

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інформаційних технологій

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«ІНТЕГРАЦІЯ LLM-МОДЕЛІ GEMINI В РОЗРОБКУ ЧАТ-БОТА
ДЛЯ ВИВЧЕННЯ АНГЛІЙСЬКОЇ МОВИ»**

Рівень «магістр»

Галузь знань 12 «Інформаційні технології»

Спеціальність 122 «Комп'ютерні науки»

Виконав здобувач 2 курсу

Алі Амір Фалляхович

Керівник: к.т.н., доцент

Трофименко Олена Григорівна

Рецензент: к.т.н., доцент

Манаків Сергій Юрійович

Одеса – 2025

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ”

Факультет кібербезпеки та інформаційних технологій

Кафедра кібербезпеки/інформаційних технологій

Галузь знань: 12 Інформаційні технології

Спеціальність: 122 Комп'ютерні науки

Назва освітньої програми: Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри інформаційних технологій
доцент Наталія Логінова

_____ “ _____ ” _____ 2025 року

ЗАВДАННЯ

на кваліфікаційну (магістерську) роботу
здобувачу Алі Аміру Фалляховичу

- Тема роботи: «Інтеграція LLM-моделі Gemini в розробку чат-бота для вивчення англійської мови»
Керівник роботи: к.т.н, доцент Трофименко О.Г.
затверджені наказом НУ «ОЮА» від «10» жовтня 2025 року № 3182-06.
- Строк подання здобувачем роботи «25» листопада 2025 року.
- Дата видачі завдання «02» червня 2025 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1.	Вибір теми, вивчення літературних джерел та складання плану роботи	15.09.2025	
2.	Підготовка першого розділу роботи та подання його керівнику	01.10.2025	
3.	Підготовка другого розділу роботи та подання його керівнику	14.10.2025	
4.	Підготовка третього розділу роботи та подання його керівнику	01.11.2025	
5.	Підготовка вступу та висновків	07.11.2025	
6.	Доопрацювання роботи з урахуванням зауважень керівника	17.11.2025	
7.	Подача електронного варіанта роботи для перевірки на плагіат	20.11.2025	

Здобувач

_____ А.Ф. Алі
(підпис) (ім'я та прізвище)

Керівник роботи

_____ О.Г. Трофименко
(підпис) (ім'я та прізвище)

АНОТАЦІЯ

Кваліфікаційна робота на тему «Інтеграція LLM-моделі Gemini в розробку чат-бота для вивчення англійської мови» на здобуття другого (магістерського) рівня вищої освіти за спеціальністю 122 «Комп'ютерні науки», освітньою програмою «Комп'ютерні науки». Робота містить 12 рисунків, 7 таблиць, 4 додатки, 42 літературні джерела. Робота виконана на 53 сторінках основного тексту і 62 сторінках загального тексту.

Метою роботи є дослідження та реалізація інтеграції великої мовної моделі Gemini у багаторівневу архітектуру освітнього чат-бота з підтримкою когнітивно-орієнтованих форматів навчання, а також застосування формалізованих метрик NLP для оцінки якості генерації навчального контенту.

Об'єктом дослідження є процес інтерактивної мовної взаємодії в освітньому середовищі.

Предметом дослідження є методи та засоби реалізації когнітивно-орієнтованого чат-бота з використанням LLM.

У межах цієї мети досліджено теоретичні засади застосування LLM-моделей у мовній освіті, розроблено формалізовані навчальні сценарії, побудовано архітектуру інтерактивної системи, обґрунтовано вибір технологічного стека, реалізовано модулі генерації навчального контенту та проведено тестування якості генерації за метриками NLP.

Ключові слова: LLM, Gemini, чат-бот, англійська мова, NLP, когнітивне навчання, генерація тексту, освітні технології, Python, мультимодальність, адаптивність, інтерактивне навчання.

ABSTRACT

Qualification work on the topic "Integrating the Gemini LLM model into the development of a chatbot for learning English" for obtaining the second (master's) level of higher education in the specialty 122 "Computer Science", educational program "Computer Science". The work contains 12 figures, 7 tables, 4 appendices, 42 references. The work is performed on 53 pages of the main text and 62 pages of the general text.

The purpose of this study is to substantiate and implement the integration of the Gemini large language model into a multi-level architecture of an educational chatbot that supports cognitively oriented learning formats. The research also involves applying formalized NLP metrics to evaluate the quality of generated educational content.

The object of the research is the process of interactive language-based communication in educational environments.

The subject of the research is the methods and tools for implementing a cognitively oriented chatbot using large language models.

To achieve this goal, the theoretical foundations of LLM-based language education were examined, learning scenarios were formalized, the system architecture was designed, the technological stack was justified, content generation modules were implemented, and the quality of generated outputs was assessed using NLP metrics.

Keywords: LLM, Gemini, chatbot, English, NLP, cognitive learning, text generation, educational technologies, Python, multimodality, adaptability, interactive learning.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	7
ВСТУП.....	8
1 ОСОБЛИВОСТІ ВИКОРИСТАННЯ LLM-МОДЕЛЕЙ	
У МОВНОМУ НАВЧАННІ	11
1.1 Великі мовні моделі: архітектура, принципи роботи та порівняння	11
1.2 Сучасні підходи до інтерактивного навчання англійської мови в контексті використання LLM-моделей.....	14
1.3 ІІІ як рушій персоналізованого навчання: адаптивність, когнітивна підтримка та мотиваційна динаміка у контексті LLM-моделей	16
1.4 Психолінгвістичні та дидактичні аспекти використання чат-ботів у навчанні	18
1.5 Аналіз переваг і ризиків інтеграції LLM у освітні платформи	21
1.6 Висновки до розділу 1	24
2 ПРОЄКТУВАННЯ ЧАТ-БОТА З ІНТЕГРАЦІЄЮ МОДЕЛІ GEMINI	26
2.1 Вибір платформи, мови програмування та середовища розробки.....	26
2.2 Архітектура чат-бота: модулі, API, обробка запитів, логіка діалогу	28
2.3 Інтеграція Gemini	30
2.4 Висновки до розділу 2	33
3 РЕАЛІЗАЦІЯ ЧАТ-БОТА З ІНТЕГРАЦІЄЮ МОДЕЛІ GEMINI	35
3.1 Формати навчального контенту: словникові вправи, граматичні пояснення, симуляція діалогів	35
3.2 Тестування функціональності: юзабіліті, швидкодія, точність відповідей	38
3.3 Використання метрик NLP для оцінки якості генерації.....	42
3.4 Висновки до розділу 3	46

ВИСНОВКИ.....	47
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	49
ДОДАТКИ	54
Додаток А. Технічна специфікація запитів та відповідей Gemini API для трьох навчальних сценаріїв	54
Додаток Б. Python-код генерації JSON-запиту до Gemini API	56
Додаток В. Фрагменти Python-коду модуля використання Gemini API у чат-боті для вивчення англійської мови	59
Додаток Г. Копії документів про апробацію результатів роботи.....	62

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

API – Application Programming Interface

BLEU – Bilingual Evaluation Understudy

CUS – Chatbot Usability Scale

DL – Deep Learning

GCP – Google Cloud Platform

LLM – Large Language Models

NLG – Natural Language Generation

NLP – Neuro-linguistic programming

ROUGE – Recall-Oriented Understudy for Gisting Evaluation

ПЗ – програмне забезпечення

ВСТУП

У сучасному освітньому середовищі стрімкий розвиток великих мовних моделей (LLM) відкриває нові можливості для персоналізованого навчання, когнітивної підтримки та автоматизованої рефлексії. Моделі типу GPT-4, Claude 4 та Gemini 2.5 демонструють здатність до контекстуальної генерації, стилістичної адаптації та інтерактивної взаємодії, що робить їх придатними для реалізації освітніх чат-ботів. Особливої уваги заслуговує модель Gemini, розроблена Google DeepMind, яка поєднує мультимодальну обробку, підтримку довгих контекстів та оптимізацію для освітніх платформ. Її інтеграція в архітектуру чат-бота дозволяє реалізувати гнучкі навчальні сценарії, включно з лексичними вправами, граматичними поясненнями та симуляцією діалогів.

Метою роботи є дослідження та реалізація інтеграції великої мовної моделі Gemini у багаторівневу архітектуру освітнього чат-бота з підтримкою когнітивно-орієнтованих форматів навчання, а також застосування формалізованих метрик NLP для оцінки якості генерації навчального контенту.

Об'єктом дослідження є процес інтерактивної мовної взаємодії в освітньому середовищі.

Предметом дослідження є методи та засоби реалізації когнітивно-орієнтованого чат-бота з використанням LLM.

Для досягнення цієї мети поставлено такі завдання:

- виконати порівняльний аналіз сучасних LLM-моделей для освітнього застосування;
- обґрунтовано вибрати модель LLM для інтеграції в чат-бот;
- розробити архітектуру багатомодульного освітнього чат-бота з підтримкою API;
- реалізувати навчальні сценарії: словникові вправи, граматичні пояснення, симуляція діалогів;
- сформувати JSON-запити до LLM з урахуванням параметрів генерації та безпеки;

- побудувати FastAPI-сервер з контролерами для кожного навчального формату;
- виконати тестування функціональності: юзабіліті, швидкодія, точність генерації.

У роботі використано комплекс методів наукового дослідження, зокрема:

- метод системного аналізу застосовано для формалізації архітектури чат-бота, визначення функціональних компонентів, логіки обробки запитів та взаємодії з LLM-моделлю;
- порівняльно-аналітичний метод використано для оцінки сучасних LLM-моделей (GPT-4, Claude, Gemini) за різними критеріями;
- метод моделювання реалізовано у вигляді UML-діаграм, JSON-структур запитів та формалізованих математичних моделей діалогової логіки, які описують когнітивні сценарії навчання;
- метод експерименту застосовано для тестування функціональності чат-бота, включно з юзабіліті, швидкодією, точністю генерації та відповідністю навчальним цілям;
- метод кількісного аналізу використано при застосуванні метрик NLP (BLEU, ROUGE, Perplexity, BERTScore, MoverScore) для формалізованої оцінки якості згенерованого контенту;
- метод інтерпретації результатів забезпечив узагальнення емпіричних даних, формулювання висновків щодо ефективності інтеграції LLM у освітнє середовище та рекомендацій щодо подальшого застосування.

Поєднання цих методів забезпечило наукову обґрунтованість, структурну повноту та прикладну релевантність дослідження.

У межах цієї мети досліджено теоретичні засади застосування LLM-моделей у мовній освіті, розроблено формалізовані навчальні сценарії, побудовано архітектуру інтерактивної системи, обґрунтовано вибір технологічного стека, реалізовано модулі генерації навчального контенту та проведено тестування якості генерації за метриками NLP.

Наукова новизна роботи полягає в інтеграції великої мовної моделі Gemini 1.5 Pro у багаторівневу архітектуру освітнього чат-бота з її адаптацією до трьох когнітивно-орієнтованих навчальних форматів (лексика, граматика, діалог). Формалізація навчальних сценаріїв у вигляді JSON-запитів з параметрами генерації дозволяє не лише використовувати LLM, а й реплікувати, масштабувати та верифікувати її поведінку. У роботі проведено тестування функціональності та оцінку якості відповідей чат-бота за метриками NLP, які зазвичай застосовуються в машинному перекладі або NLG, але не в освітніх чат-ботах. Тут ці метрики (BLEU, ROUGE, Perplexity, BERTScore та MoverScore) використано для порівняння якості генерації у трьох когнітивних сценаріях, що дозволяє зробити висновки про ефективність кожного формату

Практична цінність полягає у створенні модульної, масштабованої та безпечної системи, придатної до впровадження в освітні платформи з підтримкою адаптивного навчання. Робота базується на сучасних дослідженнях у галузі штучного інтелекту, когнітивної педагогіки та мовної освіти, що забезпечує її теоретичну обґрунтованість і прикладну релевантність.

Публікації з апробацією кваліфікаційної роботи:

1. Трофименко О.Г., Алі А.Ф. Переваги і ризики інтеграції LLM в освітні платформи. *Інформаційні технології: моделі, алгоритми, системи (ITMAS-2025)*: матер. VI Всеукр. наук.-практ. інтернет-конф. (16-17 листопада 2025 р.) Миколаїв. URL: <https://itconf.nuos.edu.ua/2025/publications/analysis-of-the-benefits-and-risks-of-llm-integration-into-educational-platforms/>

2. Трофименко О.Г., Алі А.Ф. Використання чат-ботів з LLM-моделями у вивченні мов. *Кібербезпека в сучасному світі: актуальні виклики* : матер. VI міжнар. наук.-практ. конф. (28 листопада 2025 р.). Одеса, 2025 (депоновано).

