У подібних сценаріях використання LLM не дає суттєвого виграшу в якості, проте значно збільшує обчислювальні витрати.

Таблиця 4.2 – Точність класифікації за групами складності

Група	Характеристика кейсів	Traditional	LLM+RAG
A	Прості звернення з явними ключовими словами	100.0%	100.0%
B	Звернення середньої складності, частково неповні	100.0%	100.0%
C	Перефразовані звернення без типових keywords	40.0%	60.0%
D	Звернення, де значущі дані містяться у PDF	80.0%	100.0%

Для групи B, яка включала звернення середньої складності з частково відсутніми полями або контекстною інформацією, обидві системи також продемонстрували 100% за intent-класифікацією. Це свідчить про те, що для типових напівструктурованих страхових листів добре працюють як логіка на основі правил і ML, так і LLM-контур. Однак саме на цьому рівні стає важливою не лише правильна класифікація, а й коректне виявлення missing fields, що є сильнішою стороною детермінованого контуру.

Найбільш показовою є група C, яка містить перефразовані звернення без типових ключових слів. Тут Traditional показав лише 40%, а LLM+RAG – 60%. Це один із найважливіших результатів усього дослідження, оскільки він підтверджує, що мовна модель дійсно краще справляється із семантичним розумінням нетипових формулювань. У фінальному звіті наведено конкретні приклади, де Traditional повертав UNKNOWN, тоді як LLM+RAG правильно визначав теми листів: “звернення щодо виплати”, “звернення щодо зміни адреси” тощо. Водночас LLM не був безпомилковим: наприклад, один перефразований кейс щодо зміни внеску було помилково класифіковано як “звернення щодо розміру страхового внеску”, а нерелевантний лист – як “звернення щодо зміни адреси”. Це означає, що хоча LLM краще працює із

семантично складним текстом, ризик хибних позитивних спрацьовувань у нього також вищий.

Група D містила кейси, у яких ключова інформація була винесена у PDF-вкладення. У фінальному Ollama-запуску LLM+RAG показав тут 100%, тоді як Traditional – 80%. Це свідчить про значний потенціал LLM у випадках, коли потрібно інтерпретувати зміст вкладеного документа, а не лише шукати очевидні шаблони у тілі листа. Однак цей результат потрібно інтерпретувати обережно: при всій перевазі LLM за intent-класифікацією у PDF-сценаріях, загальна метрика PDF Field Hit для LLM+RAG все одно виявилася значно нижчою за Traditional. Тобто модель краще здогадувалася про тип кейсу, але не так надійно вилучала конкретні значення полів. Для страхових процесів це означає, що LLM придатний як інструмент semantic fallback, але не як єдиний механізм структуризації даних.

Таким чином, груповий аналіз підтверджує таку закономірність: для простих і середніх звернень LLM не має суттєвої переваги над Traditional, для перефразованих і PDF-орієнтованих кейсів він часто корисніший, але при цьому не забезпечує достатньої стабільності у вилученні полів. Саме ця картина логічно веде до використання каскадної гібридної моделі, де складні нетипові випадки передаються на LLM-рівень лише після того, як детермінований контур не зміг дати достатньо надійного результату.

4.5 Аналіз якості вилучення полів і генерації відповідей

Окремої уваги заслуговує оцінка якості вилучення структурованих полів. Таблиця 4.3 демонструє, що детермінований підхід значно краще справляється з вилученням формалізованих реквізитів, таких як номер договору, номер тікета, кілька номерів полісів або зміна суми внеску. Це пояснюється тим, що такі поля мають чіткі формати та надійніше виявляються правилами і регулярними виразами.

Таблиця 4.3 – Порівняння якості вилучення окремих полів

Поле	Traditional F1	LLM+RAG F1
date	100.0%	100.0%
delta_eur	100.0%	0.0%
employer_consent_required	50.0%	100.0%
policy_number	100.0%	11.1%
policy_numbers	100.0%	0.0%
ticket_id	100.0%	0.0%
Мікросереднє значення (Micro F1)	96.0%	32.3%

У фінальному звіті Traditional продемонстрував майже ідеальні результати для більшості ключових полів. Для date, delta_eur, policy_number, policy_numbers і ticket_id значення F1 дорівнювало 100%, а для employer_consent_required – 50%, що все одно суттєво перевищує відповідні результати LLM-контру. У LLM+RAG лише поле date та employer_consent_required показали високі результати, тоді як для policy_number F1 становив лише 11.1%, а для delta_eur, policy_numbers і ticket_id – 0%. Це свідчить, що мовна модель може бути корисною для інтерпретації ситуації, але значно гірше справляється з відтворюваним точним вилученням формалізованих реквізитів.

Такий результат добре узгоджується зі структурою самої задачі. Поля на кшталт VSNR, номерів тікетів, кількох номерів договорів або дельти зміни внеску мають чіткий формальний вигляд і тому природно краще вилучаються детермінованими правилами, регулярними виразами та перевірками формату. Натомість LLM може або пропускати такі поля, або некоректно їх нормалізувати, особливо якщо промпт орієнтований більше на загальне розуміння змісту, ніж на жорсткий extraction.

Аналіз прикладів відповідей також демонструє принципову різницю між підходами. У звітах видно, що Traditional формує короткі, контрольовані, стилістично прості, але доменно коректні відповіді. Вони мають шаблонний характер, проте не виходять за межі перевірених фактів. Натомість LLM+RAG у частині відповідей генерує більш розгорнуті та «людяні» тексти, але часто включає зайві формулювання, повтори або некоректно підміняє точку зору мовця. У прикладах із фінального Ollama-запуску модель іноді формує відповіді в стилі запиту на відсутні поля навіть тоді, коли частина даних уже була надана, а в кейсі з автоматичними відповідями страховика вона взагалі створює текст, який не відповідає логіці внутрішньої службової нотатки. Це ще раз показує, що в прикладній страховій системі вільну генерацію відповіді необхідно обмежувати шаблонами, verified facts та постперевіркою.

З практичної точки зору це означає, що найбільш доцільною є така організація системи, за якої вилучення формалізованих полів і базове формування чернетки відповіді виконуються детермінованим контуром, а LLM використовується лише для тих етапів, де справді потрібне семантичне узагальнення або стилістичне поліпшення тексту.

