DOI: https://doi.org/10.37075/ISA.2025.2.09

Изкуствен интелект като услуга

Ваня Лазарова*

Резюме: В тази разработка се прави опит да се даде отговор на въпроса как малките и средни компании да използват изкуствения интелект като услуга (AIaaS), като включат и контекст с фирмените си данни, без това да води до публичност и достъп до чувствителна фирмена информация. Разгледани са технологиите, които биха подпомогнали изграждането на една такава система за заявки, като всяко запитване се съпровожда от допълнителна фирмена информация или „контекст на запитването“.

Ключови думи: големи езикови модели, модели за влагащи вектори, RAG-технологии, векторни бази данни.

JEL: C3, C6.

1. Въведение

След старта на ChatGPT през 2022 г. големите езикови модели (LLM) се прилагат за решения на много различни задачи в науката и бизнеса. Разработването и използването на LLM изисква много изчислителни и финансови ресурси. Малките и средни компании нямат такива ресурси, те могат да използват LLM само като услуга. В тази разработка се опитваме да дадем отговор на въпроса как малките и средни компании да използват изкуствения интелект като услуга, като използват като контекст фирмените си данни без това да води до публичност и достъп до тяхната чувствителна информация. Данните, които трябва да бъдат запазени, най-вече са финансовите данни, данните за здравето и друга лична информация за служителите, пароли за достъп, банкови сметки и пр. Има различни модели, които могат да запазят тази вътрешнофирмена информация (Hristov, 2024).

Проблемът, на който трябва да се намери решение, е следният: фирма има разработен онлайн магазин, книжарница или сайт за достъп до услуги. Когато потребителите търсят дадена стока или услуга, те обикновено въвеждат в полето за търсене няколко ключови думи. Информационната система ще даде отговор, но само ако в нея услугите, книгите, стоките и пр. са описани със същите думи. И ще бъдат пропуснати много, които също биха отговорили на потребителското търсене, но не са описани в системата по търсения от потребителя начин. Тъй като фирмата не иска да изпуска нито един клиент, може тази заявка, която е направил потребителят, да я насочи към някой от безплатните чатове с изкуствен

* Ваня Лазарова е доктор, доцент в катедра „Информационни технологии и комуникации“ на УНСС.

интелект и това, което той отговори, да се сложи в полето като отговор. Няма възможност за разработка и обучение на свой собствен чатбот, затова може да ползва всеки популярен публичен чатбот, като услуга. Проблемът е, че отговорът, който ще даде публичният чатбот, ще бъде твърде общ и няма да бъде обвързан по никакъв начин с тези услуги, книги, стоки и пр., които има във фирмената база данни. От друга страна, фирмените данни могат да се използват като контекст на заявката, но това не трябва да води до публичност и разпространение на чувствителна за фирмата информация.

Трябва така да се обработят заявката и отговора, че те да съответстват на фирмените данни, да ги запазят и същевременно да бъдат максимално полезни на крайния потребител.

За да се направи това, тук трябва да се използват няколко нови съвременни технологии и метод, наречен **„Учене с малко допълнителна информация“** (Few-shot learning, FSL). Това е метод, който се използва при моделите за разпознаване на образи. Показват се няколко нови изображения на предварително обучен модел и се извлича точен отговор на задача за откриване или класифициране на обект. В по-широк смисъл, това е способността на големия лингвистичен модел да отговаря на запитвания от нова област (в която моделът не е обучен) само с помощта на допълнителна информация (контекст на запитването), без да се прилага допълнително обучение на модела.

2. Използвани нови технологии

2.1. Големи езикови модели (LLM)

Големите езикови модели LLM (Large Language Models) са вид невронномрежови модели, предназначени да обработват и генерират текст на естествен език. Те са „големи“, защото са обучени върху огромни количества текстови данни и съдържат милиарди параметри. Размерът на LLM нарасна много бързо от 2023 г. насам.

LLM са големи също и от гледна точка на енергията и времето, необходими за обучението им. Компанията Мета обяви публично времето и електрическата енергия, необходими за обучението на един модел.