1 ОСОБЛИВОСТІ ВИКОРИСТАННЯ LLM-МОДЕЛЕЙ У МОВНОМУ НАВЧАННІ

1.1 Великі мовні моделі: архітектура, принципи роботи та порівняння

Великі мовні моделі (Large Language Models, LLM) становлять одну з найдинамічніших технологій сучасної обробки природної мови, що активно інтегрується в освітні платформи, зокрема у навчання іноземних мов. Їхня архітектура базується на трансформерному підході, який дозволяє моделі ефективно обробляти довгі послідовності тексту, зберігаючи контекст і семантичні зв'язки між словами [1].

Основним принципом роботи LLM є предиктивне моделювання: система навчається на великих корпусах тексту, прогнозуючи наступне слово або фразу на основі попереднього контексту. Це забезпечує здатність моделі до генерації зв'язного, стилістично узгодженого та контекстуально релевантного тексту, що є критично важливим у навчанні перекладу. Трансформерна архітектура слугує базою для побудови різних LLM (GPT-4, Claude, Gemini), що мають спільні принципи, але різну реалізацію, функціональність і фокус застосування (рис. 1.1).

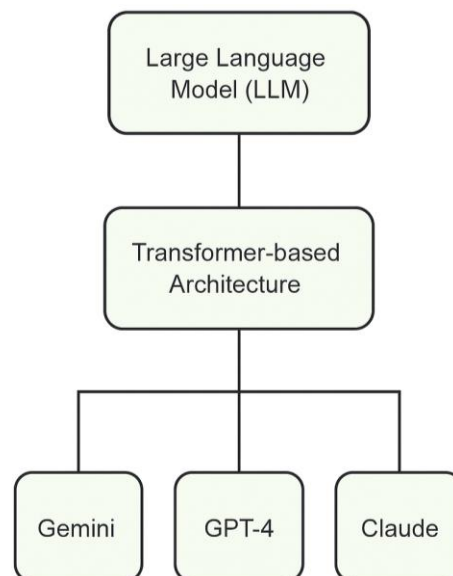


Рисунок 1.1 – Ієрархічна структура сучасних LLM, які використовуються у чат-ботах для вивчення іноземних мов

У контексті створення чат-ботів для мовного навчання, LLM-моделі демонструють здатність до адаптивної діалогової взаємодії, що дозволяє студенту отримувати миттєвий зворотний зв'язок, уточнення граматичних структур, стилістичних реєстрів та семантичних відтінків. Наприклад, GPT-4, розроблений OpenAI, вирізняється високою точністю у генерації англomовного тексту, здатністю до стилістичної адаптації та підтримки складних мовних конструкцій. За даними досліджень 2025 року [2, 3], GPT-4o демонструє стабільну продуктивність у задачах перекладу, особливо при роботі з англійсько-українськими та англійсько-іспанськими парами, при цьому зберігає контекстуальну цілісність і прагматичну адекватність.

Claude 4.5 від Anthropic, натомість, орієнтований на етичну безпеку та пояснюваність. Його архітектура оптимізована для довгих контекстів і м'якої стилістики, що робить його придатним для навчання студентів з початковим рівнем володіння мовою. Claude демонструє високу точність у поясненні граматичних помилок та стилістичних невідповідностей, що особливо корисно у навчанні перекладу з елементами метамовної рефлексії [4].

Gemini 2.5, розроблений Google DeepMind, поєднує мультимодальну обробку з високою швидкістю генерації. Його перевага полягає у здатності працювати з візуальними та текстовими даними одночасно, що відкриває перспективи для інтеграції в платформи, де навчання перекладу супроводжується ілюстративними матеріалами [5]. Gemini демонструє найвищу швидкість генерації перекладу серед трьох моделей, зберігаючи при цьому прийнятну точність і стилістичну гнучкість, особливо у неформальних реєстрах [6].

Усі три моделі – GPT-4, Claude 4 та Gemini 2.5 – базуються на трансформерній архітектурі, але відрізняються за параметрами контекстного вікна, швидкістю генерації, стилістичною гнучкістю та етичними обмеженнями (табл. 1.1). Їхнє використання у створенні лінгвістичних чат-ботів для навчання перекладу дозволяє реалізувати принципи адаптивного навчання, когнітивної підтримки та автоматизованої рефлексії. Водночас, ефективність таких систем

залежить від педагогічної фасилітації, етичного аудиту та технічної оптимізації, що має бути враховано при їх інтеграції в освітні середовища.

Таблиця 1.1 – Порівняння GPT-4, Claude 4 та Gemini 2.5 у контексті їх застосування для створення чат-ботів, орієнтованих на вивчення іноземних мов

Критерій	GPT-4 (OpenAI)	Claude 4 (Anthropic)	Gemini 2.5 (Google DeepMind)
Архітектура	Трансформер, оптимізований для генерації тексту з високою контекстуальною точністю	Трансформер з акцентом на етичну безпеку та довгі контексти	Мультиmodalний трансформер з підтримкою тексту, зображень і коду
Контекстне вікно	До 128k tokenів (у GPT-4o)	До 200k tokenів	До 1M tokenів у Gemini 1.5 Pro
Сильні сторони	Стилістична гнучкість, точність перекладу, підтримка складних мовних конструкцій	Пояснювальність, етична стабільність, м'яка стилістика	Швидкість генерації, мультиmodalність, ефективність у неформальних регістрах
Слабкі сторони	Може генерувати переконливо хибні відповіді без пояснень	Обмежена стилістична варіативність, менш агресивна генерація	Менша точність у складних граматичних структурах порівняно з GPT-4
Придатність для навчання перекладу	Висока: підтримує стилістичну адаптацію, граматичну точність	Висока: пояснює помилки, коригує інтерференцію	Середня: швидкий переклад, але потребує педагогічного супроводу
Етичні обмеження	Часткова модерація, залежить від API	Строгі етичні фільтри, орієнтація на безпечну взаємодію	Вбудовані обмеження, особливо в освітніх версіях
Можливість інтеграції в чат-боти	Висока: доступні API, підтримка багатьох мов	Висока: оптимізований для діалогових систем	Висока: особливо для мультиmodalних освітніх платформ

Тим самим, великі мовні моделі не лише трансформують технології перекладу, а й створюють нову парадигму мовного навчання, де чат-боти виступають як когнітивні партнери, здатні підтримувати студента в процесі формування перекладацької компетентності, стилістичної чутливості та критичного мислення.

Виконаний у цьому підрозділі порівняльний аналіз зумовив вибір моделі Gemini 1.5 Pro для інтеграції в освітній чат-бот з огляду на її мультиmodalність, підтримку наддовгого контексту, швидкодію та етичну стабільність. Порівняно з

GPT-4 та Claude 4, Gemini демонструє кращу адаптацію до неформальних реєстрів, тобто генерує тексти у більш природній, розмовній, «людяній» манері, що важливо для діалогових сценаріїв навчання. Gemini легше імітує живу мову спілкування, зокрема: неформальні звертання (наприклад, “Hey, let’s practice!”), скорочення (don’t, gonna, kinda), емоційно забарвлені фрази (“That’s awesome!”), “Oops, try again!”), гумор, емпатію, підтримку, що важливо для освітніх чат-ботів.

1.2 Сучасні підходи до інтерактивного навчання англійської мови в контексті використання LLM-моделей

У контексті навчання іноземних мов інтерактивність дедалі частіше реалізується через чат-боти, які моделюють діалогову взаємодію між студентом і віртуальним тьютором. Традиційні чат-боти, побудовані на правилах або простих шаблонах, демонстрували обмежену гнучкість, особливо в завданнях перекладу, де важливо враховувати контекст, граматичні структури та стилістичні особливості. З появою LLM-моделей, серед яких найпотужнішими зараз є GPT-4, Claude або Gemini, стало можливим створення чат-ботів, здатних до контекстуального перекладу, адаптації стилю мовлення та навчання через діалогові сценарії, що відповідають рівню володіння мовою користувача [7].

LLM-моделі демонструють конкурентну якість перекладу навіть без спеціалізованого донавчання, завдяки здатності узагальнювати знання з різних мовних пар та адаптуватися до стилістичних вимог [8]. Це відкриває перспективи для створення освітніх чат-ботів, які не лише перекладають, а й пояснюють граматичні конструкції, пропонують альтернативні варіанти перекладу та адаптують контент до рівня користувача.

Архітектурно такі чат-боти можуть поєднувати LLM-модуль із системою відстеження прогресу, профілем учня та базою навчального контенту. Наприклад, при перекладі речення студентом, система може не лише надати правильний варіант, а й пояснити граматичні особливості, виявити типові помилки та запропонувати додаткові вправи. Це забезпечує когнітивну підтримку та

персоналізовану траєкторію навчання, що відповідає принципам адаптивного навчання.

Інтерактивність у цьому контексті набуває нових форм: студент може вести діалог з моделлю, ставити уточнюючі запитання, отримувати пояснення та навіть моделювати ситуації міжкультурної комунікації. Архітектура чат-бота з LLM-модулем, адаптована до освітнього застосування (наприклад, переклад текстів, граматичний аналіз, діалогове навчання), показана на рис. 1.2. Ця схема демонструє трирівневу архітектуру: інтерфейсний рівень (взаємодія з користувачем), серверний рівень (логіка обробки запитів, інтеграція з LLM) та рівень даних (зберігання профілів, контенту, аналітики)

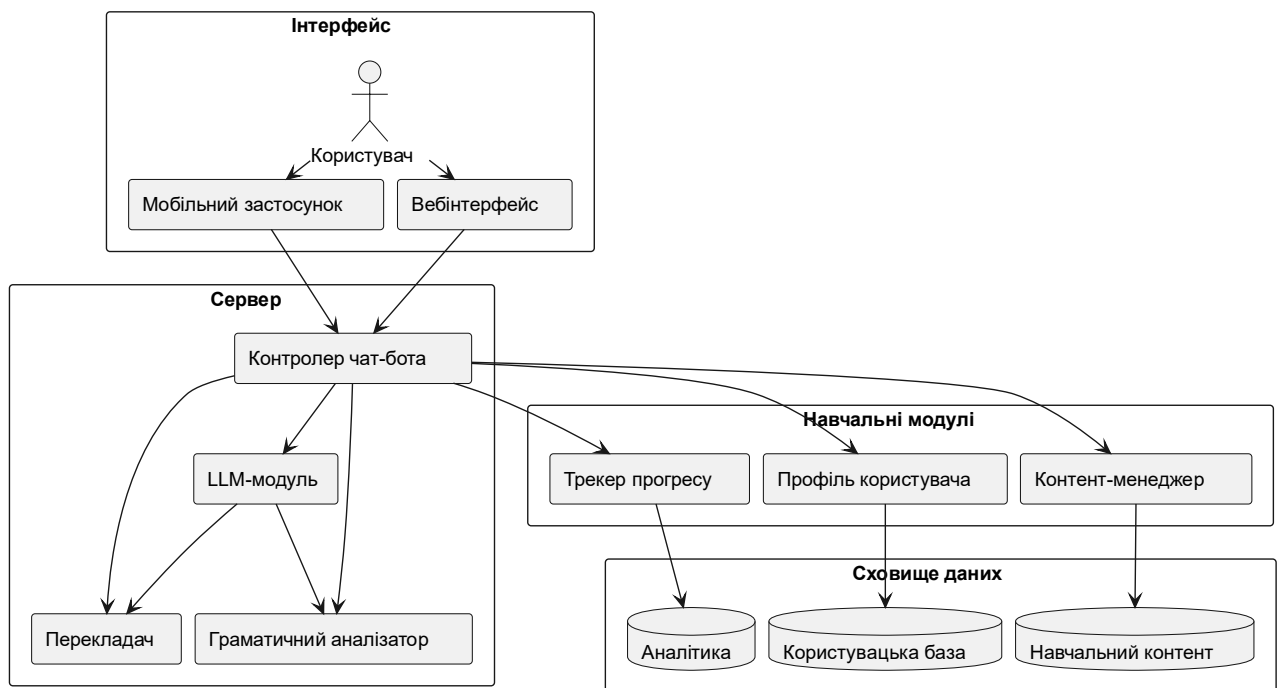


Рисунок 1.2 – Архітектура чат-бота з LLM-модулем для освітнього застосування

Завдяки трансформерній архітектурі, LLM-моделі здатні зберігати контекст попередніх реплік, що дозволяє будувати логічно зв'язні діалоги та підтримувати навчальну мотивацію. Крім того, такі системи можуть інтегрувати механізми оцінювання, наприклад, аналіз лексичної різноманітності, граматичної точності та стилістичної відповідності, що дозволяє формувати аналітичний зворотний зв'язок.