4.6 Обмеження експерименту та підсумкові висновки

Інтерпретуючи отримані результати, необхідно враховувати низку обмежень експериментального дослідження. По-перше, частина запусків виконувалася з mock-LLM, що не відображає реальної поведінки повноцінної мовної моделі й може давати як надто слабкі, так і надто штучно стабільні результати. По-друге, тестовий набір складався лише з 20 кейсів, тому він є достатнім для якісного порівняння та демонстрації поведінки системи, але не для статистично вичерпної оцінки на великій генеральній сукупності. По-третє, PDF-вкладення в експерименті є контрольованими й синтетичними, тоді як у реальній практиці документи можуть мати складнішу верстку, різну якість сканування та більше шуму. По-четверте, knowledge base не охоплює абсолютно

всі можливі інтенти та сценарії, що обмежує потенціал RAG-контру. Усе це прямо відзначено в експериментальних матеріалах і повинно бути враховано під час формулювання підсумкових висновків.

Попри ці обмеження, результати дають достатньо підстав для обґрунтованого висновку. Детермінований підхід Traditional (Rules + ML) є найкращим вибором для швидкої, стабільної та відтворюваної обробки типових страхових звернень, особливо якщо потрібне надійне вилучення структурованих полів. Він демонструє дуже високі значення Field Micro F1 та низькі затримки, що є критично важливим для production-використання. Натомість LLM+RAG показує свої переваги у випадках перефразованих листів і складних PDF-кейсів, де потрібне семантичне розуміння нестандартного тексту. Однак його слабкими сторонами залишаються повільність, нестабільність extraction та ризик менш контрольованої генерації відповіді.

Отже, результати дослідження підтверджують доцільність гібридної каскадної моделі, у якій швидкий і надійний rules+ML-контур використовується як основний, а LLM+RAG залучається лише у складних випадках, коли детермінований підхід не досягає достатнього рівня впевненості. Такий підхід дозволяє поєднати продуктивність, контрольованість і здатність системи працювати з нетиповими зверненнями, що й становить основний практичний результат проведеного дослідження.

5 ОБҐРУНТУВАННЯ ГІБРИДНОЇ МОДЕЛІ АВТОМАТИЗОВАНОЇ ОБРОБКИ СТРАХОВОЇ КОРЕСПОНДЕНЦІЇ

5.1 Передумови побудови гібридної моделі

Результати експериментального дослідження показали, що жоден із розглянутих базових підходів до автоматизованої обробки страхової кореспонденції не є універсальним для всіх типів звернень. Traditional-підхід, який поєднує правила, ML-класифікацію, регулярні вирази, NER-модуль, PDF-парсер і шаблонну генерацію відповіді, продемонстрував високу швидкодію, стабільність і якісне вилучення формалізованих полів. Водночас його слабкою стороною є обмежена здатність працювати з перефразованими, неповними або семантично складними зверненнями, у яких відсутні типові ключові слова чи стандартні доменні формулювання.

LLM+RAG-підхід, навпаки, показав кращу гнучкість у задачах розпізнавання змісту складних звернень, особливо у випадках, коли користувач формулює запит непрямо або коли частина важливої інформації міститься у PDF-вкладеннях. Велика мовна модель здатна враховувати ширший контекст, інтерпретувати нетипові формулювання та формувати більш природні текстові відповіді. Проте цей підхід має істотні практичні обмеження: високу затримку обробки, більші вимоги до обчислювальних ресурсів, нестабільність точного вилучення формалізованих реквізитів і ризик появи непідтверджених фактів у згенерованому тексті.

Отже, результати порівняння Traditional та LLM+RAG свідчать, що доцільним є не повне заміщення одного підходу іншим, а їх поєднання в межах єдиної гібридної моделі. Така модель повинна використовувати кожен метод лише на тому етапі, де він має найбільшу практичну перевагу. Для типових звернень з явними ознаками та формалізованими реквізитами доцільно застосовувати швидкий і контрольований Traditional-контур. Для складних,

перефразованих або неоднозначних звернень варто залучати LLM+RAG як допоміжний рівень семантичного аналізу.

Головна ідея гібридної моделі полягає в каскадній організації обробки. Система не звертається до ресурсоємного LLM-компонента для кожного листа, а спочатку використовує дешеві, швидкі й більш контрольовані механізми. Лише у випадку низької впевненості, нестачі критичних полів або виявлення складного сценарію звернення система переходить до LLM+RAG-рівня. Такий підхід дозволяє зберегти високу продуктивність для більшості типових кейсів і водночас підвищити якість обробки нестандартних повідомлень.

У межах даної роботи гібридна модель розглядається як результат експериментального аналізу двох базових підходів. Traditional та LLM+RAG були реалізовані й порівнянні експериментально, а гібридна модель формується на основі виявлених переваг і недоліків кожного з них. Тому її доцільно розглядати не як третій незалежний baseline, а як запропоноване підсумкове архітектурне рішення для подальшого розвитку системи автоматизованої обробки страхової кореспонденції.

5.2 Логіка каскадного прийняття рішення

Гібридна модель будується за принципом послідовного уточнення рішення. На кожному етапі система оцінює, чи достатньо отриманого результату для автоматичного завершення обробки, чи потрібно перейти до наступного, складнішого рівня аналізу. Основними критеріями переходу між рівнями є рівень впевненості класифікації, наявність обов’язкових полів, статус верифікації договору, складність звернення та ризик формування некоректної відповіді.

Першим рівнем обробки є rule-based-модуль. Він аналізує текст листа, тему повідомлення та, за потреби, текст із вкладень на наявність ключових слів, регулярних виразів і доменних шаблонів. Якщо правило спрацьовує однозначно, система отримує попередній intent, рівень упевненості та

пояснення рішення. Наприклад, якщо лист містить явні формулювання щодо зміни адреси або розірвання договору, такий випадок може бути швидко класифікований без залучення складніших моделей.

Другим рівнем є ML-класифікатор. Він використовується тоді, коли rule-based-рівень не дав достатньо впевненого результату або коли знайдені ознаки є суперечливими. ML-класифікатор працює з векторним представленням тексту та дозволяє враховувати ширший набір статистичних ознак, ніж набір явно заданих правил. Його використання особливо доцільне для звернень, які не повністю відповідають шаблонам, але все ще належать до типових доменних сценаріїв.