Таблица 1. *Време и електрическа енергия, необходими за обучението на един модел*

Модел	Размер на модула (В - милиарди)	Време за обучение (процесорно време в милион часа)	Изразходвана електрическа енергия (в мегават часа)**
Llama 3.1 405B	**405 B**	30.8 M	**21 560 MWh**
Llama 3.1 70B	**70 B**	7.0 M	**4 900 MWh**

Преработено от източник: *https://huggingface.co/meta-llama/Llama-3.1-405B-Instruct (meta-llama/Llama-3.1-405B-Instruct, 2025)*

2.2. Езикови модели за влагащи вектори (embedding models)

Езиковите модели за влагащи вектори на текстове (Embedding models) са вид езикови модели, които генерират вектор, съответстващ на даден текст. Текстът може да е единично изречение, но също така може да бъде бизнес документ, статия от списание, книга и т.н. Векторът представя координатите на този текст в многомерното пространство на модела.

Независимо от това, че моделите за влагащи вектори се наричат „езикови“, някои модели могат да представят аудио и видео обекти. Фокусът по-нататък ще бъде само върху моделите, които представят текст.

Влагащите вектори (embedding vectors) представят семантичното значение на текстовете в числов вид. Текстовете с близко семантично значение имат вектори, които са близо един до друг, а текстове с различно значение имат вектори, които са раздалечени един от друг.

Влагащите вектори могат да се използват в широка гама от приложения, като:

1) **Търсене със заявки на естествен език:** Намиране на подходящи документи (например правни документи) в големи бази данни, чрез сравняване на векторите от заявката и на документите от базата.
2) **RAG технология: Генериране с допълнено извличане** (Retrieval-Augmented Generation – RAG) е технология, при която, при запитване към голям лингвистичен модел, се добавят релевантни данни от локални база данни, като контекст на заявката.
3) **Групиране и категоризиране:** Групиране на подобни текстове заедно, като се идентифицират тенденции и теми във данните.
4) **Класификация:** Автоматично категоризиране на текст въз основа на съдържанието му, като анализ на настроението или откриване на спам.
5) **Подобие на текст:** Идентифициране на дублирано съдържание, позволявайки задачи като премахване на дублиране на уеб страница или откриване на плагиатство.

Публичните сървъри, обслужващи векторни модели, имат програмни интерфейси (API) със специфични команди, различни от програмните интерфейси при моделите, предназначени за чат.

Тук са разгледани само първите два типа приложения на векторите за влагане, които могат да се използват в почти всякакви информационни системи на малки и средни фирми:

1) **Търсене в бази данни със заявки на естествен език.**
2) **RAG технологията**, която позволява универсалните чатботове да работят в контекста на документите на компанията.

Моделите за влагане на вектори на текстове ще генерират вектори, но те трябва някъде да бъдат съхранени

някъде, за да могат да се обработват и използват.

За съхранение и обработка на векторите, информационните системи използват бази данни, които поддържат:

- колони тип VECTOR, в които се съхраняват влагащите вектори.
- функции с вектори, основно функцията за определяне на дистанцията между два вектора.

Базите данни с такова разширение на функционалността често се наричат „векторни бази данни“, макар че те не съхраняват само вектори, а всички досегашни типове данни плюс вектори.

2.3. Семантична интерпретация на модела

Измеренията в моделите при класическите задачи за класификация, например KNN имат ясна семантична интерпретация.

Езиковите модели за влагащи вектори в LLM се различават от моделите в класическите задачи за класификация, в два аспекта.

Първата разлика е количествена: векторното пространство на LLM има много голям брой измерения. Например моделът **gemini-embedding-exp-03-07** (Gemini Api docs, 2025) създава вектори с 3072 измерения.

Втората разлика е качествена: ролята на измеренията е само да разграничават клъстери от подобни текстове. Влаганията „улавят“ семантичното значение на текстовете в едно многомерно пространство, но нито едно измерение няма изрично дефинирано семантично значение. Семантичната интерпретация на приликите в LLM изисква допълнителна разяснителна работа. Процесът на обяснение трябва да трансформира клъстерите от вектори в смислено дефинирани групи от документи. Естествен начин за изследване на семантичните представяния на набор от документи е да бъдат проектирани в пространства с по-ниски измерения (2D или 3D) и да бъде визуализирано мястото на документите в тези проекции.

Ето накратко два метода за намаляване на размерността и семантична интерпретация – **UMAT** и **HDBSCAN**.

UMAP (*Uniform Manifold Approximation and Projection for Dimension Reduction*) е метод, базиран на изчисляване на k-най-близки съседи, и последващо представяне на обектите в модел с по-малък брой димензии, който се оптимизира така че близостта между обектите да е структурно подобна на тази в модела с по-висока размерност.