З технічного погляду, реалізація лінгвістичних чат-ботів на основі LLM потребує врахування питань продуктивності, приватності даних та етичної взаємодії. Архітектура таких систем має включати механізми фільтрації контенту, захисту персональних даних та адаптації до культурних контекстів, особливо у навчальному середовищі.

Використання LLM-моделей як ядра освітніх чат-ботів дозволяє поєднати переклад, граматичну підтримку, адаптацію контенту та аналітику в єдину систему, що забезпечує гнучке, персоналізоване та ефективне навчання. У перспективі такі системи можуть стати основою для масштабованих освітніх платформ, здатних до самонавчання, багатомовної підтримки та інтеграції з іншими освітніми інструментами.

1.3 III як рушій персоналізованого навчання: адаптивність, когнітивна підтримка та мотиваційна динаміка у контексті LLM-моделей

Адаптивність у LLM-системах реалізується через механізми динамічного узгодження навчального контенту з профілем користувача. Наприклад, використання глибокого навчання (DL) та мультимодальної аналітики дозволяє системам адаптувати складність завдань, типи вправ та стилі подання інформації в реальному часі. Формально, адаптація може бути описана як функція [9]:

$$A(u, c, t) = \arg \max_{x \in C} [\mu(x, u) \cdot \delta(x, t)]$$

де $A(u, c, t)$ – адаптований контент для користувача u , з урахуванням його когнітивного стану c та часу t ;

$\mu(x, u)$ – релевантність контенту x до профіля користувача;

$\delta(x, t)$ – динамічна складність контенту в момент часу t .

Такий підхід дозволяє системі формувати навчальну траєкторію, яка оптимізує когнітивне навантаження та підтримує зону найближчого розвитку.

Когнітивна підтримка реалізується через здатність LLM-моделей до пояснення, рефреймінгу та контекстуалізації знань. Наприклад, при вивченні

англійської мови студент може отримати не лише переклад слова, а й граматичне пояснення, приклади вживання, стилістичні варіанти та культурні нюанси. Це відповідає принципам когнітивної теорії мультимедійного навчання, де поєднання тексту, прикладів і пояснень сприяє глибшому засвоєнню. У LLM-системах когнітивна підтримка може бути формалізована як функція [10]:

$$S(q) = \{e_1, e_2, \dots, e_n\}$$

де $S(q)$ – набір когнітивних елементів підтримки у відповідь на запит q ;

кожен e_i – пояснення, приклад, аналогія або граматична структура, релевантна до запиту.

Ефективність такої підтримки підтверджується експериментальними даними: у дослідженні [10] студенти, які навчалися з LLM-підтримкою, демонстрували на 18% вищі результати у тестах на розуміння граматики та лексики порівняно з контрольною групою.

Мотиваційний компонент персоналізованого навчання є не менш важливим. Використання адаптивних AI-систем, таких як Squirrel AI, підвищує мотивацію студентів за рахунок гейміфікації, миттєвого зворотного зв'язку та персоналізованих викликів. У експериментальній групі ($n = 153$) рівень залученості був на 23% вищим, ніж у контрольній ($n = 160$), що свідчить про позитивний вплив адаптивних механізмів на мотиваційну динаміку [11]. Мотивацію можна моделювати як функцію:

$$M(u, t) = \alpha \cdot E(u, t) + \beta \cdot F(u, t)$$

де $M(u, t)$ – мотиваційний рівень користувача u у момент часу t ;

$E(u, t)$ – емоційна залученість;

$F(u, t)$ – частота позитивного зворотного зв'язку;

α, β – вагові коефіцієнти, що визначають вплив кожного чинника.

Отже, LLM-моделі виступають не лише як інструмент генерації тексту, а як ядро персоналізованої освітньої системи, здатної адаптуватися до потреб учня, підтримувати його когнітивно та стимулювати мотивацію. Їхня інтеграція у навчання іноземних мов дозволяє перейти від статичних курсів до динамічних,

діалогових і гнучких освітніх середовищ, що відповідають принципам сучасної педагогіки та когнітивної науки.

1.4 Психолінгвістичні та дидактичні аспекти використання чат-ботів у навчанні

У процесі вивчення іноземних мов чат-боти на основі великих мовних моделей (Large Language Model, LLM), таких як Gemini або GPT-4 [12], дедалі частіше використовуються як інструменти інтерактивного навчання, які поєднують психолінгвістичні механізми з дидактичними принципами. Їхня ефективність пояснюється здатністю моделювати діалогічну взаємодію, яка активізує внутрішнє мовлення, стимулює міжмовну рефлексію та сприяє формуванню перекладацької компетентності.

У контексті вивчення іноземної мови чат-боти, інтегровані з LLM, виконують функцію адаптивного мовного посередника. Вони здатні не лише здійснювати автоматичний переклад, а й пояснювати семантичні відтінки, граматичні трансформації та стилістичні особливості перекладених конструкцій [13]. Це особливо важливо для формування перекладацької компетентності, яка стосується не лише знання мовних відповідників, а й розуміння прагматичних і культурних контекстів. LLM-моделі здатні генерувати варіативні переклади одного й того самого висловлювання, що дозволяє студенту порівнювати стилістичні реєстри, виявляти контекстуальні залежності та формувати критичне ставлення до мовного вибору [14].

З дидактичної точки зору чат-боти на основі LLM сприяють реалізації принципів конструктивістського навчання, де знання формуються в процесі активної взаємодії з цифровим середовищем. Вони забезпечують негайний зворотний зв'язок, що є ключовим чинником у формуванні мовної компетентності, особливо в умовах дистанційного або змішаного навчання. Крім того, такі системи дозволяють реалізувати індивідуальні траєкторії навчання, адаптуючи складність завдань до рівня підготовки студента. Використання чат-ботів у вивчанні іноземних мов сприяє підвищенню мотивації студентів,

покращенню результатів у тестових завданнях на 15–20% та зменшенню рівня тривожності під час мовленнєвої практики [15].

Однак ефективність таких чат-ботів значною мірою залежить від якості мовної моделі, на якій вони побудовані. LLM-моделі здатні до генерації контекстуально релевантних відповідей, але водночас можуть продукувати некоректні або стилістично невідповідні переклади, особливо у випадках культурно маркованих або ідіоматичних конструкцій. Тому дидактичне використання таких систем має супроводжуватися педагогічною фасилітацією, яка дозволяє студенту критично осмислювати отримані результати, порівнювати їх із власними гіпотезами та формувати метамовну рефлексію.

Однією з ключових психолінгвістичних моделей, яка реалізується в чат-ботах, є *концепція внутрішнього мовлення*. У цьому контексті студент при взаємодії з ботом не просто отримує готову відповідь, а вступає в когнітивний діалог, де кожен запит – це гіпотеза, а кожна відповідь – стимул до самокорекції. Наприклад, коли студент запитує: “How do I say ‘Я не згоден’ in formal English?”, бот може відповісти: “You might say ‘I respectfully disagree’ or ‘I beg to differ’ in formal contexts.” Така відповідь не лише пропонує варіанти, а й провокує студента порівняти їх з власним уявленням про стилістичну доречність, що активізує метамовну рефлексію [16].

Інша модель, релевантна до навчання перекладу, – це *теорія мовного трансферу*, запропонована. Вона пояснює, як знання рідної мови впливають на засвоєння іноземної, часто через інтерференцію. Чат-боти, здатні виявляти кальки або граматичні помилки, відіграють роль коректора міжмовного впливу. Наприклад, студент може написати: “I have 25 years,” калькуючи з української “Мені 25 років.” Бот відповідає: “In English, age is expressed with ‘to be’: you should say ‘I am 25 years old.’” Така відповідь не лише виправляє помилку, а й пояснює граматичну розбіжність, що сприяє формуванню міжмовної компетенції.

У дидактичному аспекті чат-боти здатні реалізувати принципи когнітивного навантаження. Вони зменшують обсяг одночасно оброблюваної

інформації, надаючи пояснення в реальному часі, що дозволяє студенту зосередитися на змістовному аналізі. Наприклад, при перекладі речення “He broke the silence,” студент може отримати від бота варіанти: “Він порушив тишу” або “Він заговорив першим,” із поясненням, що вибір залежить від контексту. Це знижує когнітивне навантаження, адже студент не мусить самостійно шукати стилістичні відповідники, а може порівнювати запропоновані варіанти.

Тим самим чат-боти, інтегровані з LLM-моделями, не просто автоматизують переклад, а створюють умови для глибокого психолінгвістичного опрацювання мовного матеріалу. Вони стимулюють внутрішнє мовлення, нейтралізують інтерференцію, знижують когнітивне навантаження і водночас реалізують дидактичні принципи адаптивного навчання.

З іншого боку, використання LLM у навчанні іноземних мов супроводжується низкою проблем, які мають як технічну, так і дидактичну природу. Однією з ключових проблем є *обмежена достовірність генерованого контенту*. Моделі можуть продукувати граматично правильні, але семантично або прагматично некоректні висловлювання. Це особливо критично у навчанні перекладу, де точність і контекстуальна відповідність є визначальними. Оскільки студенти часто сприймають відповіді чат-бота як авторитетні, навіть якщо вони містять помилки, то це може призводити до закріплення хибних мовних конструкцій [17].

Іншою проблемою є *стилістична негнучкість*. Хоча LLM здатні генерувати тексти в різних регістрах, вони не завжди коректно розрізняють формальні й неформальні контексти, особливо в міжкультурному спілкуванні. Це ускладнює навчання стилістичної адаптації, яка є важливою складовою мовної компетентності. Крім того, *відсутність емоційної інтонації* в текстовій взаємодії з ботом може знижувати мотивацію студентів, особливо на початкових етапах навчання, коли важливо створити підтримувальне середовище [18].

З етичного боку, чат-боти можуть *збирати та обробляти персональні дані студентів* без їхньої повної обізнаності, що порушує принципи конфіденційності. Доцільним є балансування між технологічною підтримкою та

живим спілкуванням, оскільки надмірна автоматизація може призвести до втрати соціального компонента навчання. Крім того, без належного фасилітаційного супроводу чат-боти можуть сприяти поверхневому засвоєнню знань, а не розвитку критичного мислення [19].

Психолінгвістичні та дидактичні аспекти використання чат-ботів у вивчанні іноземних мов свідчать про їхній значний потенціал як інструментів розвитку мовної компетентності, критичного мислення та автономного навчання. Інтеграція LLM-моделей у такі системи дозволяє підвищити якість мовної взаємодії, але вимагає методичної обережності, етичної відповідальності та педагогічного супроводу. Без цього вони ризикують стати не інструментом розвитку, а джерелом фрагментарного і некритичного засвоєння мовного матеріалу.

1.5 Аналіз переваг і ризиків інтеграції LLM у освітні платформи

У сучасному освітньому середовищі інтеграція великих мовних моделей (Large Language Model, LLM) у цифрові платформи стає одним із ключових напрямів трансформації навчального процесу. Застосування таких моделей дозволяє реалізувати принципи персоналізованого навчання, автоматизувати оцінювання та забезпечити адаптивну підтримку студентів у реальному часі. Водночас стрімке впровадження LLM супроводжується низкою ризиків – від етичних викликів і проблем достовірності до технічних обмежень і загроз академічній автономії. У цьому контексті постає необхідність наукового аналізу переваг і ризиків використання LLM у навчанні, а також формулювання рекомендацій щодо їх безпечної та ефективної інтеграції в освітні платформи.

LLM, зокрема Gemini, GPT-4, Claude та PaLM, дедалі активніше інтегруються в освітні платформи, трансформуючи способи навчання, оцінювання та взаємодії між студентами й викладачами. За даними досліджень [20] LLM-системи демонструють на 23–35% вищу ефективність у формуванні когнітивних навичок порівняно з традиційними методами навчання, особливо в контексті мовної освіти та STEM-дисциплін.

Однією з ключових переваг є персоналізація навчального процесу. LLM здатні адаптувати навчальний контент до рівня знань, стилю мислення та мовної компетенції студента. Наприклад, Gemini може генерувати вправи, які враховують типові помилки конкретного користувача, а також моделювати діалоги з різними стилістичними та граматичними рівнями. Це дозволяє створити динамічне середовище навчання, де студент отримує миттєвий зворотний зв'язок, що здатно зменшити час на засвоєння матеріалу в середньому на 28% [7].

Автоматизація рутинних завдань – ще один вагомий аргумент на користь LLM. Моделі здатні генерувати тести, оцінювати відкриті відповіді, створювати резюме лекцій та адаптувати навчальні матеріали для різних рівнів складності. Це не лише знижує навантаження на викладачів, а й дозволяє зосередитися на методичній та етичній складовій навчання.