Третім рівнем є LLM+RAG-контур. Він застосовується лише тоді, коли попередні рівні не забезпечили достатньо надійного результату або коли звернення має ознаки складного семантичного випадку. До таких випадків належать перефразовані листи, звернення з кількома можливими intent, повідомлення з мінімальним текстом і значущими PDF-вкладеннями, а також ситуації, де потрібно узагальнити контекст із бази знань. Узагальнено каскадна логіка описана на рисунку 5.1.

Формально рішення про завершення обробки після rule-based-рівня можна подати за формулою:

$$C_{rules} \geq T_{rules} \wedge M = \emptyset, \tag{5.1}$$

де C_{rules} – рівень впевненості rule-based-класифікації;

T_{rules} – порогове значення впевненості для прийняття рішення;

M – множина відсутніх обов’язкових полів.

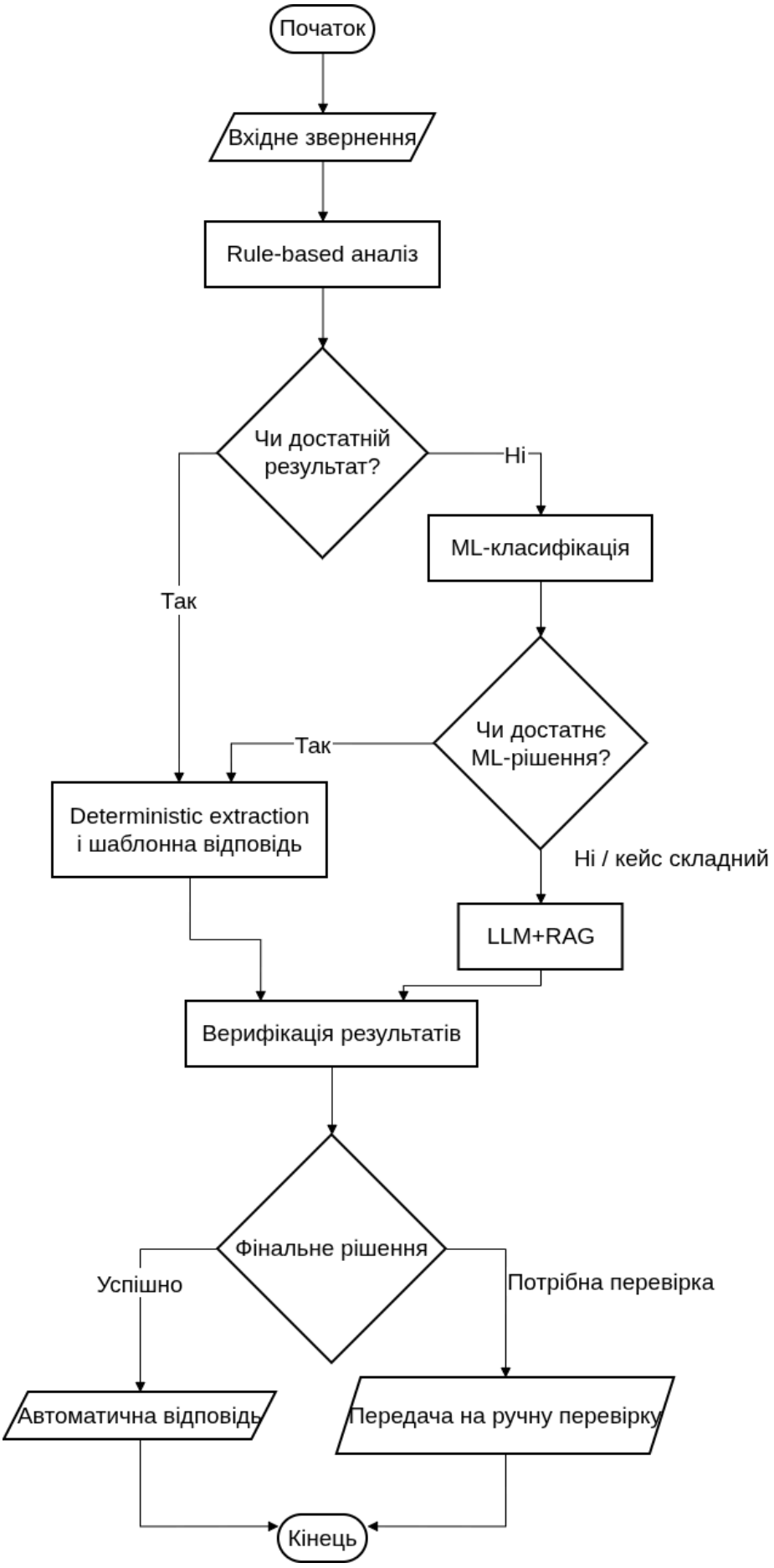

Рисунок 5.1 – Каскадна логіка обробки звернення гібридним підходом

Якщо ця умова виконується, система може завершити обробку без переходу до ML або LLM. Якщо ж рівень впевненості нижчий за поріг або множина відсутніх полів не є порожньою, система переходить до наступного рівня.

Для ML-рівня аналогічна умова виглядає як на формулі:

$$C_{ml} \geq T_{ml} \wedge M_{critical} = \emptyset, \tag{5.2}$$

де C_{ml} – рівень впевненості ML-класифікатора;

T_{ml} – поріг прийняття ML-рішення;

$M_{critical}$ – множина критичних відсутніх полів.

Якщо після ML-класифікації система все ще не має достатньо надійного результату, активується LLM+RAG-рівень згідно формули:

$$C_{ml} < T_{ml} \vee M_{critical} = \emptyset \vee S_{complex} = 1, \tag{5.3}$$

де $S_{complex}$ – ознака складного кейсу. Вона може набувати значення 1, якщо звернення є перефразованим, містить неоднозначний intent, залежить від PDF-вкладення або не може бути оброблене стандартним deterministic-контуром.

Після отримання результату від LLM+RAG система не приймає його автоматично без перевірки. Результат проходить верифікацію за структурованими даними, перевірку обов’язкових полів і контроль наявності непідтверджених фактів. Якщо хоча б одна з критичних перевірок не пройдена, кейс передається на ручну перевірку.

Таким чином, каскадна логіка дозволяє мінімізувати використання LLM, зменшити середній час обробки, зберегти контрольованість рішень і водночас забезпечити можливість обробки складних звернень, з якими Traditional-підхід справляється гірше [20].

