https://doi.org/10.31891/2219-9365-2025-81-20
УДК 004.8

ВОЛОХОВСЬКИЙ Віталій
Харківський національний університет радіоелектроніки
https://orcid.org/0009-0006-5682-1889
e-mail: vitalii.volokhovskyi@nure.ua

АНАЛІЗ МЕТОДУ ПОШУКОВО-ДОПОВНЕНОЇ ГЕНЕРАЦІЇ У ГАЛУЗІ ГЕНЕРАЦІЇ ДОГОВОРІВ

Предметом дослідження є метод пошуково-доповненої генерації машинного навчання для генерації договорів в умовах обмежених ресурсів і способи порівняння та оцінки їхньої ефективності. Метою роботи є аналіз методу пошуково-доповненої генерації тексту для розробки незалежних спеціалізованих систем та оцінка їхньої ефективності для генерації договорів різними мовами у різних правових системах. У статті вирішуються такі завдання: визначення методу пошуково-доповненої генерації для адаптації моделей до вузькоспрямованих галузей; аналіз способів оцінки та порівняння таких систем; виявлення обмежень та недоліків існуючих рішень та підходів; пошук оптимального підходу за умови обмежених ресурсів. Отримано наступні результати: досліджено метод пошуково-доповненої генерації у поєднані з великими мовними моделями; розглянуто архітектуру систем на основі цього підходу, її основні структурні компоненти та можливі варіації; визначено переваги та недоліки використання у спеціалізованих галузях; порівняно два методи: великі мовні моделі загального призначення без додаткового налаштування і модифікацій та системи з використанням обраного підходу для адаптації до обраної предметної галузі. В результаті практичного експерименту було визначено, що метод пошуково-доповненої генерації значно покращує точність та повноту відповідей у юридичній галузі порівняно зі стандартними моделями, проте потребує більше часу та обчислювальних ресурсів. Стаття надає огляд методу пошуково-доповненої генерації текстової інформації з використанням великих мовних моделей, розглядає його переваги та недоліки, а також можливість застосування в умовах обмежених матеріальних та людських ресурсів. У якості прикладу в роботі розглядається спеціалізована юридична галузь та проблема генерації договорів і визначається ефективність застосування обраного методу для її вирішення.

Ключові слова: велика мовна модель, генерація природньої мови, договір, пошуково-доповнена генерація.

VOLOKHOVSKYI Vitalii
Kharkiv National University of Radio Electronics

ANALYSIS OF RETRIEVAL-AUGMENTED GENERATION METHOD IN THE AREA OF LEGAL CONTRACTS GENERATION

The subject of the study is the method of Retrieval-Augmented Generation of machine learning for generating contracts under limited resources and methods of comparing and assessing their effectiveness. The work aims to analyze the method of search-augmented text generation for the development of independent specialized systems and to assess their effectiveness for generating contracts in different languages in different legal systems. The following tasks are solved in the article: determining the method of Retrieval-Augmented Generation for adapting models to narrowly focused industries; analyzing the methods of evaluating and comparing such systems; identifying the limitations of existing solutions and approaches; finding the optimal approach under limited resources. The following results were obtained: the method of RAG was investigated in combination with large language models; the architecture of systems based on this approach, its main structural components, and possible variations were considered; the advantages and disadvantages of use in specialized industries were determined; two methods were compared: large general-purpose language models without additional tuning and modifications and systems using RAG for adaptation to the selected subject area. As a result of a practical experiment, it was determined that the RAG method significantly improves the accuracy and completeness of answers in the legal field compared to standard models but requires more time and computational resources. The article provides an overview of Retrieval-Augmented Generation method for text information generation using large language models, considers their advantages and disadvantages, and discusses the possibility of application in limited financial and human resources conditions. The paper considers a specialized legal field and the problem of contract generation and determines the effectiveness of the selected method for its solution.

Keywords: large language model, natural language generation, contract, Retrieval-Augmented Generation.

ПОСТАНОВКА ПРОБЛЕМИ У ЗАГАЛЬНОМУ ВИГЛЯДІ ТА ЇЇ ЗВ'ЯЗОК ІЗ ВАЖЛИВИМИ НАУКОВИМИ ЧИ ПРАКТИЧНИМИ ЗАВДАННЯМИ

Договори відіграють ключову роль у повсякденному житті людей та забезпеченні стабільної роботи компаній. Їхнє правильне складання та оформлення відповідно до чинного законодавства та інтересів зацікавлених сторін вимагає глибоких юридичних знань. Створення договорів вручну та внесення змін навіть для досвідчених юристів є трудомістким процесом, що може спричинити помилки під час копіювання фрагментів з інших документів, а також ускладнює повторне використання договорів у майбутньому. Через значні витрати часу на ці завдання юристи мають менше можливостей приділяти увагу клієнтам та їхнім потребам, що безпосередньо впливає на якість послуг та ефективність роботи.