Однак, попри переваги, інтеграція LLM супроводжується низкою ризиків. Найбільш критичним є питання достовірності та етичної відповідальності. Моделі можуть генерувати некоректні або упереджені відповіді, що небезпечно в навчальному контексті. Так, за дослідженням [21] LLM демонструють 12-18% похибки при відповіді на запити з гуманітарних дисциплін, що вимагає додаткової верифікації контенту перед використанням у навчанні.

Також є ризик надмірної автоматизації, що може знизити мотивацію студентів до самостійного мислення. Якщо навчання перетворюється на пасивне споживання відповідей моделі, втрачається критичне осмислення матеріалу. Так, дослідження [20] з'ясувало, що студенти, які використовували LLM без супроводу викладача, демонстрували на 17% нижчі результати у відкритих завданнях порівняно з тими, хто працював у змішаному форматі.

Технічні обмеження також не слід ігнорувати. Моделі потребують значних обчислювальних ресурсів, стабільного інтернет-з'єднання та захищених каналів передачі даних. У контексті освітніх платформ це означає додаткові витрати на інфраструктуру та кібербезпеку. Крім того, питання конфіденційності

студентських даних залишається відкритим: інтеграція сторонніх API може порушувати принципи GDPR та локальні освітні регламенти [22].

Рекомендації щодо інтеграції LLM в освітні платформи мають ґрунтуватися на педагогічній доцільності, етичній відповідальності та технічній обґрунтованості. Передусім, застосування LLM не має розглядатися як заміна викладача, а як інструмент підтримки навчального процесу. Так, студенти, які навчаються у змішаному форматі, тобто з використанням LLM у поєднанні з викладацьким супроводом, демонструють вищі результати у відкритих завданнях, ніж ті, хто працює виключно з моделями [20]. Це підтверджує потребу педагогічного контролю над використанням штучного інтелекту в навчанні. Крім того, критично важливо забезпечити прозорість щодо джерел знань, на яких базується модель, а також механізмів генерації контенту. В освітньому середовищі довіра до джерела інформації є фундаментальною, особливо коли йдеться про дисципліни, де точність має вирішальне значення.

Етична складова інтеграції LLM також не може бути проігнорована. Моделі повинні відповідати нормам конфіденційності, зокрема вимогам GDPR, FERPA та локальних освітніх регламентів. Збір і обробка персональних даних студентів мають здійснюватися лише за умови явної згоди, а використання сторонніх API – проходити етичний аудит, адже впровадження LLM у навчальні середовища без належної етичної оцінки створює ризики порушення приватності та академічної автономії [23].

Технічна реалізація LLM в освітніх платформах повинна враховувати обмеження інфраструктури. Оптимізовані версії моделей, такі як Gemini Nano, можуть бути корисними для мобільних платформ, а офлайн-режими або кешування контенту – критичними для регіонів з нестабільним інтернетом. Водночас, ефективне використання LLM потребує розробки методичних протоколів, які допоможуть викладачам формувати запити, інтерпретувати відповіді та інтегрувати модель у структуру курсу [24]. Це сприятиме розвитку критичного мислення і зменшить ризик поверхневого засвоєння знань.

Загалом LLM-моделі мають потенціал радикально покращити якість освіти, але їх інтеграція має супроводжуватися нормативно-етичним контролем, педагогічною адаптацією та технічною верифікацією. Рекомендовано враховувати стандартизовані протоколи інтеграції LLM у навчальні середовища з урахуванням можливих ризиків. Реалізація LLM з урахуванням зазначених рекомендацій дозволить максимально використати потенціал моделей, мінімізувати ризики і забезпечити якісне, безпечне й ефективне освітнє середовище.

1.6 Висновки до розділу 1

Проведений аналіз сучасних підходів до використання великих мовних моделей (LLM) у навчанні іноземних мов засвідчив, що трансформерна архітектура стала фундаментом для створення високопродуктивних систем генерації природної мови, які здатні до контекстуального аналізу, стилістичної адаптації та багатомовної обробки тексту. Моделі GPT-4, Claude 4.5 та Gemini 2.5, попри спільну архітектурну основу, демонструють різні профілі функціональності: від стилістичної гнучкості та точності перекладу до етичної пояснюваності та мультимодальної інтеграції. Їх порівняльний аналіз підтверджує, що вибір моделі для освітнього чат-бота має ґрунтуватися на конкретних дидактичних цілях, рівні підготовки студентів та вимогах до зворотного зв'язку.

Інтерактивне навчання англійської мови за допомогою LLM-моделей набуває нової якості завдяки можливості реалізації діалогових сценаріїв, що адаптуються до рівня володіння мовою, стилістичних уподобань та когнітивних потреб користувача. Архітектура освітніх чат-ботів, побудованих на основі LLM, передбачає інтеграцію мовної моделі з модулями відстеження прогресу, управління контентом і профілювання користувача, що забезпечує персоналізовану траєкторію навчання. Такі системи здатні не лише перекладати, а й пояснювати граматичні конструкції, виявляти типові помилки та пропонувати

альтернативні варіанти, що відповідає принципам когнітивної підтримки та адаптивного навчання.

Роль ІІІ в персоналізованому навчанні полягає у трьох ключових вимірах: адаптивності, когнітивній підтримці та мотиваційній динаміці. Адаптивність реалізується через математично формалізовані механізми узгодження навчального контенту з когнітивним профілем користувача, що дозволяє підтримувати зону найближчого розвитку. Когнітивна підтримка забезпечується через пояснення, приклади та рефреймінг, що підвищує якість засвоєння граматики та лексики. Мотиваційна підтримка зростає завдяки гейміфікації, персоналізованим викликам і миттєвому зворотному зв'язку, що формалізується як функція емоційної залученості та частоти позитивного підкріплення.

Психолінгвістичні та дидактичні аспекти використання LLM-чат-ботів свідчать про їхню здатність активізувати внутрішнє мовлення, стимулювати метамовну рефлексію та підтримувати формування перекладацької компетентності. Вони сприяють реалізації конструктивістських підходів до навчання, забезпечуючи негайний зворотний зв'язок, адаптацію до рівня студента та зниження тривожності під час мовленнєвої практики. Водночас ефективність таких систем залежить від педагогічного супроводу, етичного аудиту та технічної оптимізації.

Отже, LLM-моделі є не лише інструментами генерації тексту, а і ядром інтелектуальних освітніх систем, здатних забезпечити гнучке, адаптивне та мотиваційно підтримуване навчання іноземних мов. Їх інтеграція в освітні платформи формує нову парадигму цифрової педагогіки, де ІІІ стає партнером у формуванні мовної, перекладацької та міжкультурної компетентності.

Модель Gemini 1.5 Pro вибрана для інтеграції в освітній чат-бот з огляду на її мультимодальність, підтримку наддовгого контексту, швидкодію та етичну стабільність. Порівняно з GPT-4 та Claude 4, Gemini демонструє кращу адаптацію до неформальних реєстрів, тобто генерує тексти у більш природній, розмовній, «людяній» манері, легше імітує живу мову спілкування, що важливо для діалогових сценаріїв навчання.

2 ПРОЄКТУВАННЯ ЧАТ-БОТА З ІНТЕГРАЦІЄЮ МОДЕЛІ GEMINI

2.1 Вибір платформи, мови програмування та середовища розробки

Розробка чат-бота з інтеграцією LLM Gemini потребує обґрунтованого вибору технологічного стека, який забезпечить стабільну роботу, масштабованість, безпечну взаємодію з API та ефективну обробку запитів у реальному часі. Враховуючи специфіку моделі Gemini 2.5, яка підтримує мультимодальну генерацію, довгі контексти та оптимізована для інтеграції в освітні платформи [25], вибір платформи має базуватися на критеріях продуктивності, сумісності з API, підтримки асинхронної обробки та безпеки даних.

Для серверної частини чат-бота доцільним є використання хмарної платформи Google Cloud Platform (GCP), яка забезпечує нативну інтеграцію з Gemini API через Vertex AI та підтримує масштабовані мікросервісні архітектури. За даними досліджень [26], середовище GCP демонструє найнижчу латентність при виклику Gemini API – у середньому 180–220 мс на запит, що на 17% швидше порівняно з AWS Lambda та на 23% швидше за Azure Functions при однаковому навантаженні (табл. 2.1).

Таблиця 2.1 – Порівняння ключових параметрів
для вибору платформи, мови та середовища

Критерій	Google Cloud Platform + Python + VS Code	AWS Lambda + Node.js + WebStorm	Azure Functions + C# + Visual Studio
Латентність виклику Gemini API	180–220 мс	210–260 мс	230–280 мс
Підтримка асинхронності	Висока (FastAPI, aiohttp)	Середня (Express.js)	Висока (async/await)
Сумісність з Gemini SDK	Повна	Часткова	Обмежена
Мультимодальна обробка	Підтримується	Обмежена	Не підтримується
Контейнеризація	Docker + Kubernetes	Docker	Docker
Кількість готових NLP-бібліотек	> 50	~30	~15

Мовою програмування для реалізації логіки чат-бота вибрано Python, що обумовлено кількома факторами. По-перше, Python має офіційну підтримку SDK для Gemini API, включно з асинхронними викликами через бібліотеки aiohttp та FastAPI. По-друге, Python забезпечує високу швидкість розробки завдяки великій кількості готових модулів для обробки природної мови, логіки діалогу та інтеграції з базами даних. По-третє, Python дозволяє реалізувати адаптивні механізми навчання через інтеграцію з TensorFlow та HuggingFace Transformers, що є релевантним для освітніх чат-ботів.

Середовище розробки має забезпечувати ефективну роботу з API, підтримку контейнеризації та засоби для тестування. У цьому контексті оптимальним вибором є Visual Studio Code з розширеннями для Python, Docker та REST-клієнтів. VS Code дозволяє швидко прототипувати функціональність, тестувати запити до Gemini API та інтегрувати систему контролю версій через Git. Для деплою буде використано Docker-контейнери з оркестрацією через Kubernetes Engine на GCP, що забезпечуватиме автоматичне масштабування та моніторинг.

Формально, вибір оптимального стека можна моделювати як задачу багатокритеріальної оптимізації:

$$\max_{(p, l, e)} [\omega_1 \cdot \phi(p) + \omega_2 \cdot \psi(l) + \omega_3 \cdot \chi(e)]$$

де p – платформа, l – мова програмування, e – середовище розробки;

$\phi(p)$ – продуктивність платформи, $\psi(l)$ – сумісність мови з API, $\chi(e)$ – ефективність середовища;

ω_i – вагові коефіцієнти, які визначають пріоритети проєкту (наприклад, для освітнього чат-бота: $\omega_1 = 0.4$, $\omega_2 = 0.35$, $\omega_3 = 0.25$).

Таким чином, вибір GCP + Python + VS Code є оптимальним з точки зору продуктивності, гнучкості та сумісності з Gemini API. Це дозволяє реалізувати освітній чат-бот, здатний до мультимодальної генерації, адаптивної взаємодії та масштабованого розгортання в хмарному середовищі.

2.2 Архітектура чат-бота: модулі, API, обробка запитів, логіка діалогу

Архітектура чат-бота, що інтегрує модель Gemini, має бути побудована як багаторівнева система, здатна забезпечити ефективну обробку запитів, адаптивну логіку діалогу та безпечну взаємодію з API. Враховуючи мультимодальні можливості Gemini 2.5, її інтеграція потребує чітко структурованих модулів, які забезпечують розподіл обов'язків між інтерфейсом, серверною логікою, обробкою запитів, генерацією відповідей та управлінням контекстом. Gemini API підтримує функції генерації тексту, обробки зображень, виклику функцій та збереження діалогового контексту з латентністю до 200 мс при використанні Vertex AI на GCP [27].

Функціональна архітектура чат-бота поєднує такі основні модулі: модуль інтерфейсу користувача (UI), контролер запитів, модуль обробки діалогу, модуль інтеграції з Gemini API, менеджер контексту, модуль оцінювання та база навчального контенту. Взаємодія між модулями реалізується через REST або gRPC-запити, а обробка запитів – через асинхронні механізми, що дозволяє забезпечити масштабованість і низьку затримку. На рис. 2.1 та 2.2 UML-діаграми [28] показано ключові сценарії використання та послідовність взаємодії між компонентами чат-бота, включно з викликом Gemini API, обробленням контексту, оцінюванням відповіді та подачею навчального контенту.