HDBSCAN (*Hierarchical Density-Based Clustering*). HDBSCAN е метод за клъстеризиране, базиран на плътност на разпределение на обектите, т.е. определяне на клъстери от близко разположени обекти. Този метод е по-ефективен за разграничаване на групи обекти, в сравнение с метода на k-най-близки съседи. Недостатък на метода е, че с увеличаването на размерността на модела, клъстерирането се затруднява.

Двата метода могат да се използват последователно. Например, при анализ с използване на модел, обхващащ

статии в обществено хранилище на данни, които са индексирани от голям езиков модел с 1024 димензии, стъпките са следните:

Стъпка 1. Редуциране на модел с 1024 димензии до модел с 10 димензии по метода UMAP.

Стъпка 2. Резултатният модел с 10 димензии се преобразува по метода HDBSCAN до двумерен модел с клъстери, които могат да бъдат интерпретирани семантично като тематични раздели, близки до тези, използвани в една класическа библиотечна система.

3. Методи за изчисляване на разстояния между вектори

За да се извърши семантично търсене във векторните бази данни, трябва да се разгледат два математически метода за изчисляване на разстояния между вектори, най-често използвани в семантичното търсене:

- Евклидово разстояние между вектори
- Косинус разстояние между вектори

3.1. Евклидово разстояние между вектори

Евклидовото разстояние е основен метод при изчисляване на k-най-близки съседи (kNN). Евклидовото разстояние между два вектора с общо начало в нулевата точка, е дължината на най-късата линия, свързваща двете крайни точки на векторите (фигура 2).

Евклидовото разстояние между два вектора в N-мерното пространство е:

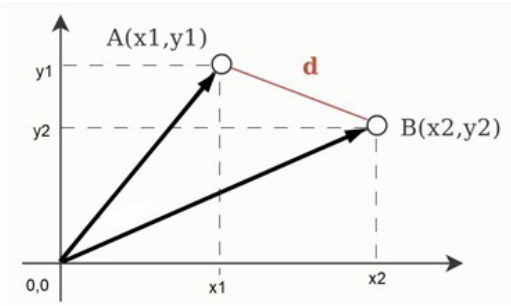
$$d(p,q) = \sqrt{[\ (p_1-q_1)^2 + (p_2-q_2)^2 \ \dots + \ (p_n - q_n)^2\ ]}$$

3.2. Косинусово разстояние между вектори

Косинусово сходство

Косинусово сходство между вектори е измерител, който се използва основно при семантичното търсене на текст.

Този измерител се основава на няколко характеристики от косинусовата функция (фигура 3).



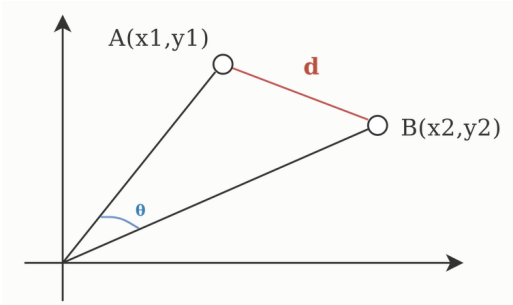
От Питагорова теорема:

$$d(A,B)^2 = (x1--x2)^2 + (y1-y2)^2$$

Евклидовото разстояние между A и B е

$$d(A,B) = \sqrt{[\ (x1-x2)^2 + (y1-y2)^2\ ]}$$

Фигура 2. *Евклидовото разстояние между два вектора*



Косинусовото сходство се измерва чрез косинуса на ъгъла между два вектора:
Косинус от 1 означава, че ъгълът е нула и векторите съвпадат перфектно,
Косинус от 0 означава, че векторите са ортогонални и се считат за напълно различни.
Косинус от -1 означава, че векторите са противоположни по посока (ъгъл от 180 градуса).

Фигура 3. *Косинусовото сходство*

Косинусово разстояние

Косинусовото разстояние между вектори е допълнение към косинусовото сходство между вектори, което се изчислява чрез изваждане на косинусовото сходство от 1,

косинус_разстояние(v) =
1 – косинус_сходство(v)

Косинусовото разстояние има стойност между 0 и 2, като 0 показва, че двата вектора са идентични, а 2 показва, че са напълно различни.