5.3 Проєктування гібридного підходу обробки звернень

Гібридний підхід обробки звернень доцільно реалізувати як окремий orchestration-рівень, який координує роботу всіх компонентів системи. Такий компонент можна умовно назвати Hybrid Orchestrator. Його завдання полягає не в тому, щоб самостійно виконувати класифікацію або вилучення даних, а в тому, щоб приймати рішення, який саме інструмент потрібно використати на конкретному етапі обробки.

Основні компоненти гібридного підходу зображені на рисунку 5.2. Hybrid Orchestrator отримує на вхід кейс, який містить тему листа, текст повідомлення, метадані відправника та інформацію про вкладення. Далі він послідовно запускає компоненти обробки відповідно до каскадної логіки. Спочатку виконується rule-based-класифікація. Якщо її результат є достатньо впевненим, система переходить до deterministic extraction. Якщо результат недостатній, Hybrid Orchestrator звертається до ML-класифікатора. Якщо і ML-рівень не дає надійного рішення, запускається RAG-пошук і LLM-аналіз.

Окреме місце в архітектурі займає Confidence Evaluator. Цей компонент оцінює не лише числову впевненість класифікатора, але й загальну якість поточного результату. До його критеріїв можуть належати:

– рівень упевненості intent-класифікації;

– наявність обов’язкових полів для відповідного intent;

– кількість суперечливих ознак;

– факт наявності або відсутності PDF-вкладень;

– результат перевірки договору;

– рівень ризику конкретного типу звернення;

– наявність персональних або фінансових даних;

– потреба в ручному підтвердженні.

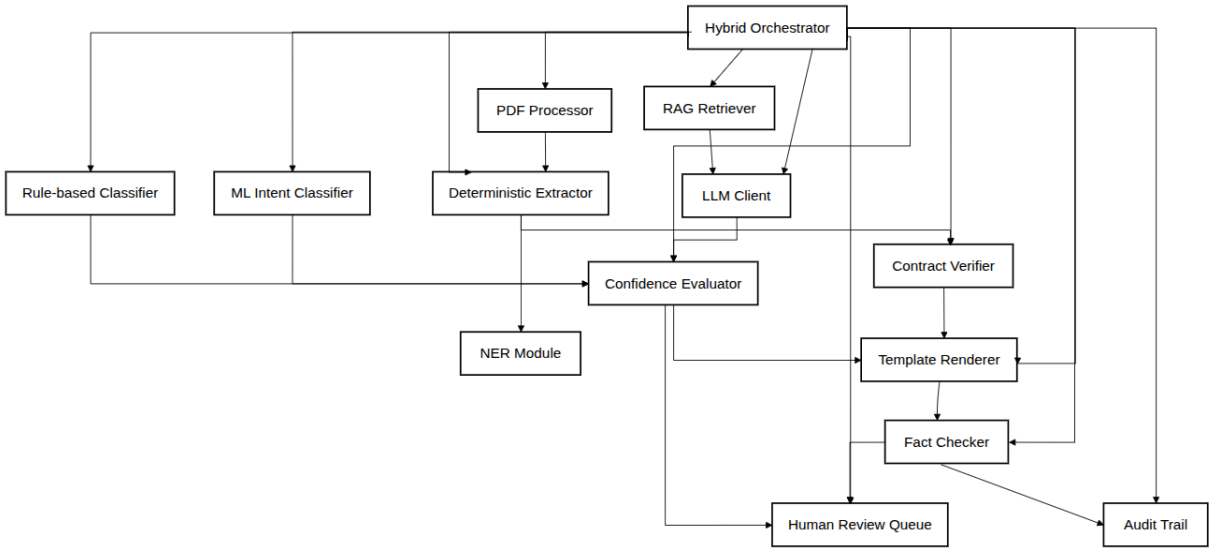

Рисунок 5.2 – Основні компоненти гібридного підходу

Наприклад, для простого інформаційного запиту можна встановити нижчий поріг автоматичної обробки, тоді як для фінансових або юридично значущих дій поріг має бути вищим. Це важливо для страхової предметної області, оскільки різні типи звернень мають різний рівень операційного та юридичного ризику.

У гібридному підході deterministic extraction залишається основним способом вилучення формалізованих реквізитів. Це пов’язано з тим, що експеримент показав значно кращу якість Traditional-підходу саме у вилученні структурованих полів. Тому навіть якщо intent був уточнений за допомогою LLM, вилучення номерів договорів, дат, сум, номерів тікетів та IBAN доцільно виконувати через регулярні вирази, NER і спеціалізовані парсери. LLM може використовуватися лише як допоміжний механізм для заповнення пропущених полів або інтерпретації складного текстового контексту.

RAG Retriever у гібридній моделі використовується не для кожного звернення, а тільки в тих випадках, коли потрібна додаткова доменна

інформація. Наприклад, якщо лист стосується поновлення договору, розірвання договору або зміни умов внеску, система може знайти відповідний документ у базі знань і передати його фрагмент мовній моделі. Це дозволяє обмежити генерацію відповіді релевантним контекстом і не покладатися виключно на загальні знання LLM.

Генерація відповіді в гібридній моделі має залишатися переважно шаблонною. Основна відповідь формується Template Renderer на основі intent, вилучених полів і результатів верифікації. LLM може використовуватися для стилістичного поліпшення тексту, формування короткого резюме або підготовки уточнювального запиту, але не повинна додавати нові факти без підтвердження з бази даних і бази знань. Такий підхід дозволяє зберегти природність відповіді, але не втратити контрольованість. Узагальнена схема гібридного підходу описана на рисунку 5.3.

Таким чином, гібридний підхід не є простим механічним об’єднанням Traditional та LLM+RAG. Він є керованою каскадною системою, у якій кожен компонент використовується відповідно до його сильних сторін [21].

5.4 Механізми верифікації, контролю впевненості та HITL

Для страхової інформаційної системи недостатньо лише правильно класифікувати звернення або згенерувати граматично коректну відповідь. Результат автоматичної обробки повинен бути перевіреним, пояснюваним і контрольованим. Це особливо важливо у страховій сфері, де навіть незначна помилка у визначенні типу звернення, трактуванні статусу договору або формулюванні відповіді може призвести до затримки обслуговування, фінансових ризиків, порушення внутрішніх регламентів або некоректної комунікації з клієнтом чи страховиком. Саме тому гібридна модель повинна містити спеціальні механізми верифікації, контролю впевненості та передачі складних випадків на ручне опрацювання.