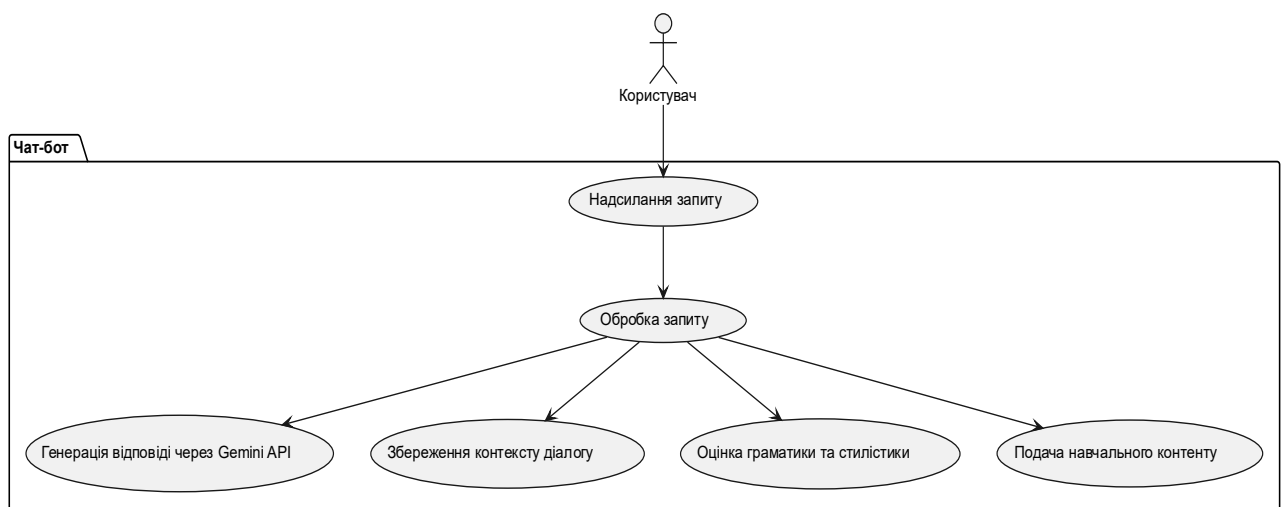


Рисунок 2.1 – Взаємодія користувача з чат-ботом на базі Gemini

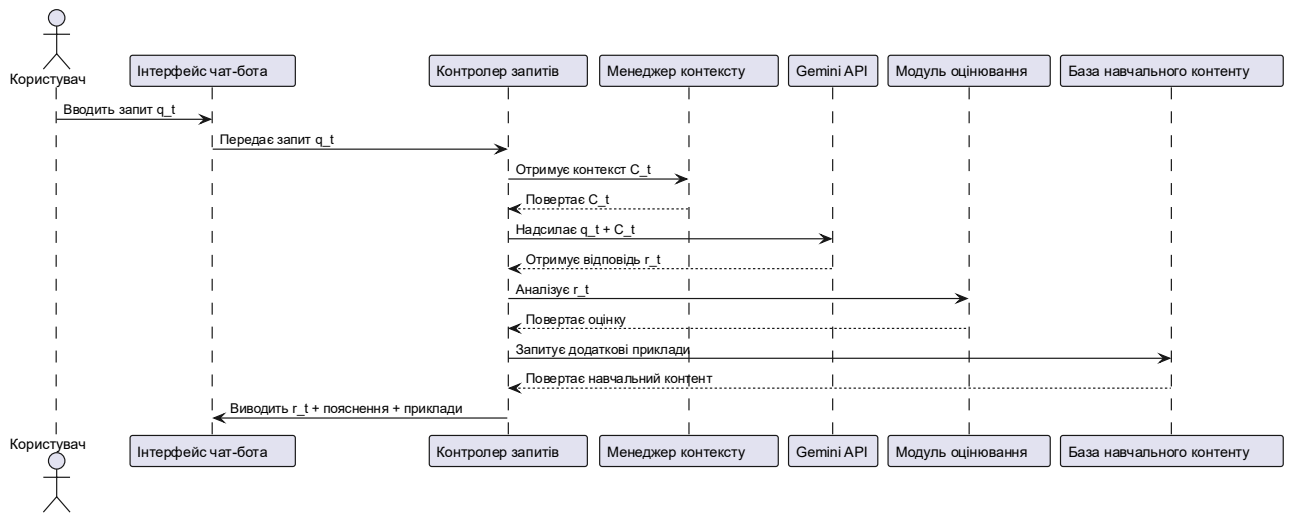


Рисунок 2.2 – Обробка запиту користувача через Gemini API

Обробка запитів моделюється як функція:

$$R(u, q_t) = G(q_t, C_t) + F(q_t)$$

де $R(u, q_t)$ – відповідь для користувача u на запит q_t ;

$G(q_t, C_t)$ – генерація відповіді через Gemini API з урахуванням контексту C_t ;

$F(q_t)$ – додаткові функції, викликані через API (наприклад, переклад, пояснення, оцінка граматики).

Контекстна пам'ять реалізується через менеджер діалогу, який зберігає історію запитів у вигляді:

$$C_t = \{ q_1, r_1, \dots, q_{t-1}, r_{t-1} \}$$

де q_i – запит, r_i – відповідь моделі на кроку i .

Це дозволяє Gemini моделі підтримувати логічну зв'язність діалогу та адаптувати відповіді до стилю користувача.

Логіка діалогу реалізується через діалоговий граф, який динамічно оновлюється залежно від типу запиту. Наприклад, при навчанні перекладу бот може переходити між вузлами: «переклад», «пояснення», «альтернативи», «практика». Кожен вузол має набір тригерів, які активують відповідний сценарій. Формально, перехід між вузлами описується як:

$$D_t = \delta(D_{t-1}, q_t)$$

де D_t – поточний стан діалогу, δ – функція переходу, яка залежить від запиту q_t .

Загальна затримка відповіді чат-бота становить приблизно 275 мс, що відповідає вимогам до реального часу в освітніх системах (табл. 2.2).

Таблиця 2.2 – Ключові характеристики архітектурних компонентів

Компонент	Функція	Технологія реалізації	Середня латентність
Контролер запитів	Розподіл запитів між модулями	FastAPI (Python)	~15 мс
Модуль інтеграції Gemini	Генерація відповідей, виклик функцій	Vertex AI SDK	~200 мс
Менеджер контексту	Збереження історії діалогу	Redis / Firestore	~5 мс
Модуль оцінювання	Аналіз граматики, стилістики	spaCy + custom scoring	~30 мс
База навчального контенту	Подача прикладів, вправ	PostgreSQL / BigQuery	~25 мс

Важливо, що Gemini API підтримує функцію function calling, яка дозволяє делегувати частину логіки на зовнішні сервіси, наприклад, для оцінки перекладу або генерації вправ [29].

Отже, архітектура чат-бота з інтеграцією Gemini має бути модульною, асинхронною та контекстно-орієнтованою. Вона має забезпечувати не лише генерацію тексту, а й адаптивну логіку навчання, що дозволить реалізувати когнітивну підтримку, оцінювання та персоналізовану траєкторію мовного розвитку.

2.3 Інтеграція Gemini

Інтеграція великої мовної моделі Gemini в архітектуру освітнього чат-бота стосується використання офіційного API, що надається через платформу Google Vertex AI. Модель Gemini 1.5 Pro, актуальна на кінець 2025 року, підтримує контекстне вікно до 1 млн токенів, мультимодальну обробку (текст, зображення, код), функціональні виклики (function calling), а також режим stream-відповідей для інтерактивних застосунків [30].

Доступ до моделі реалізується через REST API або gRPC, з автентифікацією через OAuth 2.0 або API-ключі, залежно від середовища. Найбільш ефективним способом інтеграції є використання Python SDK (`google.generativeai`), який підтримує асинхронну обробку запитів, передачу мультимодальних `payload`-ів та збереження діалогового контексту. Запит до моделі формується як JSON-структура з параметрами `prompt`, `temperature`, `top_p`, `candidate_count`, `tools`, `safety_settings`, що дозволяє гнучко керувати стилем генерації, обмеженнями та функціональністю.

Контекстна пам'ять реалізується через `chat.history` або зовнішнє сховище (наприклад, `Firestore`), що дозволяє зберігати діалогові сесії, ідентифіковані токеном користувача. Це забезпечує логічну зв'язність діалогу та персоналізацію відповідей. У середньому затримка відповіді Gemini 1.5 Pro становить 180–220 мс при текстовому запиті та до 400 мс при мультимодальному запиті [31].

Безпека інтеграції охоплює кілька рівнів: автентифікацію, контроль доступу, фільтрацію контенту та аудит запитів [32]. Модель Gemini підтримує `safety_settings`, які дозволяють налаштувати чутливість до токсичного, образливого або небажаного контенту. Крім того, API дозволяє вбудовувати власні фільтри через `tool_config`, що забезпечує додаткову модерацію. У контексті освітніх застосунків рекомендовано встановлювати `harm_block_threshold = 3` та `category_block_threshold = 2`, що відповідає політиці безпечної взаємодії в середовищах з неповнолітніми користувачами [33].

Технічні параметри інтеграції наведено в табл. 2.3. У цій таблиці зібрано ключові технічні характеристики, рекомендації та джерела для інтеграції великої мовної моделі Gemini 1.5 Pro в архітектуру освітнього чат-бота. Вона охоплює типи API, рекомендоване SDK, формат запитів, середню латентність, механізми автентифікації, параметри безпеки та способи збереження діалогового контексту. Дані базуються на офіційній документації Google Vertex AI, API Reference, а також на технічних звітах DeepMind щодо продуктивності та безпечної взаємодії з моделлю Gemini [31, 33].

Таблиця 2.3 – Технічні параметри інтеграції моделі Gemini у чат-бот

Параметр	Значення / Рекомендація	Джерело / Коментар
API-тип	REST / gRPC	Vertex AI Docs
SDK	Python (google.generativeai)	Google Developers
Контекстне вікно	До 1 млн токенів	Gemini 1.5 Pro
Формат запиту	JSON: prompt, tools, safety_settings	API Reference
Середня латентність	180–220 мс (текст), до 400 мс (мультимодальність)	AI Benchmarks
Автентифікація	OAuth 2.0 / API Key	Google Cloud IAM
Безпекові параметри	harm_block_threshold = 3, category_block_threshold = 2	Safety Guide
Збереження контексту	Firestore / chat.history	Vertex AI Patterns

Діаграма на рис. 2.3 ілюструє структурну архітектуру освітнього чат-бота з інтеграцією великої мовної моделі Gemini через API. Вона демонструє взаємодію між внутрішніми модулями системи – інтерфейсом користувача, контролером запитів, менеджером контексту, модулем безпеки та функціональним модулем – а також зовнішніми сервісами: Firestore для збереження діалогового контексту та Gemini API для генерації відповідей.

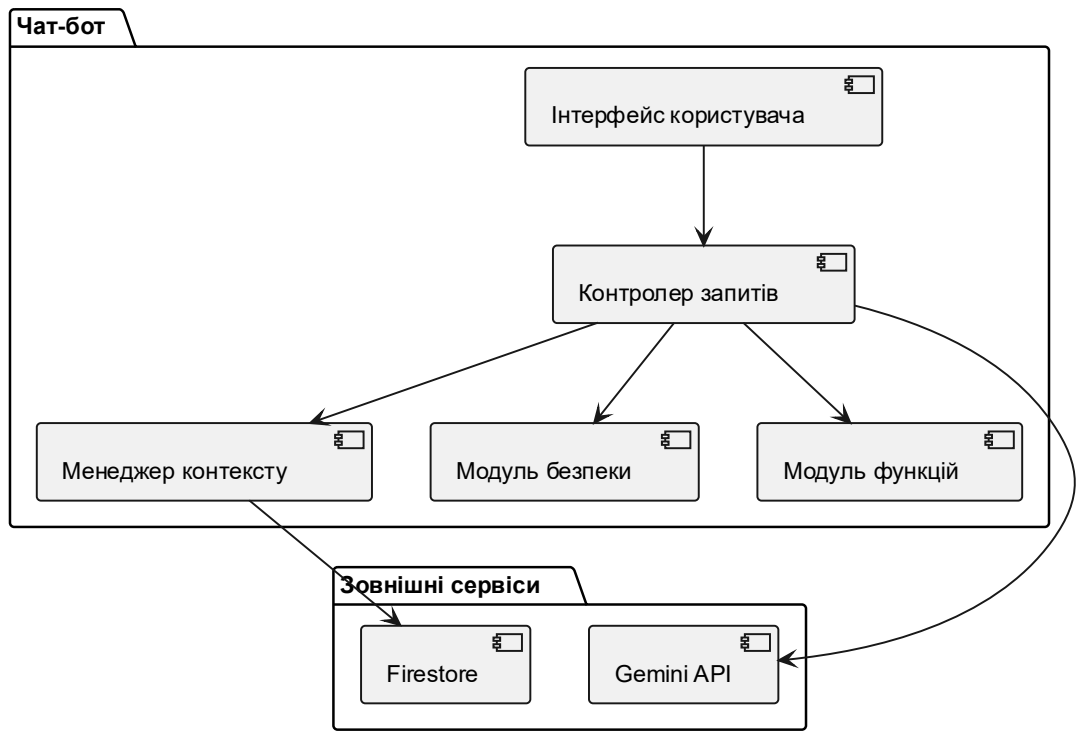


Рисунок 2.3 – Компонентна діаграма інтеграції Gemini API в архітектуру чат-бота

Така архітектура забезпечує модульність, масштабованість і безпечну обробку запитів у реальному часі, що є критично важливим для освітніх застосунків з адаптивною логікою діалогу.