Серед основних проблем, з якими стикаються юристи та компанії при роботі з договорами, можна виділити [1]:

– значні витрати часу та ресурсів на складання та аналіз договорів;

– залежність неюридичних відділів компанії від юридичної команди для укладання договорів;

– відсутність єдиної узгодженої системи документів через різноманітність форматів і формулювань.

Автоматизація створення та аналізу договорів за допомогою сучасних технологій може суттєво полегшити вирішення цих проблем. Зокрема, методи машинного навчання та великі мовні моделі продемонстрували високу ефективність у задачах генерації природної мови. Проте більшість комерційних рішень у цій сфері належать приватним компаніям, а розробка власних систем на основі мовних моделей вимагає значних обчислювальних ресурсів і великого масиву спеціалізованих тренувальних даних. Це обмежує можливості впровадження таких технологій для малих компаній, наукових установ та незалежних дослідників.

АНАЛІЗ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

У галузі розуміння, обробки та генерації природньої мови в останні роки було проведено багато досліджень. У роботі [2] представлена революційна архітектура "Трансформер", яка замінила існуючи рекурентні нейронні мережі, а робота [3] зробила важливий внесок у її вдосконалення для отримання кращих результатів автоматичної генерації тексту. Базові моделі, створенні в результаті багатьох досліджень, використовуються у різних сферах [4, 5]. Розробка та дослідження вузькоспрямованих моделей показали переваги адаптації моделей до певної галузі порівняно з загальними моделями [6, 7]. З іншого боку, сучасні великі мовні моделі можуть виконувати нові завдання у різних спеціалізованих галузях без додаткового налаштування, використовуючи тільки інформацію отриману з контексту [8, 9]. У роботі [10] було сформовано великий масив юридичних документів на різних мовах з різних країн. Дослідження вводить нову метрику оцінки виконання завдань обробки природньої мови [11]. Розроблені набори метрик оцінюють знання моделей, їхні можливості у виконанні різних завдань та дотримуванні правил [12, 13]. У спеціалізованих роботах було сформовано аналогічні набори метрик для оцінки ефективності моделей у обробці та розумінні юридичних документів [11, 14, 15]. В результаті досліджень можливостей адаптації моделей до спеціалізованих галузей без додаткового тренування параметрів було запропоновано метод пошуково-доповненої генерації [16]. Робота [17] продемонструвала високу ефективність цього методу для обробки великих баз знань. Подальші дослідження спрямовані на розробку варіацій цієї архітектури з використанням додаткових компонентів пошуку даних, різних стратегій поділу та структурування знань, а також верифікації отриманих даних [18, 19, 20].

ВИЗНАЧЕННЯ НЕВИРІШЕНИХ РАНІШЕ ЧАСТИН ЗАГАЛЬНОЇ ПРОБЛЕМИ

Обробка та генерація текстів природною мовою є складним завданням. Сучасні методи та моделі вимагають значних обчислювальних, матеріальних і людських ресурсів, що обмежує їхнє широке впровадження. Розробка архітектури "Трансформер", яка перевершує попередні нейронні мережі за ефективністю, відкрило нові перспективи для її застосування в науці, промисловості та бізнесі. Водночас навчання і розробка таких моделей стали більш ресурсозатратними, що зробило їх доступними переважно для великих корпорацій та дослідницьких установ. Проте, поява значної кількості базових мовних моделей у відкритому доступі дозволила невеликим компаніям та науковим інститутам проводити власні дослідження та створювати нові рішення.

Оскільки ці моделі тренуються на загальнодоступних даних, їхні можливості в спеціалізованих сферах є обмеженими. Тому чимало досліджень присвячені розробці методів адаптації моделей до вузькоспеціалізованих галузей. Інші роботи зосереджені на створенні моделей для конкретних сфер та їхньому налаштуванні для виконання специфічних завдань.

У сфері обробки юридичних документів та договорів дослідження здебільшого фокусуються на тренуванні параметрів моделі. Цей процес вимагає значних ресурсів і часу, а до його реалізації залучаються численні команди науковців із різних університетів і компаній. Через це для невеликих наукових та комерційних організацій можливості використання передових технологій та методів залишаються досить обмеженими.