Косинусовото разстояние между вектори или евклидовото разстояние между вектори е по-добро при търсене в текстови документи?

Евклидовото разстояние и косинусовото разстояние отчитат по различен начин голям текст и кратък текст на една и съща тема. Например в базата данни са представени глава от книга от 60 страници за храненето на котки и есе в списание от 6 страници по тази тема; и въпросът е „С какво да храним 2-месечно коте“. Евклидовата мярка зависи от множеството думи и фрази, използвани в дългия текст, и ще отчете разстоянието между дългия текст и краткия текст; възможно е да бъде избран само един от тях. Косинусовото разстояние върху нормализираните вектори отчита само относителната употреба на някакъв вид думи, които определят посоката в семантичното пространство. Според косинусовата мярка разликата между дългия текст и краткия текст може да бъде много малка и двата текста ще бъдат оценени като еднакво подходящи за отговор на заявката.

3.3. Нормализация на вектори

Разстоянието на вектор A от нулевата точка представлява норма на вектор A. Векторната норма, изчислена като евклидово разстояние, е L2 норма.

Нормализацията на вектор означава коригиране на дължината му до 1, като същевременно се запазва посоката му.

Полученият вектор се нарича единичен вектор. За да се изчисли нормализираният вектор, стойностите координатите на вектора v по димензии, които означаваме v1, v2, …vn, се разделят на дължината на вектора (L)

v_normalized = (v1/L, v2/L, ... vn/L)

В пространството тримерното всички вектори с единична дължина лежат на повърхността на единичната сфера. В резултат на нормализацията, единичният вектор с дължина 1, представя точно пропорциите на стойностите на координатите, а не номиналните стойности. Косинусовото разстояние се използва главно с нормализирани вектори.

3.4. Методи за приблизително търсене

Търсене „k-най-близки съседи“ (kNN) може да бъде реализиран по два начина:

- със стандартния алгоритъм за kNN, означаван като **точно** търсене **ENN (***exact nearest neighbors*).
- с алгоритъм за **приблизително** търсене на k-най-близки съседи **ANN**, (*approximate nearest neighbors*).

Базите данни, поддържащи колони тип вектор и векторни операции, се наричат векторни бази данни. Векторните бази данни използват както точно (ENN) търсене, така и приблизително (ANN) търсене. Приблизителните алгоритми работят по-бързо от точния алгоритъм, но изборът не е 100% правилен.

Метод HNSW (Hierarchical Navigable Small Worlds) е един от най-популярните методи за приблизително търсене. Векторните индекси в **Microsoft SQL Server 2025 и Oracle database 23ai** са реализирани на база на този метод. HNSW търсенето се основава на създаването на многослоен граф-модел за индексиране на данни. Всеки влагащ вектор е проектиран на най-ниския слой. На по-високите слоеве в графа точките от данни са групирани въз основа на сходството, като се намалява експоненциално броят на точките от данни във всеки слой. Във всеки слой точките са свързани въз основа на тяхното сходство. Точките от данни във всеки слой също са свързани с тези в следващия слой.

4. Семантично търсене, базирано на езикови модели за влагащи вектори за текст

Последователност от стъпки при семантичното търсене

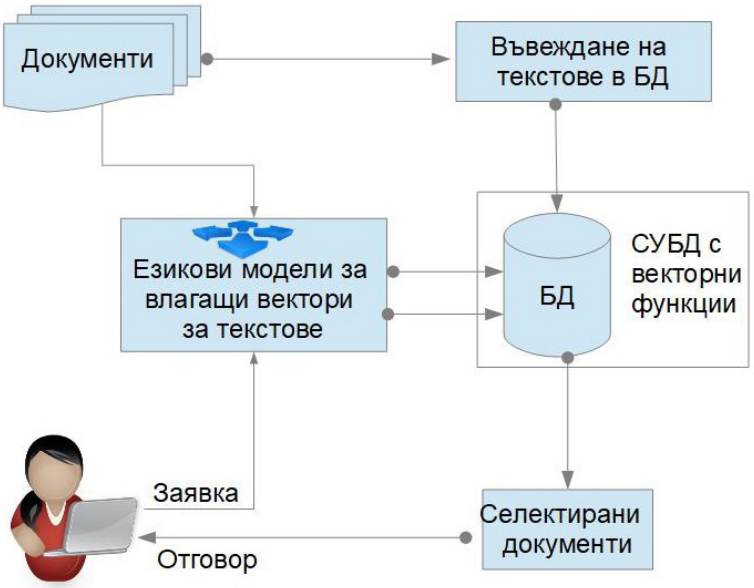
Семантичното търсене надхвърля възможностите, които има традиционното търсенето с ключови думи. Информацията за връзките между думи, синоними и обекти, съхранявана в LLM, позволява да се интерпретира заявката на потребителя, формулирана на естествен език.