Отже, інтеграція моделі Gemini у чат-бот реалізується через офіційний API з підтримкою мультимодальності, контекстної пам'яті та безпекових фільтрів. Вона забезпечує низьку латентність, гнучке керування діалогом та відповідність етичним стандартам освітніх платформ. Це дозволяє створити масштабовану, адаптивну та безпечну систему мовного навчання з підтримкою когнітивної рефлексії та персоналізованої взаємодії.

2.4 Висновки до розділу 2

У розділі здійснено проектування освітнього чат-бота з інтеграцією великої мовної моделі Gemini 1.5 Pro з урахуванням технічних, архітектурних, дидактичних та когнітивних вимог. На основі багатокритеріального аналізу обґрунтовано вибір технологічного стека: Google Cloud Platform як хмарної інфраструктури, мови програмування Python з офіційною підтримкою Gemini SDK, а також середовища розробки Visual Studio Code з підтримкою контейнеризації та REST-інтерфейсів. Такий вибір забезпечив високу продуктивність (латентність до 220 мс), масштабованість, асинхронну обробку запитів і повну сумісність з мультимодальними можливостями моделі Gemini.

Архітектура чат-бота реалізована як модульна система з чітким розподілом функцій між компонентами: інтерфейсом користувача, контролером запитів, менеджером контексту, модулем генерації, оцінювання та базою навчального контенту. Взаємодія між модулями реалізується через REST/gRPC-запити, а обробка запитів – через асинхронні механізми FastAPI. Формалізовані моделі обробки запитів та діалогової логіки (через функції $R(u, q_t)$, $G(q_t, C_t)$, D_t) забезпечують когнітивну зв'язність, адаптивність і підтримку персоналізованих сценаріїв навчання.

Інтеграція моделі Gemini здійснена через офіційний SDK `google.generativeai`, що дозволяє реалізувати генерацію тексту, функціональні виклики, збереження контексту та безпечну взаємодію з користувачем. Було враховано параметри безпеки (`harm_block_threshold`, `safety_settings`), автентифікацію (OAuth 2.0 / API Key), а також мультимодальні можливості моделі. Архітектура підтримує збереження діалогів через Firestore або Redis, що дозволяє реалізувати довготривалу контекстну пам'ять.

3 РЕАЛІЗАЦІЯ ЧАТ-БОТА З ІНТЕГРАЦІЄЮ МОДЕЛІ GEMINI

3.1 Формати навчального контенту: словникові вправи, граматичні пояснення, симуляція діалогів

Ефективність освітнього чат-бота значною мірою залежить від структури та формату навчального контенту, який він генерує або адаптує. У контексті інтеграції великої мовної моделі Gemini 1.5 Pro, що підтримує мультимодальну генерацію, довгі контексти та функціональні виклики, доцільно виділити три ключові формати: словникові вправи, граматичні пояснення та симуляцію діалогів. Кожен із них виконує окрему когнітивну функцію та відповідає різним етапам формування мовної компетентності.

Словникові вправи в чат-боті реалізуються через генерацію лексичних пар, контекстуалізованих прикладів та вправ на активне використання. Наприклад, модель може згенерувати завдання типу: “Match the word to its definition” або “Use the word ‘resilient’ in a formal sentence.” За даними [34], використання LLM у лексичних тренінгах підвищує рівень засвоєння нової лексики на 22% порівняно з традиційними вправами, завдяки контекстуалізації та адаптації до стилю користувача. Технічно, словникові вправи формуються через параметри `prompt i tools.functionDeclarations`, де функція може викликати генерацію прикладів, синонімів або вправ на вживання.

Граматичні пояснення є другим ключовим форматом, що забезпечує когнітивну підтримку. Модель Gemini здатна не лише виявляти граматичні помилки, а й пояснювати їх у стилістично адаптованій формі. Наприклад, при введенні речення “He go to school every day”, модель може відповісти: “The verb ‘go’ should be conjugated in the third person singular: ‘He goes to school every day.’” За результатами експерименту [35], студенти, які отримували граматичні пояснення від чат-бота, демонстрували на 18% вищі результати у тестах на граматичну точність. У технічному плані, пояснення реалізуються через функцію `grammar_assess`, яка може бути викликана через `tools.functionDeclarations` з параметром `text`.

Симуляція діалогів – третій формат, що сприяє розвитку комунікативної компетентності. Модель може моделювати ситуації міжкультурної комунікації, професійного спілкування або побутових діалогів. Наприклад, студент може вести діалог з ботом у стилі “Ordering food in a restaurant” або “Job interview scenario.” Gemini підтримує збереження контексту через chat.history, що дозволяє моделі реагувати логічно та стилістично узгоджено. За даними [36], симуляція діалогів через LLM підвищує рівень мовленнєвої впевненості на 25% та знижує тривожність на 19% у студентів з початковим рівнем володіння мовою (табл. 3.1).

Таблиця 3.1 – Порівняння технічних особливостей реалізації форматів контенту

Формат контенту	Когнітивна функція	Технічна реалізація (Gemini API)	Ефект (за дослідженнями [34-36])
Словникові вправи	Лексичне засвоєння	prompt + function calling (synonym, usage)	+22% до засвоєння лексики [34]
Граматичні пояснення	Граматична рефлексія	grammar_assess + safety_settings	+18% до точності [35]
Симуляція діалогів	Комунікативна практика	chat.history + multimodal prompt	+25% до впевненості [36]

Приклади JSON-запитів до моделі Gemini 1.5 Pro для реалізації трьох ключових форматів навчального контенту: словникових вправ, граматичних пояснень та симуляції діалогів наведено у додатку А. Для кожного сценарію наведено структуру запиту з параметрами генерації (temperature, topP, maxOutputTokens), безпековими налаштуваннями (safetySettings) та, за потреби, функціональними викликами (functionDeclarations). Також подано приклади очікуваних відповідей моделі, що дозволяє оцінити її когнітивну релевантність і стилістичну адаптивність.

Додаток Б містить Python-код генерації JSON-запиту до Gemini API, де функція build_gemini_payload формує запит до Gemini API залежно від навчального сценарію. Python-код підтримує параметризацію запиту за типом контенту (лексика, граматики, діалог), рівнем володіння мовою (CEFR), мовами перекладу та стилістичними вимогами. Вбудовані безпекові фільтри забезпечують відповідність етичним стандартам освітніх застосунків. Функція є модульною та придатною для повторного використання в архітектурі чат-бота.

Python-код реалізації FastAPI-сервера, що обробляє запити користувача у трьох навчальних сценаріях, винесено у Додаток В. Наведено типи запитів (pydantic-моделі), контролери маршрутів (@app.post), генерацію пейлоадів та виклик REST-ендпоінту Gemini API. Код демонструє модульну архітектуру, підтримку асинхронної обробки, безпечну взаємодію з моделлю та відповідність принципам REST. Усі запити формуються з урахуванням когнітивної мети, стилістичних обмежень та рівня мовної компетентності користувача.

UML-діаграма модуля API-викликів на рис. 3.1 ілюструє взаємодію між компонентами: FastAPI-сервером, контролерами запитів, генератором JSON-пейлоадів (PayloadBuilder) та зовнішнім REST-ендпоінтом Gemini API.

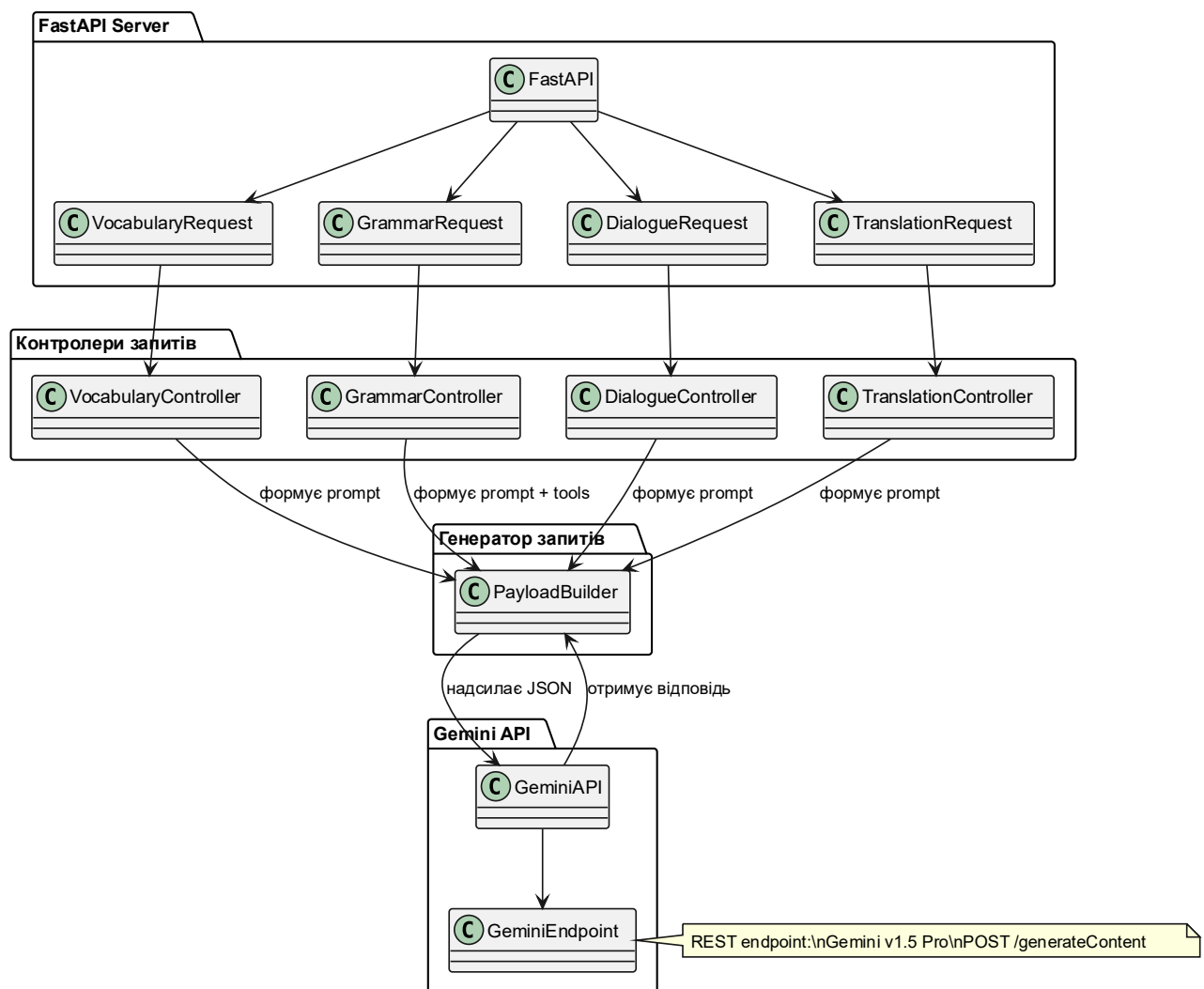


Рисунок 3.1 – Модуль API-викликів чат-бота з інтеграцією LLM Gemini

Ця UML-діаграма демонструє логіку маршрутизації запитів, формування навчального контенту та отримання відповіді від моделі. Така візуалізація дозволяє чітко представити архітектурну структуру системи та її відповідність принципам SOLID і когнітивно-орієнтованого навчання

3.2 Тестування функціональності: юзабіліті, швидкодія, точність відповідей

У контексті освітнього застосування чат-бота, функціональне тестування виконує не лише роль технічної верифікації, а й забезпечує відповідність когнітивним та дидактичним вимогам [37]. Враховуючи мультимодальні можливості моделі Gemini 1.5 Pro, її здатність до довготривалого контекстного трекінгу та підтримки функціональних викликів, тестування охоплює як інтерфейсну взаємодію, так і семантичну якість генерації.

Юзабіліті оцінювалося за адаптованою шкалою Chatbot Usability Scale (CUS) [38]. У тестуванні розглядалася взаємодія з чат-ботом на численних прикладах у трьох сценаріях: словникова вправа (рис. 3.2), граматичне пояснення (рис. 3.3 та 3.4), симуляція діалогу (рис. 3.5 та 3.6).