У попередньому дослідженні було проаналізовано ефективність різних підходів адаптації загальних моделей та виявлено, що метод пошуково-доповненої генерації є найбільш оптимальним для вирішення цієї задачі в умовах обмежених ресурсів [21]. Виходячи з отриманих результатів попередньої роботи, визначимо мету поточного дослідження.

ФОРМУЛЮВАННЯ ЦІЛЕЙ СТАТТІ

Метою роботи є аналіз методів пошуково-доповненої генерації тексту для розробки незалежних спеціалізованих систем та оцінка їхньої ефективності для генерації договорів різними мовами у різних правових системах.

Сформуємо *завдання*, які необхідно виконати для досягнення поставленої мети:

– визначення підходів пошуково-доповненої генерації для адаптації моделей до вузькоспрямованих галузей;

– аналіз способів оцінки та порівняння таких систем;
– виявлення обмежень та недоліків існуючих рішень та підходів;
– пошук оптимального підходу за умови обмежених ресурсів.

Визначені завдання націлені на аналіз методу пошуково-доповненої генерації та виявлення оптимальних способів вирішення проблеми генерації договорів у юридичній сфері.

АНАЛІЗ МЕТОДІВ ПОШУКОВО-ДОПОВНЕНОЇ ГЕНЕРАЦІЇ

Розглянемо методи пошуково-доповненої генерації для вирішення проблеми адаптації загальної великої мовної моделі до спеціалізованої юридичної галузі та їх застосування для автоматичного створення договорів.

Метод *пошуково-доповненої генерації* (Retrieval-Augmented Generation) використовується у сферах, які потребують взаємодії з актуальними і точними знаннями, що постійно змінюються [16]. Він базується на використанні зовнішніх джерел даних, які містять перевірену та точну інформацію. Перед генерацією відповіді, система шукає релевантні дані у базі знань. Потім на основі запиту, контексту та додаткової інформації модель надає більш точні та аргументовані відповіді, засновані на фактах та достовірних даних.

Метод пошуково-доповненої генерації має низку переваг. Його впровадження та використання є значно економічнішим, оскільки він базується на вже існуючих мовних моделях і не вимагає додаткового навчання. Завдяки отриманню даних із баз знань система може використовувати інформацію, яка з'явилася після тренування моделі та не входила до її початкового набору даних. Звернення до перевірених і надійних джерел допомагає зменшити ймовірність появи хибної інформації (галюцинацій).

Серед основних викликів цього методу можна виокремити отримання якісних даних із баз знань, їхню оцінку та ранжування за релевантністю. Оскільки метод не передбачає навчання моделі чи зміни її параметрів, її загальні знання та функціональні можливості залишаються незмінними. Аналогічно, підтримка різних мов залежить від обраної базової моделі.

Серед недоліків можна вказати обмеження розміру вхідного контексту, що лімітує кількість прикладів та розмір інструкцій, що можна навести. Оскільки сама модель не змінюється, ефективність цього методу може бути нижчою порівняно з підходами тонкого налаштування або тренування нової моделі. Вплив цих факторів можна зменшити за рахунок використання моделей з більшим розміром контексту та покращенням способів пошуку інформації.

RAG-модель описується наступним чином: на основі вхідної послідовності x система отримує текстові документи z і використовує їх як додатковий контекст під час генерації цільової послідовності y. Основними елементами системи є пошуковий компонент та генератор. На рисунку 1 наведено схему роботи системи на основі цього підходу.

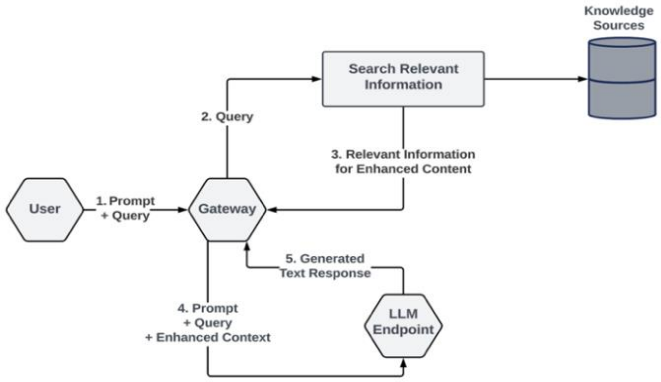

Рис. 1. Архітектура системи на основі підходу пошуково-доповненої генерації

Пошуковий компонент $p_\eta(z|x)$ знаходить k-найбільш релевантних фрагментів тексту на основі x. Dense Passage Retriever використовує щільний кодувальник, який перетворює фрагменти тексту у d-вимірні вектори дійсних чисел та створює індекс для всіх фрагментів тексту, які використовуються для пошуку [22].

