„Платформа за локално обучение на големи езикови модели“

“Platform for local fine-tuning of Large Language Models”

маг. инж. С. Стойкова

Технически университет – София, филиал Пловдив
Факултет Електроника и Автоматика
email: stoykova@tu-plovdiv.bg

Abstract: The study presents a platform designed for local training and inference of popular large language models (LLMs), addressing the growing need for solutions that preserve the privacy of training data. The platform enables researchers and developers to further fine-tune base models using locally stored and processed data through a modular software architecture. It supports a wide range of models and tasks, offering seamless integration with existing machine learning workflows and tools. Benchmark results demonstrate competitive performance in terms of latency, throughput, and resource utilization compared to cloud-based alternatives. By decentralizing model development and inference, the platform empowers institutions to maintain data sovereignty, reduce operational costs, and promote sustainable practices in the field of artificial intelligence.

Keywords: large language models, management information systems, artificial intelligence, machine learning, local fine-tuning.

Въведение

Осъществено литературно проучване по темата “Приложения на изкуствен интелект в информационните системи за управление” [1], показва, че бизнес организациите, генерират стойност чрез итегрирането на приложения, базирани на изкуствен интелект в три основни направления.

- Автоматизация на процесите, т.е. стойността, генерирана поради автоматизацията на бизнес процесите и прехвърлянето на изпълнението на задачи с ниска добавена стойност от служители към интелигентни инструменти, базирани на ИИ.

- Интелигентен бизнес и оперативен анализ, т.е. стойността, генерирана благодарение на анализ на бизнес информация и данни, базиран на методи и техники от областта на машинното обучение.

- Когнитивно взаимодействие, т.е. стойността, генерирана поради взаимодействия на служители или клиенти с дигитални асистенти, чатботове и базирани на генеративен изкуствен интелект инструменти и приложения.

В идентифицираните основни литературни източници преобладава внедряването на приложения, базирани на ИИ в областта на автоматизацията на процеси, като по-малко от половината от разгледаните изследвания се концентрират върху извличането на стойност чрез интегриране на ИИ в направленията на бизнес оперативен анализ и когнитивно взаимодействие. Това показва необходимостта от допълнително изследване на приложимостта на ИИ в тези две направления.

Задачите, основаващи се на когнитивно взаимодействие представляват значителен дял от ежедневните операции, осъществявани в информационните системи за управление в индустрията, което ги превръща във фокус на редица научни изследвания и разработки, както в държавния, така и в частния сектор [2], [3]. Тези задачи включват динамичен обмен на информация между потребители/служители и машинен интерфейс, софтуерни платформи и интелигентни дигитални агенти, за целите на извличане на познание, систематизиране и обобщаване на информация, генериране на съдържание и подпомагане на процеса по вземане на решения. Това означава, че е необходим интердисциплинарен подход в разработването на модели и изследването на възможни решения в областта.

Въпреки, че взаимодействията човек-машина не са нова област на изследване, а всъщност са обект на засилен научено-изследователски интерес още от зората на компютърните технологии и информационните науки, развитието на тази област е драстично повлияно в последните няколко години от приоретизираното разработване и масово интегриране на технологии, базирани на методите на изкуствен интелект, специфично тези с генеративен характер като големите езикови модели (Large Language Models, LLMs).

Приоретизираната масова комерсиализация на големи езикови модели (LLMs) като ChatGPT, Copilot, Gemini в перода от 2022 до момента доведе до преориентиране на редица дистрибутори на интегрирани информационни системи за управление към внедряването на т.нар „интелигентни дигитални агенти“ (Joule за SAP, Copilot за Dynamics 365, Oracle AI Assistant), базирани на LLM модели към облачните им платформи. Прилагането на LLM-базирани решения носи редица предизвикателства по отношение на точност на генерираното съдържание и конфиденциалност на обработваните данни. Отсъствието на прозрачност (explanability) на компоненти, базирани на машинно обучение и уязвимостта на приложения, базирани на генеративен ИИ към кибератаки поради голямата свързаност и отвореност на базовите модели

(LLMs) се явяват сериозни предизвикателства пред успешното интегриране на технологиите, базирани на изкуствен интелект в съществуващите системи за индустриална автоматизация [4]. В този контекст, осигуряването на прозрачност, проследимост и обяснимост в процесите на последващо обучение (fine-tuning) и инференция на големи езикови модели се превръщат в основни цели при разработката на приложения, базирани на генеративен изкуствен интелект и при тяхното внедряване към съществуващи информационни системи за управление в индустрията.