На фигура 4 е представена последователността от стъпки при семантичното търсене.

Процесът на векторно-базирано семантично търсене в база данни има две фази:

Подготвителна фаза: Потребителят въвежда текст на заявка. Езиковият модел за влагащи вектори генерира влагащ вектор за заявката.

Семантично търсене: Векторът на заявката се сравнява с влагащите вектори на документите. Когато се използва KNN методът, резултатът



Фигура 4. *Последователност от стъпки при семантичното търсене*

е селектиране на k-броя документи с най-висок резултат за сходство спрямо заявката.

5. Разширено извличане с добавяне на знания (RAG)

5.1. Цел на RAG подхода

Досега беше разгледано как се използва семантичното търсене, в случай че целта е да се намери списък от документи в корпоративна база данни, съответстващи на заявка на естествен език.

Разширеното извличане с добавяне на знания (RAG) е подход, при който семантичното търсене в базата данни се използва за други цели. Типичен случай на използване в RAG е ситуация, когато потребителят иска да изпрати заявка до публичен голям генеративен модел като Gemini 2.0, DeepSeek 3.1, GTP-4, с познания в почти всички области на човешкото познание, преструвайки се, че е общ изкуствен интелект (AGI – Artificial General Intelligence). Но потребителят иска отговорът да отчете някои специфични знания от документите, които са налични в корпоративната база данни. Тези документи не са включени в обучението на публичните генеративни модели.

Това могат да бъдат документи, които съдържат частни данни на компанията, като договори, технологична информация и ценови листи, които не трябва да бъдат публично достояние. Следователно потребителят не може да изпрати въпроса си директно до чатбот на Gemini 2.0, DeepSeek 3.1, GTP-4. При RAG подхода, заедно с

въпроса, до големия езиков модел се изпращат някои съответни корпоративни документи, наречени „контекст“ на въпроса. Контекстът може да бъде текст до 1 милион токена (токън – малки части от текста – фрази, думи или няколко символа, които моделът обработва като елементи за сравняване на текстове), т.е. дори някои много дълги документи. Специално внимание се обръща на факта, че контекстът се записва на LLM сървъра във временна контекстна памет, която се използва поверително за въпроси от една компания (едно API потребителско име и ключова дума). След диалоговия прозорец временната контекстна памет се изтрива от LLM сървъра и доставчикът не трябва да запазва копия на тези данни, нито да ги използва за обучение на LLM модела.

5.2. Последователност от стъпки при използване на RAG

При RAG подхода приложението обработва потребителската заявка в три стъпки:

1) Потребителската заявка се изпраща към модела за влагащи вектори и моделът генерира вектор за заявката.
2) От корпоративната векторна база данни се избират съответните документи и се използват като контекст на въпроса.
3) Въпросът, заедно с контекста се изпраща на публичен диалогов модел с общо предназначение.
4) Публичният диалогов модел дава отговор на заявката, основан на контекста и общите знания на модела.