Компонент генератора $p_\theta(y_i|x, z, y_{1:i-1})$, який представляється великою мовною моделлю, генерує токени тестової послідовності на основі контексту попередніх $i-1$ токенів $y_{1:i-1}$, вхідних даних x та фрагментів z, отриманих від пошукового компонента.

RAG-модель на основі токенів дозволяє компоненту генератора обирати вміст із кількох документів під час генерації відповіді. Найбільш релевантні k документів отримуються за допомогою пошукового компонента. Наступний токен розраховується таким чином:

$$p(y|x) \approx \prod_i^N \sum_{z \in top\text{-}k(p(\cdot|x))} p_\eta(z|x) p_\theta(y_i|x, z, y_{1:i-1}), \quad (1)$$

де $top\text{-}k\left(p(\cdot|x)\right)$ – k-документів з набору z найвищою попередньою ймовірністю $p_\eta(z|x)$.

Можна виділити 3 групи підходів, що використовуються для реалізації цього методу [23]:

– наївний (naive) – підхід, що заснований на індексації бази знань, отриманні даних зі сховища та генерації відповіді, в результаті можуть зустрічатися галюцинації, а отримана інформація може бути нерелевантною або повторюваною;

– розширений (advanced) – покращує якість отриманої інформації за допомогою додаткової обробки перед та після пошуку, використовує складніші методи індексування за допомогою дрібної сегментації та включення метаданих;

– модульний (modular) – вводить різноманітні стратегії для покращення компонентів шляхом додавання нових модулів: пошукових, злиття, пам'яті, маршрутизації та адаптації до завдання.

Для визначення схожості між векторами використовують наступні метрики: косинус подібності, скалярний добуток та Евклідову відстань. Проте, найбільш використаною є косинус подібності, яку можна записати наступним чином:

$$S_c(A,B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \cdot \sqrt{\sum_{i=1}^{n} B_i^2}}, \quad (2)$$

де $S_c(A,B)$ – косинус подібності, *A, B* – вектори вбудувань.

Для пошуку векторів використовують різні алгоритми. Найбільш популярними є KNN (k-nearest neighbours) пошук, а також ANN (approximate nearest neighbours) з використанням HNSW (hierarchical navigable small world) алгоритму. KNN пошук – це лінійний наївний алгоритм, який виконує порівняння вихідного вектору з усіма наявними у базі даних та повертає k-найближчих векторів. HNSW – алгоритм наближеного пошуку найближчого сусіда на основі графа, який є швидшим та ефективнішим за наївний алгоритм та підтримується сучасними векторними базами даних [24, 25].

Фрагментування даних

Розбиття документів на фрагменти (chunking) є важливим фактором ефективності систем на основі RAG, який може покращити якість отриманих даних шляхом пошуку найбільш релевантних фрагментів тексту. Існують різні стратегії поділу тексту, які можна розділити на наступні групи [26]:

– поділ фіксованого розміру – розбиває текст на однакові сегменти, але не враховує його структуру;

– рекурсивний – ітеративно ділить текст за допомогою роздільників, таких як знаки пунктуації, що дозволяє йому краще адаптуватися до вмісту;

– контекстуальний – використовує методи обробки природньої мови, такі як сегментація речень, щоб представити зміст тексту;

– структурний – розбиває текст відповідно до його структури, використовуючи заголовки, абзаци та таблиці для визначення фрагментів;

– гібридний – поєднує різні підходи, пропонуючи більшу гнучкість в обробці різноманітних типів тексту.

Структурний спосіб поділу документів було запропоновано у дослідженні [20], який ділить текст на частини керуючись не тільки довжиною сегмента, але і структурою документа. Використання логічних та семантично-пов’язаних частин на стадії генерації надає моделі більше контексту та релевантних даних для формування відповіді. Алгоритм можна описати наступним чином:

– якщо довжина тексту елемента менша за визначений розмір фрагменту, виконується спроба злиття з наступним елементом;

– тексти елементів ітеративно об’єднуються, дотримуючись попереднього кроку, доки не буде досягнуто бажаної довжини, не порушуючи структуру елементу;

– якщо елемент заголовка знайдено, починається новий фрагмент;

– якщо елемент таблиці знайдено, починається новий фрагмент, зберігаючи всю таблицю.

Оскільки, договори також є структурованими і мають розділи, параграфи та абзаци, було вирішено використати саме цей підхід для поділу документів з набору даних.

Підготовка експерименту

Для дослідження було обрано 20 договорів купівлі-продажу нерухомості. До кожного з них було сформовано по 5 запитань, відповіді на які потребують спеціалізованих знань та розуміння контексту договору. Після генерації відповідей на питання великими мовними моделями та системами на основі RAG, результати було надано на оцінку галузевим експертам.