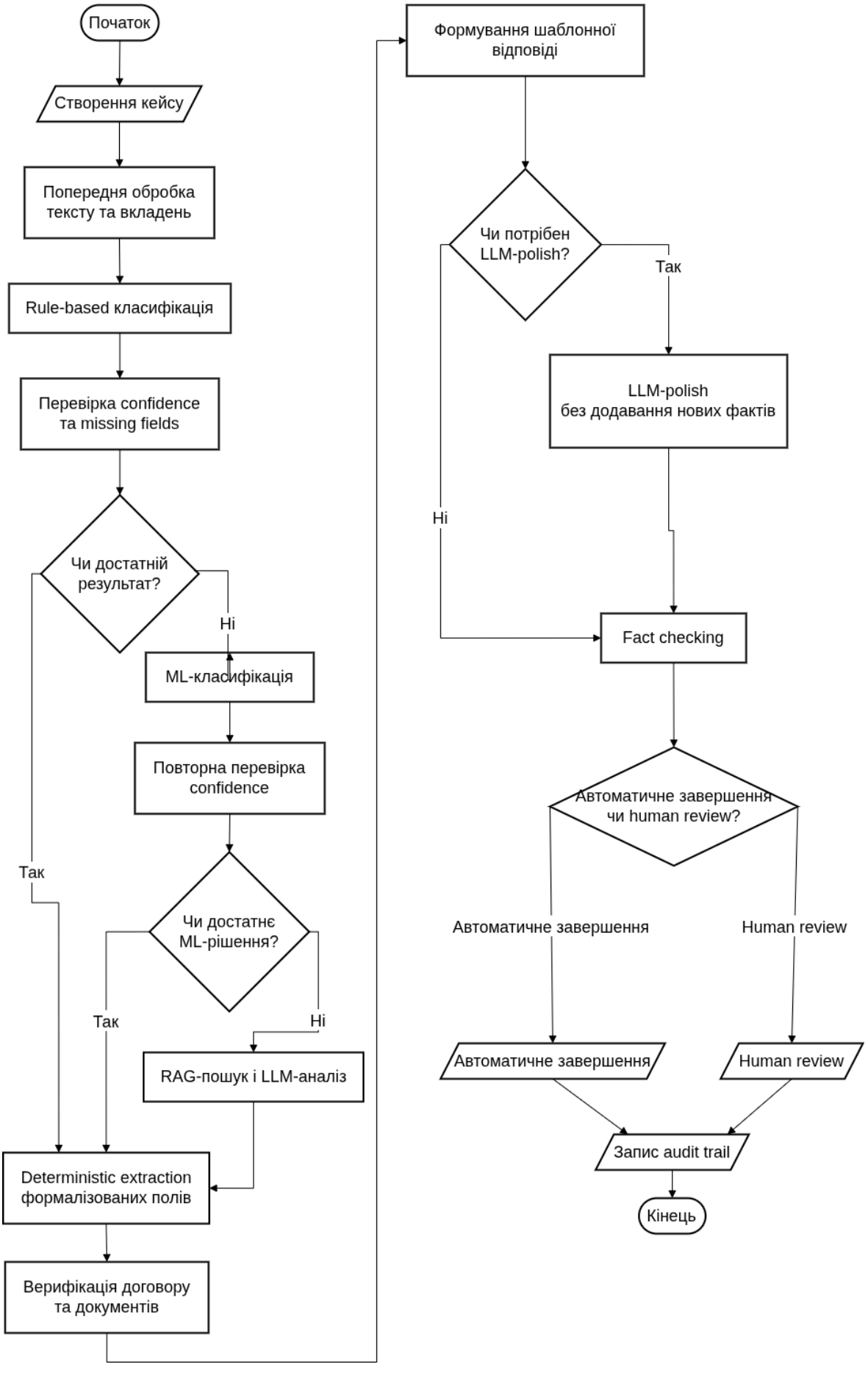

Рисунок 5.3 – Схема гібридного підходу

Першим механізмом контролю є перевірка рівня впевненості. Для кожного класифікаційного рівня задається власне порогове значення. Наприклад, rule-based-рівень може приймати рішення лише тоді, коли спрацювало однозначне правило або комбінація правил. ML-рівень може використовувати ймовірність класу або інший confidence score. LLM-рівень повинен повертати не тільки intent, але й пояснення та оцінку впевненості, хоча така оцінка має розглядатися обережно, оскільки мовні моделі не завжди коректно калібрують власну впевненість.

Другим механізмом є перевірка обов’язкових полів. Якщо частина обов’язкових полів відсутня, система не повинна створювати фінальну відповідь так, ніби вся інформація наявна. У такому випадку можливі два сценарії: або формується уточнювальний запит до клієнта, або кейс передається оператору.

Третім механізмом є верифікація за договірною базою. Якщо система вилучила номер договору, вона повинна перевірити, чи існує такий договір, який його статус, чи пов’язаний він із відповідним клієнтом, чи наявні необхідні документи. Це особливо важливо для запитів, пов’язаних зі змінами договору, виплатами, розірванням або фінансовими операціями. Відповідь клієнту має спиратися не лише на текст листа, а й на перевірені факти з інформаційної системи.

Четвертим механізмом є перевірка фактів. Його суть полягає в тому, що система формує перелік дозволених фактів, які можуть бути використані у відповіді. До такого переліку можуть входити:

– намір звернення;

– вилучені поля;

– статус договору;

– ім’я або роль клієнта;

– факт наявності документа;

– факт відсутності документа;

– релевантні правила з бази знань;

– перелік пропущених полів.

Якщо LLM-polish або LLM-generation додає інформацію, якої немає в цьому списку, відповідь повинна бути відхилена або передана на ручну перевірку. Це дозволяє зменшити ризик галюцинації, тобто генерації непідтверджених фактів.