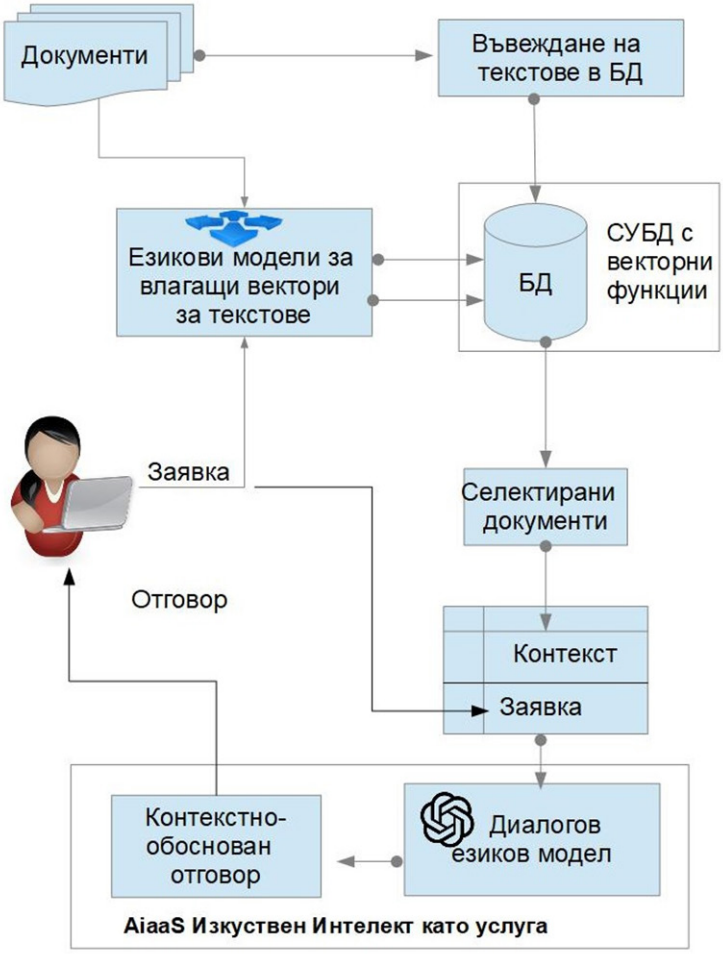
Езиковият модел за влагащи вектори и диалоговият езиков модел са независими; те могат да се използват като услуги от различни доставчици (фигура 5).

5.3. Предимства при използване на RAG подход

RAG подходът носи няколко предимства за фирмите и организациите, които искат да използват генеративни модели, като същевременно се справя с много от предизвикателствата на големите езикови модели.

- **Актуална информация.** Подходът, базиран на извличане, гарантира, че моделите имат достъп до най-актуалните и надеждни факти.
- **Контрол върху източниците на знания.** RAG също така дава на фирмите и организациите по-голям контрол върху знанията, които LLM използват, за да генерират своите отговори. Вместо да се разчита на огромните общи познания от данни за обучение, всъщност LLM генерира отговори от проверени външни източници.
- **Проверка на точността.** Потребителите имат представа за източниците на модела, за да правят препратки и да проверяват за точност на отговора.

По-ниска вероятност от халюцинации и нарушения на поверителността. Тъй като LLM се основава на фактическо знание, достъпно за потребителя, моделът не зависи толкова от извличането на информация от външни източници. В резултат на това има по-малка вероятност от халюцинации,



Фигура 5. *Последователност от стъпки при използване на RAG*

изтичане на чувствителни данни и подвеждаща информация (Velkova, 2023).

Динамични актуализации на знанията. RAG елиминира необходимостта от повторно обучение на модел върху нови данни и актуализиране на неговите параметри. Вместо това вътрешният източник се поддържа актуален, така че системата за извличане да може да търси и предоставя на LLM актуална и подходяща информация за генериране на отговори.

5.4. Сравнителна оценка на LLMs

Когато LLM се предоставят като услуга, както всеки друг вид услуга, те трябва да бъдат оценени в два аспекта: производителност и цена. Съществуват няколко популярни показателя за ефективност, които могат да се използват при оценка на моделите.

Таблица 2. *Оценка на големи езикови модели по 6 критерия*

Benchmark	DeepSeek V3	Qwen 2.5	Llama 3.1 405B	Claude 3.5	GPT 4o
MMLU	88.5	85.3	88.6	88.3	87.2
DROP	91.6	76.7	88.7	88.3	83.7
IFEval	86.1	84.1	86.0	86.5	84.3
GPQA	59.1	49.0	51.1	65.0	49.9
SimpleQA (Correct)	24.9	9.1	17.1	28.4	38.2
LongBench v2	48.7	39.4	36.1	41.0	48.1

Адаптирано от източник: *(Ertugrul Portakal, 2025)*

Таблица 2 представя някои популярни LLM, оценени по шест критерия.

MMLU (Massive Multitask Language Understanding) е критерий, предназначен да оцени многозадачните възможности на езиковите модели в различни теми. Той обхваща 57 задачи, обхващащи теми като хуманитарни науки, социални науки и други, с въпроси, вариращи от начално до професионално ниво. MMLU оценява способността на модела да обобщава знанията в голямо разнообразие от области, което го прави цялостен тест за напреднали езикови модели. Високата производителност на MMLU показва способността на модела да се справя със сложни проблеми от реалния свят, които изискват широки познания и разбиране на контекста (Massive Multitask Language Understanding: The Benchmark for Multitask AI Models, 2025).