Для оцінки якості відповідей згенерованих моделями використаємо наступні критерії: точність (Accuracy) та повнота (Comprehensiveness).

Точність визначає правильність та фактичну обґрунтованість відповіді на поставлене питання. Критерій є категоріальним через можливість значної варіації у формулюванні результату. Використаємо наступні категорії, присвоюючи кожній з них числовий еквівалент: абсолютно неправильна або нерелевантна відповідь (1); здебільшого невірно, але містить деякі елементи правди (2); частково вірно, але містить значні неточності (3); в основному правильно з невеликими неточностями (4); повністю коректно і точно (5).

Повнота визначає вичерпність відповіді на поставлене запитання. Критерій є категоріальним через можливість значної варіації у формулюванні результату. Використаємо наступні категорії, присвоюючи кожній з них числовий еквівалент: не надає корисної інформації щодо запитання, говорить про щось зовсім не пов'язане (1); відповідає на невелику частину запитання зі значними пропусками (2); охоплює більше половини необхідного вмісту, але пропускає кілька ключових аспектів (3); майже повно, але бракує кількох дрібних деталей (4); повністю вичерпний, розглядає усі аспекти питання (5).

Для оцінки ресурсів, необхідних для генерації відповіді, використаємо наступні критерії: обсяг обчислювальних ресурсів (Compute Consumption) та час генерації відповіді (Response Time).

Обсяг обчислювальних ресурсів вимірюється у кількісті спожитих токенів для генерації результату, включаючи токени запиту та відповіді. Для систем на основі RAG до вхідного контексту запиту також включаються частини документів, у яких були знайдені релевантні дані для відповіді на питання. Критерій є кількісним і має значення від 0 до 1 (після нормування).

Час генерації відповіді вимірюється у секундах, необхідних для обробки та генерації результату. Для систем на основі RAG також включається час на пошук релевантних частин документів на основі поставленого питання. Критерій є кількісним і має значення від 0 до 1 (після нормування).

Для визначення загальної ефективності системи використаємо метод адитивної згортки з ваговими коефіцієнтами. Для цього кожному з визначених критеріїв присвоюємо ваги. На основі рекомендацій експертної групи було визначено, що критерії точності та повноти мають найбільшу вагу відповідно до мети дослідження, в той час як обсяг обчислювальних ресурсів та час генерації відповіді мають менше значення. В результаті експертної оцінки було визначено наступні значення вагів: точність – 0,35, повнота – 0,35, обсяг обчислювальних ресурсів – 0,2, час генерації відповіді – 0,1.

Для дослідження було обрано наступні великі мовні моделі: GPT-4o, Gemini 1.5 Pro і Llama 3.1 8B. GPT-4o та Gemini 1.5 Pro представляють приватні великі моделі загального призначення із сотнями мільярдів параметрів, які є найкращими моделями на ринку. Llama 3.1 8B – це невелика модель загального призначення з відкритим вихідним кодом. GPT-4o демонструє високу точність і глибоке розуміння контексту та є ефективною в обробці природної мови. Gemini 1.5 Pro є конкуруючою моделлю GPT-4o, яка також забезпечує високу продуктивність у завданнях, що вимагають детальної обробки та класифікації тексту. Llama 3.1 8B значно менша порівняно з іншими моделями, проте її ефективний дизайн забезпечує конкурентоспроможні можливості обробки мови.

Результати експерименту

Після генерації відповідей кожною моделлю з використанням різних підходів, результати було передано на оцінку експертам. Кожен експерт оцінював точність та повноту відповідей на кожне запитання щодо обраних документів. Їхні оцінки було усереднено для кожного питання та документу. Також було виміряно обсяг використаних ресурсів (токенів) та час генерації відповідей.

Для визначення оптимального підходу, критерії необхідно нормувати на проміжку [0, 1], використовуючи мінімальне та максимальне значення. Виходячи з того, що вони потребують як мінімізації, так і максимізації, перетворимо їх таким чином, щоб більша корисність з точки зору поставленої задачі відповідала більшому значенню. Перетворенню підлягають критерії обсягу обчислювальних ресурсів та часу генерації відповіді. Отримані результати наведено у таблиці 1.