П’ятим механізмом є human-in-the-loop. Він означає, що система не намагається автоматизувати всі звернення без винятку. Навпаки, складні, ризикові або невизначені випадки мають передаватися людині. До таких випадків належать:

– низька впевненість класифікації;

– суперечність між rules, ML та LLM;

– відсутність критичних полів;

– невідповідність номера договору даним у базі;

– юридично або фінансово значущий запит;

– підозра на некоректну LLM-відповідь;

– наявність чутливих персональних даних;

– нестандартне звернення, для якого немає шаблону.

Human-in-the-loop не є недоліком системи. Навпаки, для страхової сфери це необхідний елемент безпечної автоматизації. Мета системи полягає не в повному усуненні людини з процесу, а в тому, щоб автоматично обробляти типові випадки, підготувати структуровану інформацію для складних кейсів і зменшити навантаження на співробітників.

Окремо важливим є audit trail. Для кожного кейсу система повинна зберігати історію прийняття рішення: які правила спрацювали, який intent визначив ML-класифікатор, чи використовувався LLM, які документи були знайдені в базі знань, які поля вилучено, які поля відсутні, чому кейс був автоматично завершений або переданий на ручну перевірку. Це забезпечує пояснюваність і можливість подальшого аналізу помилок.

Загалом механізми верифікації та контролю дозволяють зробити гібридну модель придатною для практичного використання в страховому домені. Вони зменшують ризики, пов’язані з автоматичною генерацією відповідей, і забезпечують баланс між автоматизацією та надійністю [22, 23].

5.5 Очікувані переваги, обмеження та напрями подальшого розвитку

Запропонована гібридна модель має низку очікуваних переваг порівняно з використанням Traditional або LLM+RAG у чистому вигляді. Її головна перевага полягає в раціональному розподілі задач між різними методами. Типові, добре формалізовані звернення обробляються швидким deterministic/ML-контуром, тоді як складні або неоднозначні випадки передаються на LLM+RAG-рівень. Це дозволяє уникнути надмірного використання ресурсоємної мовної моделі й водночас зберегти можливість семантичного аналізу складних повідомлень.

Очікується, що така модель забезпечить нижчу середню затримку обробки порівняно з повним LLM+RAG-підходом, оскільки більшість типових звернень не потребуватиме запуску LLM. Одночасно вона повинна мати кращу якість обробки складних кейсів порівняно з чистим Traditional-підходом, оскільки LLM+RAG буде залучатися саме там, де правила та ML-класифікатор мають обмеження.

Другою перевагою є підвищення контрольованості. У гібридній моделі LLM не приймає всі рішення самостійно, а працює в межах заданого pipeline, з обмеженням через базу знань, перевірені факти, required fields, верифікацію договору та post-processing. Це робить систему більш придатною для використання в галузі, де важливими є точність, конфіденційність і юридична безпечність відповідей.

Третьою перевагою є модульність. Оскільки модель побудована на основі окремих компонентів, кожен із них може бути вдосконалений незалежно. Наприклад, можна покращити ML-класифікатор, розширити набір правил,

додати нові регулярні вирази для вилучення полів, покращити PDF-парсер, розширити базу знань або замінити локальну мовну модель на продуктивнішу.

Водночас гібридна модель має й обмеження. По-перше, її повноцінна реалізація потребує окремого налаштування порогів T_{rules}, T_{ml} та критеріїв складності $S_{complex}$. Якщо пороги будуть занадто низькими, система може автоматично приймати недостатньо надійні рішення. Якщо вони будуть занадто високими, надто багато кейсів передаватиметься на LLM або ручну перевірку, що знизить ефективність автоматизації.

По-друге, гібридна модель потребує якісної бази знань. Якщо документи в базі знань неповні, застарілі або погано структуровані, RAG-компонент може повертати нерелевантний контекст, що негативно впливатиме на якість LLM-відповіді. Тому підтримка бази знань має бути окремим процесом, пов’язаним із оновленням правил, регламентів, шаблонів і доменних інструкцій.

По-третє, необхідне подальше експериментальне оцінювання вже самої гібридної моделі. У межах цієї роботи експериментально порівнювалися два базові підходи – Traditional та LLM+RAG. На основі їх результатів було обґрунтовано доцільність гібридної каскадної моделі. Проте для повної кількісної оцінки гібридного підходу необхідно реалізувати його як окремий підхід, запустити на розширеному тестовому наборі та порівняти з базовими підходами за тими самими метриками: Intent Accuracy, Intent Macro F1, Intent Micro F1, Field Micro F1, PDF Field Hit, latency p50, latency p90 та latency mean.

Перспективними напрямами подальшого розвитку є:

– реалізація гібридного підходу як окремого production pipeline;

– розширення тестового набору реалістичними страховими зверненнями;

– налаштування confidence thresholds для різних intent;

– окреме оцінювання якості human-in-the-loop сценаріїв;

– покращення PDF extraction для складних документів і сканів;

– розширення бази знань для RAG;

– впровадження механізмів автоматичного виявлення hallucination;

– оцінювання якості відповідей експертами страхової предметної області;

– порівняння локальних LLM різного розміру з урахуванням ресурсних обмежень;

– дослідження можливості active learning на основі виправлень операторів.

Таким чином, запропонована гібридна модель є логічним підсумком проведеного дослідження. Вона враховує сильні сторони Traditional-підходу, а саме швидкість, стабільність і якісне вилучення формалізованих полів та переваги LLM+RAG у роботі зі складними, перефразованими й контекстно залежними зверненнями. При цьому модель передбачає механізми контролю, верифікації та передачі ризикових випадків людині, що є критично важливим для страхової галузі.

Запропонований підхід може бути використаний як основа для подальшої реалізації production-ready системи автоматизованої обробки страхової кореспонденції. Його практична цінність полягає в тому, що він не намагається повністю замінити людину або повністю покластися на LLM, а формує контрольовану багаторівневу систему підтримки обробки звернень, у якій автоматизація застосовується там, де вона є надійною, а складні випадки залишаються під контролем фахівця.

ВИСНОВКИ

У кваліфікаційній роботі було досліджено методи автоматизованої обробки звернень та генерації відповідей у страхових інформаційних системах на основі аналізу даних клієнта. У процесі виконання роботи встановлено, що страхова кореспонденція є складним об’єктом автоматизації, оскільки поєднує неструктурований або напівструктурований текст, доменно-специфічну термінологію, вкладені документи, варіативність мовних формулювань, наявність персональних і фінансових даних, а також високі вимоги до точності, конфіденційності, пояснюваності та контрольованості результатів обробки.