Настоящото изследване има за цел да представи разработена платформа, предназначена за локално последващо обучение (fine-tuning) и инференция на популярни големи езикови модели (LLMs), в отговор на нарастващата нужда от решения, които да съхраняват поверителността и конфиденциалността на използваните за обучение данни и на генерираното в резултат на инференция съдържание. Платформата позволява на изследователи и разработчици да осъществяват допълнително обучение на базовите модели върху локално съхранявани и обработвани данни посредством модулна софтуерна архитектура. Платформата поддържа широк спектър от базови модели и задачи, като предлага лесна интеграция с съществуващи ML процеси и инструменти. Резултатите от тестовете показват конкурентна производителност по отношение на латентност, пропускателна способност и използване на ресурси в сравнение с облачни алтернативи. Разработената платформа „Private Edge LLMs“:

- Позволява обучение с локален ресурс (локални модели: llama, deepseek-R1, gemma с наличния изчислителен ресурс на периферното устройство: компютър, лаптоп) и локално съхранение и обработка на данните, подход, който е предпочитан при задачи, изискващи конфиденциалност на данните за обучение, подходящ за приложения в научно-изследователски институции.

- Позволява връзка с API към нелокални изчислителни ресурси (GPU, TPU масиви за обучение, достъпни в облака) и/или големи езикови модели в облака (GPT).

- Внедрява функция за „генериране с добавено извеждане” (Retrieval-augmented generation, RAG), която подобрява възможностите на големите езикови модели (LLM), като ги комбинира с външни за модела източници на знания, предоставяйки допълнителен контекст за изпълнение на промпта (тези източници могат да бъдат локални за периферното устройство, на което се осъществява допълнителното обучение или споделени в облака). Тази функция адресира някои от ограниченията на LLM, като например остаряването на информацията използвана в първоначалното обучение и помага за редуцирането на т.нар. “халюцинации” (генериране на неточна информация), като позволява на големите езикови модели да имат достъп и да включват информация от конкретни документи или бази данни при генериране на отговори.

- Предоставя подобрение в проследимостта на източника, използван за генериране на текстово съдържание чрез функция „цитиране на източник“.

За да бъде осъществено едновременно внедряване на множество големи езикови модели и да бъде управлявано тяхното последващо обучение върху локални ресурси е необходимо да бъде изградена локална софтуерна платформа, основаваща се на модулна софтуерна архитектура, която да дава необходимата гъвкавост и адаптивност към разнообразието от архитектури и подходи за обучение на големи езикови модели.

Модулна софтуерна архитектура

Модулната софтуерна архитектура, интегрирана в разработената платформа „Private Edge LLMs“, включва следните елементи:

- Потребителски интерфейс на приложния слой: позволява селекция на базовия голям езиков модел, който ще бъде обучаван от страна на потребителя; селекция на източници на данни за допълнителното обучение и селекция на режима на работа: RAG (генериране на съдържание с добавено извеждане), Search (контекстово генериране на съдържание, където се извеждат като резултат само четирите най-релевантни за потребителската заявка информационни блока) и Basic (неконтекстова инференция на дадения голям езиков модел). Потребителският интерфейс е изграден на база на отворената библиотека на Python за създаване на уеб интерфейси за модели, базирани на машинно обучение, Gradio.

- Модул за оркестрация на задачи спрямо избрания режим на работа: Модулът за оркестрация е реализиран на базата на оркестрационната рамка LlamaIndex, която осъществява управление на процесите по обработка на външни за модела източници на знания, индексиране на обработената информация, извличане на информационни блокове на база контекст и генериране на промпт (допълнена с контекст потребителска заявка). Оркестрационната рамка LlamaIndex е избрана, заради нейната гъвкавост по отношение на използването на различни модели за векторизация и възможността за интегриране на векторни бази данни с отворен код.

- Модул за планиране на етапи на изпълнение на заявката: задава последователността на изпълнение на потребителската заявка въз основа на шаблон за поведение, отговарящ на избрания режим на работа: RAG и Search режимите разчитат на допълнително контекстуализиране на потребителската заявка, докато Basic режима позволява инференция на оригиналния голям езиков модел без контекст от външна за модела база знания.

- Модул за контекстуализиране на потребителските заявки: подготвя външната за големия езиков модел база знания (може да бъде локална или разположена в облака). Изходните данни, представени в различни формати (.pdf, .docx, .csv, .pptx, .ipynb, .json, .epub), преминават през

процеси на почистване и извличане на текстово съдържание. Полученият текст се сегментира на информационни блокове на база контекст (context chunks), които се преобразуват във векторни представяния (embeddings) с помощта на предварително обучен модел и се съхраняват в динамичната памет.