Таблиця 1

Результати оцінки генерації відповідей

Модель	Точність		Повнота		Економія ресурсів		Економія часу	
	LLM	RAG	LLM	RAG	LLM	RAG	LLM	RAG
GPT-4o	0,70	0,95	0,65	0,97	0,70	0,60	0,50	0,50
Gemini 1.5 Pro	0,60	0,91	0,55	0,94	0,00	0,00	0,00	0,00
Llama 3.1 8B	0,47	0,72	0,40	0,70	1,00	1,00	1,00	1,00

Далі наведемо результати вирішення задачі оптимізації за допомогою адитивної згортки з ваговими коефіцієнтами та визначимо корисність кожного з методів та моделей (табл. 2).

Таблиця 2

Корисність методів адаптації до вузькоспрямованої галузі

Модель	Корисність	
	LLM	RAG
GPT-4o	0,663	**0,842**
Gemini 1.5 Pro	0,403	0,648
Llama 3.1 8B	0,605	0,797

Виходячи з отриманих результатів аналізу, можемо зробити висновок, що використання методу пошуково-доповненої генерації значно покращує точність та повноту відповідей у вузькоспрямованій галузі, порівняно зі стандартними моделями. Також ми бачимо, що цей підхід споживає більшу кількість ресурсів, що обумовлено передачею додаткових частин документів до контексту моделі під час запиту. Збільшення часу обробки спричинено додатковою операцією пошуку релевантних документів у базі знань, а також більшим обсягом даних, на основі якого модель генерує відповідь. GPT-4o з використанням RAG виявилась найбільш ефективною за визначеними критеріями серед обраних моделей. Значно менша за розмірами Llama 3.1 8B продемонструвала значний виграш у часі обробки запиту, а також у обсязі використаних токенів, проте точність та повнота відповідей виявились гіршим, порівняно з більшими моделями.

ВИСНОВКИ З ДАНОГО ДОСЛІДЖЕННЯ І ПЕРСПЕКТИВИ ПОДАЛЬШИХ РОЗВІДОК У ДАНОМУ НАПРЯМІ

В результаті дослідження було проаналізовано методи пошуково-доповненої генерації у поєднані з великими мовними моделями, розглянуто архітектуру систем на основі цього підходу, її структуру та можливі варіації, визначено переваги та недоліки для використання у спеціалізованих галузях. У досліджені було розглянуто та порівняно два методи: великі мовні моделі загального призначення без додаткового налаштування і модифікацій та системи з використанням RAG для адаптації до обраної предметної галузі.

Для порівняння підходів між собою, було сформовано тестовий набір даних, який складається з договорів та питань до них, а також набір критеріїв оцінки моделей. Для визначення найбільш ефективного підходу було використано метод лінійної адитивної згортки з ваговими коефіцієнтами.

В результаті практичного експерименту було виявлено, що використання методу пошуково-доповненої генерації значно покращує точність та повноту відповідей, досягаючи значень у 90%, у обраній галузі порівняно зі стандартними моделями. Однак, обсяг спожитих ресурсів та часу збільшується через необхідність пошуку та обробки додаткового контексту. В ході порівняння різних мовних моделей було виявлено, що GPT-4o генерує найбільш точні відповіді, в той час як Llama 3.1 8B виявилась найбільш швидкою та ресурсоефективною для виконання поставленого завдання.

Таким чином, метод пошуково-доповненої генерації продемонстрував високу ефективність адаптації великих мовних моделей до спеціалізованих галузей за умови обмежених обчислювальних та часових ресурсів.

ПЕРСПЕКТИВИ ПОДАЛЬШОГО РОЗВИТКУ

У наступних дослідженнях планується приділити увагу іншим варіаціям методу пошуково-доповненої генерації, таким як зміна структури зберігання даних, наприклад, у вигляді графа, та використання більш складних підходів пошуку та обробки даних перед стадією генерації, запропонованих у розширеному та модульному підходах до імплементації RAG.

ПОДЯКИ

Автор висловлює подяку Збройним Силам України за можливість написати повноцінну роботу під час повномасштабного вторгнення Російської Федерації на територію України. Також дякує науковому керівнику О. С. Назарову за підтримку та допомогу під час написання роботи.

Література

1. Generative AI for Legal Contracts. Nasdaq. URL: https://www.nasdaq.com/articles/generative-ai-for-legal-contracts (дата звернення: 27.05.2024).