У першій частині роботи було проаналізовано предметну область страхових інформаційних систем і визначено основні особливості страхової кореспонденції як джерела даних для автоматизованої обробки. Встановлено, що значна частина важливої інформації надходить через електронну пошту, вебформи та PDF-вкладення, а тому не може ефективно оброблятися лише класичними транзакційними механізмами. Це обумовлює потребу у використанні методів класифікації текстів, вилучення структурованих полів, аналізу вкладених документів і формування відповідей на основі перевірених даних.

У роботі було досліджено традиційні підходи до обробки звернень, rule-based методи, методи машинного навчання, NLP, NER, LLM та RAG. Порівняльний аналіз показав, що кожен із цих підходів має власну область ефективного застосування. Rule-based і deterministic-підходи забезпечують високу швидкодію, пояснюваність і контрольованість, однак мають обмеження при роботі з нетиповими або перефразованими зверненнями. Методи машинного навчання дозволяють краще узагальнювати мовні ознаки на нові приклади, проте потребують навчальних даних і контролю якості. LLM+RAG-підхід є більш гнучким у семантичному аналізі складних звернень, але має вищі вимоги до ресурсів, більшу затримку обробки та ризик генерації непідтверджених фактів.

У практичній частині було спроєктовано архітектуру системи автоматизованої обробки страхової кореспонденції, побудовану за принципами Ports & Adapters, або гексагональної архітектури. Запропонована архітектура передбачає розділення доменної логіки, прикладних сценаріїв і технічних адаптерів, що забезпечує модульність, тестованість і можливість подальшого розвитку системи. До складу системи включено модулі приймання кейсів, класифікації звернень, вилучення структурованих полів, обробки PDF-вкладень, верифікації договорів і документів, пошуку в базі знань, генерації чернеток відповідей, контролю фактів, журналювання та передачі складних випадків на ручну перевірку.

У межах реалізації прототипу було розглянуто два базові підходи до обробки страхової кореспонденції: Traditional та LLM+RAG. Traditional-підхід реалізовано як детермінований контур, що поєднує rule-based класифікацію, ML-класифікатор, регулярні вирази, NER, PDF-парсер, перевірку даних за договірною базою та шаблонну генерацію відповіді. LLM+RAG-підхід реалізовано як альтернативний інтелектуальний контур, який використовує базу знань, retrieval-augmented generation, prompt-інструкції, LLM-класифікацію, вилучення даних і генерацію відповіді з подальшою перевіркою результатів. Практична реалізація підтвердила технічну можливість побудови локального on-premise рішення для обробки страхової кореспонденції без обов’язкової передачі даних до зовнішніх сервісів.

Для оцінювання підходів було проведено експериментальне дослідження на тестовому наборі з 20 страхових кейсів, поділених на групи складності. У межах експерименту використовувалися метрики Intent Accuracy, Intent Macro F1, Intent Micro F1, Field Micro F1, PDF Field Hit, а також latency p50, latency p90 і latency mean. Такий набір метрик дозволив оцінити не лише правильність класифікації intent, але й якість вилучення структурованих реквізитів, здатність працювати з PDF-вкладеннями та практичну швидкодію системи.

За результатами експериментального порівняння встановлено, що Traditional-підхід має значну перевагу в задачі точного вилучення

структурованих полів і за часовими характеристиками. Його значення Field Micro F1 становило 96,0%, PDF Field Hit – 85,7%, а середня затримка обробки – близько 8 мс. Це свідчить про високу придатність детермінованого контуру для типових страхових звернень, у яких важливі реквізити мають формалізований вигляд і можуть бути надійно вилучені за допомогою правил, регулярних виразів та NER.

Водночас LLM+RAG-підхід показав вищу загальну точність класифікації intent: Intent Accuracy становила 90,0% проти 80,0% у Traditional-підходу, а Intent Micro F1 також досяг 90,0%. Особливо помітною була перевага LLM+RAG у групах перефразованих звернень і кейсів, де значуща інформація містилася у PDF-вкладеннях. Це підтверджує, що великі мовні моделі мають потенціал для семантичного аналізу складних і нетипових повідомлень. Проте LLM+RAG суттєво поступився Traditional-підходу за якістю вилучення формалізованих полів: Field Micro F1 становив лише 32,3%, а PDF Field Hit – 28,6%. Крім того, середня затримка обробки перевищувала 220 секунд, що істотно обмежує можливість використання LLM+RAG як єдиного основного production-механізму.

На основі отриманих результатів було обґрунтовано гібридну каскадну модель автоматизованої обробки страхової кореспонденції. Її сутність полягає в тому, що типові та добре формалізовані звернення мають оброблятися швидким і контрольованим Traditional-контуром, тоді як LLM+RAG доцільно залучати лише у випадках низької впевненості, складних формулювань, неоднозначного intent або залежності від змісту вкладених документів. Така модель дозволяє раціонально розподілити задачі між різними методами та використати їхні сильні сторони без надмірного збільшення затримки обробки й операційних ризиків.

У роботі було запропоновано логіку каскадного прийняття рішення, що передбачає послідовне використання rule-based аналізу, ML-класифікації, deterministic extraction, RAG-пошуку, LLM-аналізу, верифікації результатів, fact checking та human-in-the-loop. Особливу увагу приділено механізмам контролю:

перевірці рівня впевненості, перевірці обов’язкових полів, верифікації договору й документів, контролю непідтверджених фактів, журналюванню audit trail і передачі ризикових кейсів на ручну перевірку. Це дозволяє розглядати запропонований підхід не як неконтрольовану автоматичну генерацію відповідей, а як багаторівневу систему підтримки прийняття рішень.

Практичне значення отриманих результатів полягає в можливості використання запропонованої архітектури та гібридної моделі для створення або вдосконалення внутрішніх систем автоматизованої обробки страхової кореспонденції. Така система може зменшити навантаження на співробітників, скоротити час первинної обробки типових звернень, підвищити якість структуризації даних, забезпечити стандартизацію відповідей і покращити контроль за складними або ризиковими випадками. Водночас модель не передбачає повного усунення людини з процесу, а орієнтована на поєднання автоматизації з експертною перевіркою там, де це необхідно.