- Динамична памет: е концепция, която надгражда традиционната архитектура на големите езикови модели (LLMs), като им позволява да: запомнят потребителски предпочитания и контекст от предишни разговори, да адаптират поведението си спрямо натрупаната информация, да извличат релевантни спрямо потребителската заявка данни от външни или вътрешни хранилища при нужда. Тази функционалност е особено значение за приложения като дигитални асистенти, персонализирани чатботове и други системи, които трябва да поддържат дългосрочна и последователна комуникация. Динамичната памет за разработваната платформа е реализира посредством векторна база данни Qdrant. Qdrant е високопроизводителна векторна база данни с отворен код, предназначена за семантично търсене в приложения с изкуствен интелект. Тя е оптимизирана за работа с векторни представяния (embeddings), които се използват широко в системи за търсене и генериране с добавено извеждане (RAG).

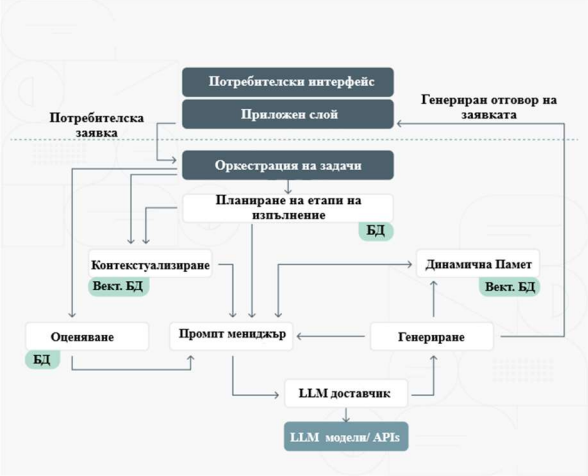
- Модул за оценяване: определя релевантността на индексираните контекстово сегментирани информационни блокове спрямо поставената потребителска заявка на база на сходство между векторните им представяния, т.е. извършва търсене във векторната база, изчислявайки метрика на сходство (напр. косинусова близост или скаларно произведение) между вектора на потребителската заявка и векторите на индексираните контекстово сегментирани информационни блокове (context chunks).

- Модул за генериране на отговор: формира допълнена с контекст потребителска заявка (контекстов промпт) за големия езиков модел. Този промпт обикновено включва оригиналната потребителска заявка и текстовото съдържание на извлечените в предната стъпка най-релевантни контекстово сегментирани информационни блокове.

- Промпт мениджър: подава, формулирания от модула за генериране на отговор, контекстов промпт на LLM доставчика.

- LLM доставчик: поддържа локална база с големи езикови модели, както и интеграция с частни или публични модели, реализирани в облака посредством API. За управление на зависимости, параметри и инсталационни пакети на големите езикови модели е използван Poetry (съвременен инструмент за управление на зависимости и пакетиране на Python проекти, който улеснява създаването, изграждането и публикуването на Python библиотеки и приложения).

На Фиг. 1 е представена обобщената структура на разработената платформа за допълнително обучение на големи езикови модели.



Фиг. 1 Модулна архитектура на разработената платформа за допълнително обучение на големи езикови модели.

Внедряване на схема за обучение RAG

“Генериране с добавено извеждане” (Retrieval-augmented generation, RAG), е подход, при който съдържанието, генерирано от голям езиков модел, се основава на данни, извлечени от външни за модела източници като файлове, бази данни, уеб страници и др. Схемата за обучение RAG осъществява допълнително обучение в два основни етапа: първо се извличат релевантни документи или техни фрагменти от външна за големия езиков модел база знания въз основа на потребителската заявка; след това тази информация се комбинира със специални контекстови посказки (т.нар контекстови векторни съответствия), които насочват модела как да използва данните, за да се формира контекстът за генериране на крайния отговор.

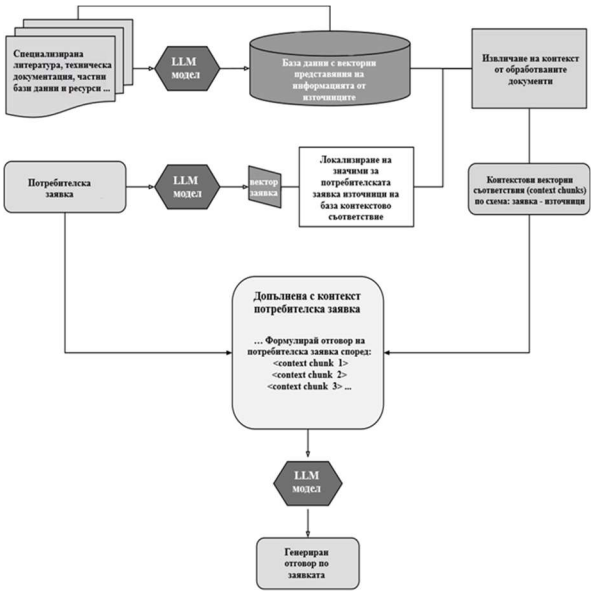
В зависимост от инструкциите в контекстовите подсказки, генерираният отговор може да включва цитати или препратки към оригиналните документи, което повишава прозрачността и доверието на потребителите, като им позволява да проверят информацията. RAG помага да се преодолеят ограниченията на LLM, като остаряла информация, неточности или „халюцинации“. Тази схема за обучение разширява базата знания на LLM до практически неограничени размери, осигурявайки бърз достъп до специализирани области или вътрешни данни на организации, научноизследователски институции, университетски библиотеки и т.н.