DROP (Discrete Reasoning Over Paragraphs) е критерий, предназначен да оцени усъвършенстваните възможности за разсъждение на езиковите модели чрез сложни задачи с отговори на въпроси. Той обхваща над 9500 сложни предизвикателства, които изискват числени манипулации, многоетапно разсъждение и интерпретация на текстови данни (DROP, 2025).

GPQA (Graduate-Level Google-Proof Q&A). Този критерий съдържа набор от специализирани данни – 448 въпроса с няколко отговора, написани от експерти в областта на биологията, физиката и химията. Въпросите са висококачествени и изключително трудни: експерти, които имат или следват докторска степен в съответните области, достигат 65% точност (David Rein, 2025).

IFEval (Instruction-Following Evaluation) е ясен и лесен за възпроизвеждане критерий за оценка. Той се фокусира върху набор от „проверяеми инструкции“ като „пишете с повече от 400 думи“ и „споменете ключовата дума на AI поне 3 пъти“ (Jeffrey Zhou, 2025).

SimpleQA е критерий, предложен от OpenAI, при който всеки отговор се оценява като „правилен“, „неправилен“ или „без опит“ (Jason Wei, 2025).

LongBench v2 е критерий, предназначен да оцени способността на LLMs да се справят с проблеми с дълъг контекст, изискващи задълбочено разбиране и разсъждение в реалния свят. LongBench v2 има следните характеристики: (1) Дължина: Дължината на контекста варира от 8k до 2M думи, като повечето са под 128k. (2) Трудност: Достатъчно предизвикателство, че дори човешки експерти, използващи инструменти за търсене в документа, не могат да отговорят правилно за кратко време. (3) Покритие: Покриват различни реалистични сценарии. (4) Надеждност: Всичко във формат на въпрос с множество отговори за надеждна оценка (LongBench v2: Towards Deeper Understanding and Reasoning on Realistic Long-context Multitasks, 2025).

6. Заключение

Освен разгледаните технологии, фирмата може да използва и други инструменти, с които да защити и използва корпоративната информация. Например могат да се използват системи за разпознаване на наименувани обекти NER (Named Entity Recognition), които четат текстовите документи, намират чувствителната информация PII (Personally Identifiable Information) и я заменят с фиктивни стойности и др. подобни (Belev, 2023), (Mihova, 2023). Като чувствителна информация се считат всички данни, които фирмата държи и обработва, но не притежава, като: кредитна карта; дата на раждане; шофьорска книжка; пълно име; пол; пощенски адрес; паспортна информация; място на раждане; раса; религия; пощенски код и др. подобни.

В тази разработка фокусът е върху Големите езикови модели (LLM), създавани от водещите разработчици на Изкуствен Интелект, които се предлагат като платена абонаментна услуга. Разгледахме два начина на използване на LLM за добавяне на нови възможности в информационните системи на малки и средни фирми, които не изискват големи разходи – добавяне на семантично търсене на документи и добавяне на адаптиран диалог към публичен генеративен ИИ с обща сфера на приложимост, като ChatGPT, Gemini, Meta AI.

Семантичното търсене позволява да се откриват релевантни документи в корпоративна база данни, съответстващи на заявка, зададена на естествен език.

За добавяне на адаптиран диалог към публичен генеративен ИИ се използва методът RAG. При този метод към запитването на потребителя се добавя и фирмена информация по конфиденциални правила, без тя да става публично известна.

Тези два метода позволяват на малките и средните фирми да използват за свои цели възможностите на големите езикови модели с общо предназначение, без да инвестират средства в закупуване и обучение на модели, и без необходимост от специално обучение на персонала.