Основними обмеженнями проведеного дослідження є невеликий обсяг тестового набору, синтетичний характер частини PDF-вкладень, залежність результатів LLM+RAG від конкретної локальної мовної моделі, обмеженість бази знань і відсутність повного окремого експериментального оцінювання гібридного підходу як самостійного production pipeline. Проте отримані результати є достатніми для обґрунтування доцільності гібридної каскадної моделі та визначення напрямів її подальшого розвитку.

Подальший розвиток роботи доцільно спрямувати на реалізацію гібридного підходу як окремого повноцінного pipeline, розширення тестового набору реальними страховими зверненнями, налаштування порогів упевненості для різних intent, покращення PDF extraction для складних і сканованих документів, розширення бази знань для RAG, оцінювання якості відповідей експертами страхової предметної області, а також використання active learning на основі виправлень операторів. Це дозволить підвищити якість системи та наблизити її до практичного впровадження в реальному страховому середовищі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Телюков Д. С. Hybrid orchestration of models for automated insurance correspondence processing in information systems. 30-й Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті»: зб. матеріалів форуму. Т. 6. Конференція «Інформаційні інтелектуальні системи» Харків, 2026.

2. Телюков Д. С., Імангулова З.А. Гібридний спосіб автоматизованої обробки страхової кореспонденції в інформаційних системах // Grail of Science. 2026. № 67. Матеріали VI International Scientific and Practical Conference “Open Science Nowadays: Main Mission, Trends and Instruments, Path and its Development”, 01 травня 2026 р., Вінниця, Україна – Відень, Австрія. P. 712–719. DOI: 10.36074/grail-of-science.01.05.2026.

3. EIOPA. Consumer Trends Report 2024. Frankfurt am Main: European Insurance and Occupational Pensions Authority, 2025. URL: https://www.eiopa.europa.eu/publications/consumer-trends-report-2024_en (accessed: 29.03.2026).

4. Davenport T. H., Ronanki R. Artificial Intelligence for the Real World // Harvard Business Review. 2018. January–February. URL: https://hbr.org/2018/01/artificial-intelligence-for-the-real-world (accessed: 29.03.2026).

5. Jurafsky D., Martin J. H. Speech and Language Processing. 3rd ed. draft. Stanford University, 2026. URL: https://web.stanford.edu/~jurafsky/slp3/ (accessed: 29.03.2026).

6. scikit-learn developers. Feature Extraction // scikit-learn User Guide. URL: https://scikit-learn.org/stable/modules/feature_extraction.html (accessed: 29.03.2026).

7. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł., Polosukhin I. Attention Is All You Need // Advances in Neural Information Processing Systems. 2017. Vol. 30. P. 5998–6008.

8. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation) // EUR-Lex. URL: https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng (accessed: 29.03.2026).

9. Russell S., Norvig P. Artificial Intelligence: A Modern Approach. 4th ed. Harlow: Pearson, 2021. 1166 p.

10. Cockburn A. Hexagonal Architecture // Alistair Cockburn. 2005. URL: https://alistair.cockburn.us/hexagonal-architecture/ (accessed: 29.03.2026).

11. Lewis P., Perez E., Piktus A., Petroni F., Karpukhin V., Goyal N., Küttler H., Lewis M., Yih W.-t., Rocktäschel T., Riedel S., Kiela D. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks // Advances in Neural Information Processing Systems. 2020. Vol. 33. P. 9459–9474. URL: https://proceedings.neurips.cc/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf (accessed: 29.03.2026).

12. Manning C. D., Raghavan P., Schütze H. Introduction to Information Retrieval. Cambridge: Cambridge University Press, 2008. 506 p.

13. Robertson S., Zaragoza H. The Probabilistic Relevance Framework: BM25 and Beyond // Foundations and Trends in Information Retrieval. 2009. Vol. 3, no. 4. P. 333–389.

14. spaCy. Models & Languages. URL: https://spacy.io/usage/models (accessed: 29.03.2026).

15. pdfminer.six documentation. URL: https://pdfminersix.readthedocs.io/ (accessed: 29.03.2026).

16. FastAPI. FastAPI Documentation. URL: https://fastapi.tiangolo.com/ (accessed: 29.03.2026).

17. Kleppmann M. Designing Data-Intensive Applications. Sebastopol: O’Reilly Media, 2017. 616 p.

18. Sokolova M., Lapalme G. A systematic analysis of performance measures for classification tasks // Information Processing & Management. 2009. Vol. 45, no. 4. P. 427–437. DOI: https://doi.org/10.1016/j.ipm.2009.03.002.

19. Powers D. M. W. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation // Journal of Machine Learning Technologies. 2011. Vol. 2, no. 1. P. 37–63.

20. Holzinger A. Interactive Machine Learning for Health Informatics: When Do We Need the Human-in-the-Loop? // Brain Informatics. 2016. Vol. 3, no. 2. P. 119–131. DOI: https://doi.org/10.1007/s40708-016-0042-6.

21. Sculley D., Holt G., Golovin D., Davydov E., Phillips T., Ebner D., Chaudhary V., Young M., Crespo J.-F., Dennison D. Hidden Technical Debt in Machine Learning Systems // Advances in Neural Information Processing Systems. 2015. Vol. 28. P. 2503–2511.

22. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1. Gaithersburg: U.S. Department of Commerce, 2023. DOI: https://doi.org/10.6028/NIST.AI.100-1.

23. Mosqueira-Rey E., Hernández-Pereira E., Alonso-Ríos D., Bobes-Bascarán J., Fernández-Leal Á. Human-in-the-loop Machine Learning: A State of the Art // Artificial Intelligence Review. 2023. Vol. 56. P. 3005–3054. DOI: https://doi.org/10.1007/s10462-022-10246-w.

24. Mitchell M., Wu S., Zaldivar A., Barnes P., Vasserman L., Hutchinson B., Spitzer E., Raji I. D., Gebru T. Model Cards for Model Reporting // Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* ’19). New York: ACM, 2019. P. 220–229. DOI: https://doi.org/10.1145/3287560.3287596.

25. Bender E. M., Friedman B. Data Statements for Natural Language Processing: Toward Mitigating System Bias and Enabling Better Science // Transactions of the Association for Computational Linguistics. 2018. Vol. 6. P. 587–604. DOI: https://doi.org/10.1162/tacl_a_00041.