2. Vaswani, A. та ін. Attention is all you need. Advances in neural information processing systems. *31st Conference on Neural Information Processing Systems*. 2017. 30. DOI: https://doi.org/10.48550/arXiv.1706.03762

3. Devlin J. та ін. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. 2018. DOI: https://doi.org/10.48550/arXiv.1810.04805

4. Touvron H. та ін. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*. 2023. DOI: https://doi.org/10.48550/arXiv.2307.09288

5. Jiang A. Q. та ін. Mixtral of experts. *arXiv preprint arXiv:2401.04088*. 2024. DOI: https://doi.org/10.48550/arXiv.2401.04088

6. Wu S. та ін. BloombergGPT: A large language model for finance. *arXiv preprint arXiv:2303.17564*. 2023. DOI: https://doi.org/10.48550/arXiv.2303.17564

7. Singhal K. та ін. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*. 2023. DOI: https://doi.org/10.48550/arXiv.2305.09617

8. Brown T. та ін. Language models are few-shot learners. *Advances in neural information processing systems*. 2020. № 33. С. 1877–1901. DOI: https://doi.org/10.48550/arXiv.2005.14165

9. Nori H. та ін. Can generalist foundation models outcompete special-purpose tuning? Case study in medicine. *arXiv preprint arXiv:2311.16452*. 2023. DOI: https://doi.org/10.48550/arXiv.2311.16452

10. Niklaus J. та ін. Multilegalpile: A 689gb multilingual legal corpus. *arXiv preprint arXiv:2306.02069*. 2023. DOI: https://doi.org/10.48550/arXiv.2306.02069

11. Guha N. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in Neural Information Processing Systems*. 2024. № 36. DOI: https://doi.org/10.48550/arXiv.2308.11462

12. Hendrycks D. та ін. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*. 2020. DOI: https://doi.org/10.48550/arXiv.2009.03300

13. Wang A. та ін. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*. 2019. № 32. DOI: https://doi.org/10.48550/arXiv.1905.00537

14. Chalkidis I. та ін. LexGLUE: A Benchmark Dataset for Legal Language Understanding in English. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*. 2022. 1. С. 4310–4330. DOI: https://aclanthology.org/2022.acl-long.297

15. Niklaus J. та ін. LEXTREME: A Multi-Lingual and Multi-Task Benchmark for the Legal Domain. *Findings of the Association for Computational Linguistics: EMNLP 2023*. 2023. С. 3016–3054. DOI: https://aclanthology.org/2023.findings-emnlp.200

16. Lewis P. та ін. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*. 2020. № 33. С. 9459–9474. DOI: https://doi.org/10.48550/arXiv.2005.11401

17. Borgeaud S. та ін. Improving language models by retrieving from trillions of tokens. *International conference on machine learning*. 2022. С. 2206–2240. DOI: https://doi.org/10.48550/arXiv.2112.04426

18. Rackauckas Z. Rag-Fusion: A New Take on Retrieval Augmented Generation. *International Journal on Natural Language Computing*. 2024. Т. 13, № 1. С. 37–47. DOI: https://doi.org/10.5121/ijnlc.2024.13103

19. Gao L. та ін. Rarr: Researching and revising what language models say, using language models. *arXiv preprint arXiv:2210.08726*. 2022. DOI: https://doi.org/10.48550/arXiv.2210.08726

20. Yepes A. J. та ін. Financial report chunking for effective retrieval augmented generation. *arXiv preprint arXiv:2402.05131*. 2024. DOI: https://doi.org/10.48550/arXiv.2402.05131

21. Volokhovskyi V. Analysis of methods for training domain-specific language models in the area of legal contracts generation. *Innovative Technologies and Scientific Solutions for Industries*. 2024. № 2(28). С. 48–64. DOI: https://doi.org/10.30837/2522-9818.2024.2.048

22. Lewis P. та ін. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*. 2020. № 33. С. 9459–9474. DOI: https://doi.org/10.48550/arXiv.2005.11401

23. Karpukhin, V. та ін. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*. 2020. DOI: https://doi.org/10.48550/arXiv.2004.04906

24. Gao Y. та ін. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*. 2023. DOI: https://doi.org/10.48550/arXiv.2312.10997

25. Malkov Y. A. та ін. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2020. Т. 42, № 4. С. 824–836. DOI: https://doi.org/10.1109/tpami.2018.2889473

26. Atlas Vector Search Overview. *MongoDB: The Developer Data Platform*. URL: https://www.mongodb.com/docs/atlas/atlas-vector-search/vector-search-overview/ (дата звернення: 23.02.2025).

27. Schwaber-Cohen R. Chunking Strategies for LLM Applications. *The vector database to build knowledgeable AI, Pinecone*. URL: https://www.pinecone.io/learn/chunking-strategies/ (дата звернення: 23.02.2025).

References

1. Generative AI for Legal Contracts. Nasdaq. URL: https://www.nasdaq.com/articles/generative-ai-for-legal-contracts (date of access: 27.05.2024).