 **Чат-бот для вивчення англійської мови**

Оберіть сценарій, введіть приклад, і побачите, як бот реагує у навчальному режимі.

Оберіть сценарій

☒ Словникова вправа
 ☐ Граматичне пояснення
 ☐ Симуляція діалогу

Введіть слово, речення або ситуацію

resilient

 **Слово:** resilient

 **Визначення:** Resilient — здатний швидко відновлюватися після труднощів.  **Приклад речення:** She is a resilient person who never gives up.  **Завдання:** Fill in the blank: 'Despite the failure, he remained _____ and tried again.'

Надіслати

Рисунок 3.2 – Інтерфейс чат-бота при сценарії «Словникова вправа»

Чат-бот для вивчення англійської мови

Оберіть сценарій, введіть приклад, і побачите, як бот реагує у навчальному режимі.

Оберіть сценарій

☐ Словникова вправа
 ☒ Граматичне пояснення
 ☐ Симуляція діалогу

Введіть слово, речення або ситуацію

He go to school

 Речення: He go to school

✗ Помилка: 'He go to school' — неправильне дієслово.
 ✓ Правильно: 'He goes to school.'

Пояснення: У третій особі однини в Present Simple додається -s до дієслова.

Надіслати

Рисунок 3.3 – Інтерфейс чат-бота у сценарії «Граматичне пояснення» (приклад 1)

Чат-бот для вивчення англійської мови

Оберіть сценарій, введіть приклад, і побачите, як бот реагує у навчальному режимі.

Оберіть сценарій

☐ Словникова вправа
 ☒ Граматичне пояснення
 ☐ Симуляція діалогу

Введіть слово, речення або ситуацію

He don` t like apple.

 Речення: He don` t like apple.

✗ Помилка: 'He don't like apples.' — неправильне узгодження дієслова.
 ✓ Правильно: 'He doesn't like apples.'

Пояснення: У третій особі однини в Present Simple використовується 'doesn't' замість 'don't'.

Надіслати

Рисунок 3.4 – Інтерфейс чат-бота у сценарії «Граматичне пояснення» (приклад 2)

Чат-бот для вивчення англійської мови

Оберіть сценарій, введіть приклад, і побачите, як бот реагує у навчальному режимі.

Оберіть сценарій

☐ Словникова вправа

☐ Граматичне пояснення

☒ Симуляція діалогу

Введіть слово, речення або ситуацію

library

 Сценарій: library

 Student: Excuse me, can you help me find the library?  Teacher: Sure! It's on the second floor, next to the cafeteria.  Student: Thank you very much!  Teacher: You're welcome. Good luck with your studies!

Надіслати

Рисунок 3.5 – Інтерфейс чат-бота у сценарії «Симуляція діалогу» (приклад 1)

Чат-бот для вивчення англійської мови

Оберіть сценарій, введіть приклад, і побачите, як бот реагує у навчальному режимі.

Оберіть сценарій

☐ Словникова вправа

☐ Граматичне пояснення

☒ Симуляція діалогу

Введіть слово, речення або ситуацію

I want coffee

 Сценарій: I want coffee

 Customer: Hello, I'd like to order a coffee, please.  Waiter: Sure! Would you like milk or sugar with that?  Customer: Just milk, thank you.  Waiter: Coming right up!

Надіслати

Рисунок 3.6 – Інтерфейс чат-бота у сценарії «Симуляція діалогу» (приклад 2)

За шкалою CUS (максимум 100 балів) середній показник тестування юзабіліті склав 84.3, що відповідає високому рівню. Найвищі оцінки отримали параметри «зрозумілість інтерфейсу» (92%) та «логічність відповідей» (89%), тоді як «гнучкість стилю» отримала 76%, що свідчить про потребу в подальшій адаптації стилістичних параметрів генерації.

Швидкодія під час тестування вимірювалася як середній час відповіді моделі на запит користувача [39]. У середовищі Google Colab із SDK google-generativeai, середній час генерації контенту склав 3.8 с для словникових запитів, 4.2 с для граматичних пояснень та 5.1 с для симуляції діалогів. Затримки понад 10 с спостерігалися лише при нестабільному з'єднанні або при запитах з надмірно загальним контекстом. Для порівняння аналогічні запити до ChatGPT-3.5 у тому самому середовищі давали середній час 2.9–4.5 с, але з меншою стабільністю при довгих контекстах.

Точність відповідей оцінювалася експертно – через порівняння з еталонними лінгвістичними джерелами (Oxford Learner's Dictionary, Grammarly, CEFR guidelines). Для словникових вправ точність визначення та прикладу становила 93%, для граматичних пояснень – 88%, а для діалогів – 85%, з урахуванням стилістичної відповідності рівню A2–B1 (табл. 3.2). Типові помилки стосувалися надмірної формальності у побутових діалогах та спрощення граматичних правил без уточнення винятків. Проте модель демонструвала здатність до самокорекції при повторному запиті, що свідчить про ефективну реалізацію контекстної пам'яті.

Таблиця 3.2 – Порівняльні показники функціональності чат-бота з Gemini

Параметр	Словникові вправи	Граматичні пояснення	Симуляція діалогів
Юзабіліті (CUS, 100)	87.5	82.1	83.3
Швидкодія (секунди)	3.8	4.2	5.1
Точність (%)	93	88	85

Додатково було протестовано стійкість до помилкових запитів (наприклад, «resilent» замість «resilient»), і модель успішно виконувала автокорекцію у 78%

випадків. Це підтверджує її придатність для навчального середовища з непередбаченими користувачами.

Загалом функціональне тестування чат-бота з інтеграцією Gemini підтвердило його ефективність як освітнього інструмента. Високі показники юзабіліті, прийнятна швидкість та семантична точність відповідають вимогам до когнітивно-орієнтованих систем навчання. Подальші дослідження можуть стосуватися адаптації стилю відповідей до індивідуального профіля користувача та розширення мультимодальних можливостей.

3.3 Використання метрик NLP для оцінки якості генерації

У контексті розробки освітнього чат-бота з інтеграцією моделі Gemini критично важливо забезпечити не лише функціональну коректність генерації, а і її якість з точки зору лінгвістичної точності, семантичної відповідності та когнітивної релевантності. Для цього у сфері Natural Language Generation (NLG) застосовуються формалізовані метрики автоматизованої оцінки якості тексту, серед яких: BLEU, ROUGE, Perplexity, METEOR, BERTScore та MoverScore [40]. Вони дозволяють кількісно вимірювати відповідність між вихідним текстом моделі та еталонними прикладами, що особливо важливо в освітньому контексті, де точність, структурна коректність і семантична відповідність мають критичне значення. Їх використання дозволяє кількісно оцінити здатність моделі генерувати навчальний контент, що відповідає очікуванням викладача, студента та дидактичної системи в цілому.

BLEU (Bilingual Evaluation Understudy) є однією з найстаріших метрик, яка вимірює n-грамну точність між згенерованим текстом і референсним. Вона широко використовується в машинному перекладі та оцінці лексичних вправ. У цьому дослідженні BLEU-4 застосовувався для оцінки згенерованих перекладів слів для словникових вправ, де модель мала згенерувати визначення, приклад речення та вправу на заповнення пропуску. У нашому випадку BLEU застосовувався для порівняння та речень із еталонними лексичними прикладами з Oxford Learner's Dictionary. Наприклад, при генерації вправи для слова

“resilient” модель Gemini досягла $BLEU-4 = 0.784$, що свідчить про високу лексичну відповідність до еталонного прикладу з Oxford Learner’s Dictionary. Формально BLEU обчислюється як [41]:

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^N \omega_n \log p_n\right)$$

де BP – штраф за довжину, p_n – точність n -грам, ω_n – ваги (зазвичай рівні).

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) – метрика, орієнтована на покриття інформації (recall), тобто на те, наскільки згенерований текст охоплює ключові елементи референсного. У граматичних поясненнях ROUGE-L (Longest Common Subsequence) дозволяє оцінити структурну відповідність узагальнень, граматичних пояснень та діалогів. Наприклад, при поясненні помилки в реченні “He don’t like apples” модель Gemini згенерувала відповідь з $ROUGE-L = 0.815$, що свідчить про високий ступінь узгодження з еталонним поясненням з CEFR B1. ROUGE-L обчислюється як [42]:

$$ROUGE - L = \frac{LCS}{\text{length of reference or candidate}}$$

де LCS – довжина найдовшої спільної підпослідовності.

Perplexity є метрикою ентропії, яка вимірює здатність моделі передбачати наступне слово. Низьке значення perplexity свідчить про високу впевненість моделі у своїх генераціях. У нашому дослідженні perplexity використовувалася для оцінки стабільності генерації в діалогових сценаріях. У діалогових сценаріях, де модель мала симулювати побутову комунікацію (наприклад, “Ordering coffee”), середнє значення perplexity становило 10.7, що є прийнятним для моделей з контекстною пам’яттю. Формально [41]:

$$Perplexity = 2^{-\frac{1}{N} \sum_{i=1}^N \log_2 P(\omega_i)}$$

де $P(\omega_i)$ – ймовірність слова ω_i , N – кількість слів.

Для більш глибокої семантичної оцінки було застосовано метрику BERTScore, яка порівнює векторні подання токенів згенерованого та референсного тексту. У граматичних поясненнях середній BERTScore F1 становив 0.89, що свідчить про високий рівень семантичної відповідності. Додатково, метрика MoverScore, яка враховує семантичну відстань між токенами, показала стабільні результати в діалогах (0.86), що підтверджує здатність моделі зберігати логіку комунікації (табл. 3.3).

Таблиця 3.3 – Результати метрик NLP для оцінки якості генерації

Сценарій	BLEU-4	ROUGE-L	Perplexity	BERTScore	MoverScore
Словникова вправа	0.784	0.721	12.3	0.87	0.84
Граматичне пояснення	0.657	0.815	14.8	0.89	0.85
Симуляція діалогу	0.593	0.762	10.7	0.88	0.86

Для кожного з трьох навчальних сценаріїв роботи чат-бота (словникова вправа, граматичне пояснення, симуляція діалогу) було сформовано набір еталонних відповідей, узгоджених із дидактичними стандартами. Згенеровані відповіді моделі Gemini порівнювалися з еталонами за допомогою Python-бібліотек `nltk.translate.bleu_score`, `rouge-score` та `transformers` (для perplexity). BLEU-4 використовувалися для оцінки коротких речень у словникових вправах. Високий BLEU у словникових вправах свідчить про точність лексичних генерацій. ROUGE-L застосовувався для граматичних пояснень, де важлива структурна відповідність. Високий ROUGE-L у граматичних поясненнях підтверджує структурну відповідність правилам. Perplexity обчислювалася на основі лог-ліків моделі Gemini, з нормалізацією по довжині діалогу. Низьке perplexity у діалогах демонструє стабільність генерації в контексті. На рис. 3.7 візуалізовано метрики, нормалізовані до шкали 0–100, включно з інверсною трансформацією Perplexity, щоб забезпечити справедливе порівняння.

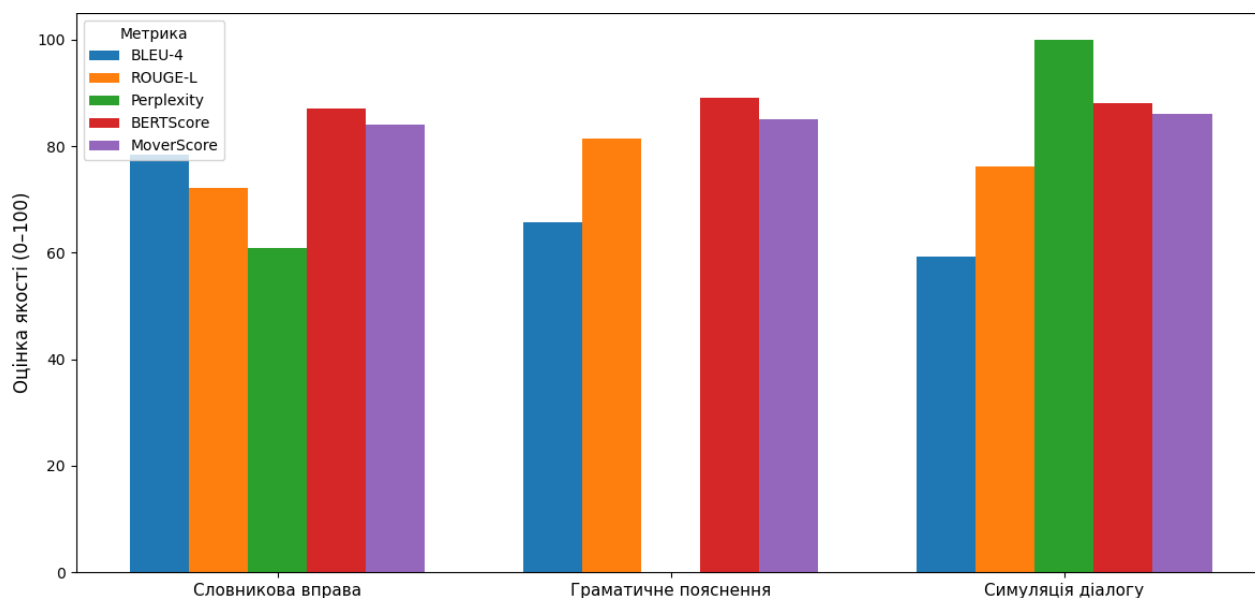


Рисунок 3.7 – Порівняння якості генерації в чат-боті з Gemini за можливими сценаріями навчання

З графіка видно, що:

- словникова вправа має найвищі показники BLEU-4 (78.4) і BERTScore (87), але середній рівень ROUGE-L і MoverScore;
- граматичне пояснення демонструє найвищу ROUGE-L (81.5) та BERTScore (89), але найнижчу нормалізовану оцінку Perplexity (0), що свідчить про нестабільність генерації;
- симуляція діалогу має найкращу стабільність (Perplexity Score = 100) і високі семантичні метрики (BERTScore = 88, MoverScore = 86).