Цитирани източници (References):

1. Belev, I. (2023). Object-Oriented Ecm Approach. Innovative Information Technologies for Economy Digitalization (IITED), University of National and World Economy, Sofia, Bulgaria, issue 1, pp. 7-14.
2. DROP (2025). Извлечено от DeepEval: https://docs.confident-ai.com/docs/benchmarks-drop
   (DROP (2025). Izvlecheno ot DeepEval: https://docs.confident-ai.com/docs/benchmarks-drop)
3. Gemini Api docs (2025). Извлечено от Google AI for developers: https://ai.google.dev/gemini-api/docs/embeddings
   (Gemini Api docs (2025). Izvlecheno ot Google AI for developers: https://ai.google.dev/gemini-api/docs/embeddings)
4. Hristov, G. (2024). Improving the Quality of Financial Information through Machine Learning. *Economic Alternatives,* pp. 529-540.
5. LongBench v2: Towards Deeper Understanding and Reasoning on Realistic Long-context Multitasks (2025). Извлечено от github: https://github.com/THUDM/LongBench
   (LongBench v2: Towards Deeper Understanding and Reasoning on Realistic Long-context Multitasks (2025). Izvlecheno ot github: https://github.com/THUDM/LongBench)
6. Massive Multitask Language Understanding: The Benchmark for Multitask AI Models. (2025). Извлечено от zilliz: https://zilliz.com/glossary/mmlu-benchmark
   (Massive Multitask Language Understanding: The Benchmark for Multitask AI Models. (2025). Izvlecheno ot zilliz: https://zilliz.com/glossary/mmlu-benchmark)
7. meta-llama/Llama-3.1-405B-Instruct (2025). Retrieved from Hugging Face: https://huggingface.co/meta-llama/Llama-3.1-405B-Instruct
8. Mihova, V. (2023). Expanding the Possibilities of Information System for Working with Voice. Innovative Information Technologies for Economy Digitalization (IITED), University of National and World Economy, Sofia, Bulgaria, pp. 152-157.
9. Portakal, E. (2025). DeepSeek v3 Review: Performance in Benchmarks & Evals. Извлечено от text.cortex: https://textcortex.com/post/deepseek-v3-review
   (Portakal, E. (2025). DeepSeek v3 Review: Performance in Benchmarks & Evals. Izvlecheno ot text.cortex: https://textcortex.com/post/deepseek-v3-review)
10. Rein, D., B.L. Hou, A.C. Stickland, J. Petty, R.Y. Pang, J. Dirani, J. Michael, S.R. Bowman (2023). GPQA: A Graduate-Level Google-Proof Q&A Benchmark. Извлечено от Cornell University: https://arxiv.org/abs/2311.12022

(Rein, D., B.L. Hou, A.C. Stickland, J. Petty, R.Y. Pang, J. Dirani, J. Michael, S.R. Bowman (2023). GPQA: A Graduate-Level Google-Proof Q&A Benchmark. Izvlecheno ot Cornell University: https://arxiv.org/abs/2311.12022)

11. Velkova, I. (2023). Reducing Advertising Fraud in Digital Marketing with Artificial Intelligence. Innovative Information Technologies for Economy Digitalization (IITED) (pp. 79-88). Sofia: University of National and World Economy, Bulgaria.
12. Wei, J., N. Karina, H.W. Chung, Y.J. Jiao, S. Papay, A. Glaese, J. Schulman, W. Fedus (2024). Measuring short-form factuality in large language models. Извлечено от Cornell University: https://arxiv.org/abs/2411.04368
(Wei, J., N. Karina, H.W. Chung, Y.J. Jiao, S. Papay, A. Glaese, J. Schulman, W. Fedus (2024). Measuring short-form factuality in large language models. Izvlecheno ot Cornell University: https://arxiv.org/abs/2411.04368)
13. Zhou, J., T. Lu, S. Mishra, S. Brahma, S. Basu, Y. Luan, D. Zhou, L. Hou (2023). Instruction-Following Evaluation for Large Language Models. Извлечено от Cornell University: https://arxiv.org/abs/2311.07911
(Zhou, J., T. Lu, S. Mishra, S. Brahma, S. Basu, Y. Luan, D. Zhou, L. Hou (2023). Instruction-Following Evaluation for Large Language Models. Izvlecheno ot Cornell University: https://arxiv.org/abs/2311.07911)

Izkustven intelekt kato usluga

Vanya Lazarova

Artificial Intelligence as a Service

Vanya Lazarova

Abstract: This paper attempts to answer the question of how small and medium-sized companies can use artificial intelligence as a service (AIaaS), including context with their company data, without this leading to publicity and access to sensitive company information. Technologies that would support the construction of such a query system are examined, with each query being accompanied by additional company information or “query context”.

Key words: LLM, embedding models, Retrieval-Augmented Generation, RAG, vector databases.

JEL: C3, C6.