2. Vaswani, A. et al. Attention is all you need. Advances in neural information processing systems. *31st Conference on Neural Information Processing Systems*. 2017. 30. DOI: https://doi.org/10.48550/arXiv.1706.03762
3. Devlin J. et al. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. 2018. DOI: https://doi.org/10.48550/arXiv.1810.04805
4. Touvron H. et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*. 2023. DOI: https://doi.org/10.48550/arXiv.2307.09288
5. Jiang A. Q. et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*. 2024. DOI: https://doi.org/10.48550/arXiv.2401.04088
6. Wu S. et al. BloombergGPT: A large language model for finance. *arXiv preprint arXiv:2303.17564*. 2023. DOI: https://doi.org/10.48550/arXiv.2303.17564
7. Singhal K. et al. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*. 2023. DOI: https://doi.org/10.48550/arXiv.2305.09617
8. Brown T. et al. Language models are few-shot learners. *Advances in neural information processing systems*. 2020. № 33. P. 1877–1901. DOI: https://doi.org/10.48550/arXiv.2005.14165
9. Nori H. et al. Can generalist foundation models outcompete special-purpose tuning? Case study in medicine. *arXiv preprint arXiv:2311.16452*. 2023. DOI: https://doi.org/10.48550/arXiv.2311.16452
10. Niklaus J. et al. Multilegalpile: A 689gb multilingual legal corpus. *arXiv preprint arXiv:2306.02069*. 2023. DOI: https://doi.org/10.48550/arXiv.2306.02069
11. Guha N. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in Neural Information Processing Systems*. 2024. № 36. DOI: https://doi.org/10.48550/arXiv.2308.11462
12. Hendrycks D. et al. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*. 2020. DOI: https://doi.org/10.48550/arXiv.2009.03300
13. Wang A. et al. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*. 2019. № 32. DOI: https://doi.org/10.48550/arXiv.1905.00537
14. Chalkidis I. et al. LexGLUE: A Benchmark Dataset for Legal Language Understanding in English. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*. 2022. 1. P. 4310–4330. DOI: https://aclanthology.org/2022.acl-long.297
15. Niklaus J. et al. LEXTREME: A Multi-Lingual and Multi-Task Benchmark for the Legal Domain. *Findings of the Association for Computational Linguistics: EMNLP 2023*. 2023. P. 3016–3054. DOI: https://aclanthology.org/2023.findings-emnlp.200
16. Lewis P. et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*. 2020. № 33. P. 9459–9474. DOI: https://doi.org/10.48550/arXiv.2005.11401
17. Borgeaud S. et al. Improving language models by retrieving from trillions of tokens. *International conference on machine learning*. 2022. P. 2206–2240. DOI: https://doi.org/10.48550/arXiv.2112.04426
18. Rackauckas Z. Rag-Fusion: A New Take on Retrieval Augmented Generation. *International Journal on Natural Language Computing*. 2024. Vol. 13, № 1. P. 37–47. DOI: https://doi.org/10.5121/ijnlc.2024.13103
19. Gao L. et al. Rarr: Researching and revising what language models say, using language models. *arXiv preprint arXiv:2210.08726*. 2022. DOI: https://doi.org/10.48550/arXiv.2210.08726
20. Yepes A. J. et al. Financial report chunking for effective retrieval augmented generation. *arXiv preprint arXiv:2402.05131*. 2024. DOI: https://doi.org/10.48550/arXiv.2402.05131
21. Volokhovskyi V. Analysis of methods for training domain-specific language models in the area of legal contracts generation. *Innovative Technologies and Scientific Solutions for Industries*. 2024. № 2(28). P. 48–64. DOI: https://doi.org/10.30837/2522-9818.2024.2.048
22. Lewis P. et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*. 2020. № 33. P. 9459–9474. DOI: https://doi.org/10.48550/arXiv.2005.11401
23. Karpukhin, V. et al. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*. 2020. DOI: https://doi.org/10.48550/arXiv.2004.04906
24. Gao Y. et al. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*. 2023. DOI: https://doi.org/10.48550/arXiv.2312.10997
25. Malkov Y. A. et al. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2020. Vol. 42, № 4. P. 824–836. DOI: https://doi.org/10.1109/tpami.2018.2889473
26. Atlas Vector Search Overview. *MongoDB: The Developer Data Platform*. URL: https://www.mongodb.com/docs/atlas/atlas-vector-search/vector-search-overview/ (date of access: 23.02.2025).
27. Schwaber-Cohen R. Chunking Strategies for LLM Applications. *The vector database to build knowledgeable AI, Pinecone*. URL: https://www.pinecone.io/learn/chunking-strategies/ (date of access: 23.02.2025).