Функционалните стадии на обработка на информация при внедряване на архитектура за обучение тип RAG са:

- Индексиране (Indexing): На този етап се подготвя външната за големия езиков модел база знания. Изходните данни, представени в различни формати,

преминават през процеси на почистване и извличане на текстово съдържание. Полученият текст се сегментира на информационни блокове на база контекст (context chunks), които се преобразуват във векторни представяния (embeddings) с помощта на предварително обучен модел. Генерираните вектори, заедно със съответните текстови фрагменти, се индексират и съхраняват в специализирана векторна база данни, оптимизирана за търсене по сходство.

- Извличане (Retrieval): При получаване на потребителска заявка, тя се векторизира с помощта на същия модел, използван при индексирането за създавне на векторните представяния (embeddings) на контекстово сегментираните информационни блокове (context chunks). След това системата извършва търсене във векторната база, изчислявайки метрика на сходство (напр. косинусова близост или скаларно произведение) между вектора на потребителската заявка и векторите на индексираните контекстово сегментирани информационни блокове (context chunks). Извлича се предварително определен брой (K) най-релевантни контекстово сегментирани информационни блока, които имат най-висока степен на сходство.



Фиг. 2 Представяне на внедрената схема за обучение тип RAG.

- Генериране (Generation): В последния етап се формира допълнена с контекст потребителска заявка (контекстов промпт, модифицирана потребителска заявка) за големия езиков модел. Този промпт обикновено включва оригиналната потребителска заявка и текстовото съдържание на извлечените в предната стъпка най-релевантни контекстово сегментирани информационни блокове. LLM моделът обработва този контекстов промпт и генерира финален отговор на потребителската заявка. По време на генерирането моделът може да използва както информацията от предоставения контекст, така и своите вътрешни параметрични знания, или да бъде инструктиран да се придържа стриктно само към извлечените данни.

На Фиг. 2 е представена диаграма на внедрената архитектура за обучение тип RAG, представяща подробно трите функционални стадия на обработка на потребителската заявка и осъществяване на последващо обучение.

Внедрената схема за обучение RAG съчетава силните страни на традиционните системи за извличане на информация като търсачки и бази данни (проследимост и прозрачност на източниците на информация) с възможностите за систематизиране и анализ на информация на генеративните големи езикови модели.

Заключения

Представената в изследването разработена платформа „Private Edge LLMs“ за локално последващо обучение (fine-tuning) и инференция на популярни големи езикови модели (LLMs) е подходяща за обработка на данни в научно-изследователски институции Внедрената схема за обучение RAG подобрява възможностите на големите езикови модели, като ги комбинира с външни за модела източници на знания, предоставяйки допълнителен контекст за изпълнение на потребителските заявки. Реализацията на схема за обучение RAG адресира и предоставя адекватна компенсация на основни слабости на приложенията, базирани на големи езикови модели, като прозрачност и проследимост на източниците на генерирано съдържание и конфиденциалност на обработваните данни за обучение.

Благодарности

Авторите изказват благодарност на сектор „Научноизследователска и развойна дейност“ при Техническия университет – София за финансовата подкрепа по проект 242ПД0033-19.

Литературни източници

[1] Stoykova, S.; Shakev, N. Artificial Intelligence for Management Information Systems: Opportunities, Challenges, and Future Directions. Algorithms 2023, 16, 357. https://doi.org/10.3390/a16080357

[2] Wessel, M., Adam, M., et al., Generative AI and its Transformative Value for Digital Platforms., *Journal of Management Information Systems*, *42*(2), 346–369. https://doi.org/10.1080/07421222.2025.2487315

[3] Storey, Veda C., et al. "Generative artificial intelligence: Evolving technology, growing societal impact, and opportunities for information systems research." *Information Systems Frontiers* (2025): 1-22. https://doi.org/10.1007/s10796-025-10581-7

[4] Хаджийски, М., Разширяване на обхвата на индустриалната автоматизация с ползване на технологиите на изкуствения интелект. – Автоматика и информатика, год. LVIII, 2025, № 1.