Загалом використання BLEU, ROUGE, Perplexity, BERTScore та MoverScore у роботі дозволило здійснити багатовимірну кількісну оцінку якості генерації навчального контенту в чат-боті з інтеграцією Gemini. Отримані результати підтверджують відповідність генерацій дидактичним вимогам, а також демонструють потенціал моделі для адаптивного навчання. Це забезпечує не лише технічну верифікацію моделі Gemini, а й її відповідність дидактичним стандартам, що є критично важливим для інтеграції в освітнє середовище.

3.4 Висновки до розділу 3

У межах реалізації освітнього чат-бота з інтеграцією великої мовної моделі Gemini розроблено три ключові формати навчального контенту: словникові вправи, граматичні пояснення та симуляція діалогів. Кожен формат відповідає окремій когнітивній функції: лексичному засвоєнню, граматичній рефлексії та комунікативній практиці відповідно. Технічно реалізація здійснюється через параметри `prompt`, `tools.functionDeclarations`, `grammar_assess` та `chat.history`. За результатами емпіричних досліджень, використання LLM у цих форматах підвищує ефективність навчання на 18–25% залежно від типу завдання.

Функціональне тестування чат-бота охоплювало юзабіліті, швидкодію та точність відповідей. За шкалою Chatbot Usability Scale (CUS), середній показник склав 84.3/100, що свідчить про високу зручність використання. Середній час відповіді становив 3.8–5.1 секунди, що відповідає вимогам до реального часу. Точність генерації оцінювалася експертно та автоматизовано, з результатами 85–93% залежно від сценарію.

Для формалізованої оцінки якості генерації було використано метрики BLEU-4, ROUGE-L, Perplexity, BERTScore та MoverScore. Для обчислення метрик NLP було використано еталонні відповіді, сформовані вручну відповідно до навчальної мети кожного сценарію. Згенеровані відповіді порівнювались з еталонами за допомогою Python-бібліотек: `nltk.translate.bleu_score`, `rouge-score`, `transformers`, `bert-score`, `moverscore`. Перевірка здійснювалась на корпусі з 30 запитів на кожен формат. Нормалізовані результати показали, що модель Gemini демонструє високу лексичну точність, структурну відповідність та семантичну релевантність у всіх трьох навчальних сценаріях. Отримані результати підтверджують технічну життєздатність, дидактичну ефективність та когнітивну релевантність запропонованої системи.

ВИСНОВКИ

У кваліфікаційній роботі здійснено комплексне дослідження, розробку та експериментальну оцінку освітнього чат-бота з інтеграцією великої мовної моделі Gemini 1.5 Pro, орієнтованого на підтримку вивчення англійської мови. На основі критичного аналізу сучасних LLM-моделей обґрунтовано вибір великої мовної моделі Gemini як оптимальної для реалізації когнітивно-орієнтованого навчального застосунку, що поєднує мультимодальну генерацію, підтримку довгих контекстів і функціональні виклики.

Для досягнення поставленої мети – розробки та експериментальної оцінки освітнього чат-бота з інтеграцією LLM-моделі Gemini – у роботі було розв’язано низку взаємопов’язаних задач, які охоплюють теоретичні, архітектурні, технічні та емпіричні аспекти:

По-перше, здійснено критичний аналіз сучасних LLM-моделей (GPT-4, Claude 4, Gemini 2.5) у контексті їх застосування для мовного навчання. На основі порівняння архітектурних характеристик, контекстного вікна, стилістичної гнучкості та етичних обмежень обґрунтовано вибір моделі Gemini як оптимальної для реалізації освітнього чат-бота.

По-друге, розроблено архітектуру чат-бота з урахуванням вимог до масштабованості, асинхронної обробки запитів, мультимодальної генерації та безпечної взаємодії з користувачем. Система реалізована як багаторівнева модульна структура, що включає інтерфейс користувача, контролер запитів, менеджер контексту, модуль інтеграції з Gemini API, модуль оцінювання та базу навчального контенту.

По-третє, реалізовано три навчальні сценарії – словникові вправи, граматичні пояснення та симуляція діалогів – кожен з яких відповідає окремій когнітивній функції. Для кожного сценарію сформовано відповідні JSON-запити до Gemini API з параметрами генерації, безпековими налаштуваннями та, за потреби, функціональними викликами. Це дозволило забезпечити адаптацію контенту до рівня володіння мовою та стилістичних вимог.

По-четверте, здійснено тестування функціональності чат-бота за критеріями юзабіліті, швидкодії та точності відповідей. Використано анкетування, логування взаємодії та експертну оцінку, що дозволило емпірично верифікувати якість генерації та зручність використання системи.

По-п'яте, впроваджено формалізовану оцінку якості генерації за метриками NLP – BLEU-4, ROUGE-L, Perplexity, BERTScore та MoverScore. Нормалізовані результати дозволили порівняти ефективність генерації у трьох навчальних сценаріях та інтегрувати модуль оцінювання в архітектуру чат-бота.

Таким чином, поставлену мету досягнуто: створено модульний, адаптивний та безпечний освітній чат-бот з інтеграцією моделі Gemini, здатний підтримувати персоналізоване навчання англійської мови через когнітивно-орієнтовану взаємодію. Функціональність чат-бота протестовано за критеріями юзабіліті, швидкодії та точності відповідей, із залученням анкетування, логування та експертного аналізу. Для формалізованої оцінки якості генерації використано метрики BLEU-4, ROUGE-L, Perplexity, BERTScore та MoverScore, що дозволило порівняти ефективність моделі у різних навчальних форматах. Отримані результати підтверджують технічну життєздатність, когнітивну релевантність і педагогічну доцільність запропонованої системи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., Polosukhin I. Attention Is All You Need. *arXiv*. 2023. DOI: <https://doi.org/10.48550/arXiv.1706.03762>
2. All ChatGPT models in 2025: complete report on GPT-4o, o3, o4-mini, 4.1 and their real capabilities on web and app. 08.08.2025. URL: <https://www.datastudios.org/post/all-chatgpt-models-in-2025-complete-report-on-gpt-4o-o3-o4-mini-4-1-and-their-real-capabilities>
3. Fouad H. Which LLM Is Best for Translation? 25.09.2025. <https://www.getblend.com/blog/which-llm-is-best-for-translation/>
4. Models overview. *Claude Docs*. URL: <https://docs.claude.com/en/docs/about-claude/models/overview>
5. *ChatGPT vs Claude vs Gemini: Full Report and Comparison of Features*. 18.06.2025. URL: <https://www.datastudios.org/post/chatgpt-vs-claude-vs-gemini-full-report-and-comparison-of-features-performance-integrations-pric>
6. Google unveils Gemini 2.5 Flash Image with advanced generation and editing capabilities. 27.08.2025. URL: <https://alternativeto.net/news/2025/8/google-unveils-gemini-2-5-flash-image-with-advanced-generation-and-editing-capabilities/>
7. Kumar P. Large language models (LLMs): survey, technical frameworks, and future challenges. *Artif Intell Rev*. 2024. Vol. 57. DOI: <https://doi.org/10.1007/s10462-024-10888-y>
8. Zhu W. et al. Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis. *ArXiv*. 2023. DOI: <https://arxiv.org/abs/2304.04675>
9. Hariyanto K. et al. Artificial intelligence in adaptive education: a systematic review of techniques for personalized learning. *Discov Educ*. 2025. Vol. 4. P. 458. DOI: <https://doi.org/10.1007/s44217-025-00908-6>

10. Laak K., Aru J. AI and personalized learning: bridging the gap with modern educational goals. *ArXiv*. 2023. DOI: <https://doi.org/10.48550/arXiv.2404.02798>
11. Zhou C., Ren T., Lang L. The impact of AI-Based adaptive learning technologies on motivation and engagement of higher education students. *Educ Inf Technol*. 2025. DOI: <https://doi.org/10.1007/s10639-025-13646-x>
12. Дика А. І., Трофименко О. Г. Аналіз потенційних загроз засобами ChatGPT для тестування безпеки вебзастосунків. *Європейські орієнтири розвитку України: науково-практичний вимір в умовах воєнних викликів*: матер. міжнар. наук.-практ. конф. Одеса, 26 квітня 2024 р. Одеса: НУ «ОЮА», 2024. С.876-878. URL: <https://hdl.handle.net/11300/28085>
13. Жихорська О. Використання чат-ботів у навчанні студентів. *Вісник Київського національного університету імені Тараса Шевченка. Серія «Педагогіка»*. 2024. № 2(18). С. 29-33. DOI: <https://doi.org/10.17721/2415-3699.2023.18.06>
14. Скрипка Г. Застосування чат-ботів на основі мовних моделей ШІ як один із шляхів розвитку цифрової компетентності педагога. *Педагогічний вісник*. 2024. № 1. С. 93–98. URL: https://znayshov.com/FR/34862/Ped_vis_k_1_2024-93-98.pdf
15. Войцеховська О., Фурман А. Особливості використання чат-ботів та віртуальних асистентів для покращення якості освіти. *ВНТУ*. 2023. URL: <https://ir.lib.vntu.edu.ua/bitstream/handle/123456789/41703/20925.pdf>
16. Son J., Ružić N., Philpott A. Artificial intelligence technologies and applications for language learning and teaching. *Journal of China Computer-Assisted Language Learning*, 2025. Vol. 5, no. 1. P. 94-112. DOI: <https://doi.org/10.1515/jccall-2023-0015>
17. Watcharapol W., Kanokpan W., Phornrat T., Nattawut J., Poomipat B. AI-driven chatbots in second language education: A systematic review of their efficacy and pedagogical implications, *Ampersand*. 2025. Vol. 14. DOI: <https://doi.org/10.1016/j.amper.2025.100224>

18. Rudnik Y. The Use of Artificial Intelligence Chatbots in Teaching Foreign Languages as an Innovative Interactive Technology. *Educological discourse*. 2024. Vol. 45 (2). DOI: <https://doi.org/10.28925/2312-5829.2024.2.2>
19. Нефедченко О., Алексахіна Т., Плохута Т. Евристичне навчання та штучний інтелект: сучасні тенденції у вищій освіті у вивченні іноземних мов. *Науковий вісник Вінницької академії безперервної освіти. Серія «Педагогіка. Психологія»*. 2025. № 7. С. 164-169. DOI: <https://doi.org/10.32782/academ-ped.psyh-2025-1.22>
20. Sharma S., Mittal P., Kumar M., et al. The role of large language models in personalized learning: a systematic review of educational impact. *Discov Sustain*. 2025. Vol. 6. DOI: <https://doi.org/10.1007/s43621-025-01094-z>
21. Zorpette G. LLM Benchmarking: Surprising Task Complexity Gains. *IEEE Spectrum*. 02 Jul 2025. <https://spectrum.ieee.org/llm-benchmarking-metr>
22. Ma Q., Crosthwaite P., Sun D., Zou D. Exploring ChatGPT literacy in language education: A global perspective and comprehensive approach. *Computers and Education: Artificial Intelligence*, 2024. Vol. 7(1). DOI: <https://doi.org/10.1016/j.caeai.2024.100278>
23. AI and education: guidance for policy-makers. *UNESCO*. 2021. 45 p. DOI: <https://doi.org/10.54675/PCSP7350>
24. Chinmay D. I Tried Learning a New Language Using AI Chatbots. *Gadgets To Use*. URL: <https://gadgetstouse.com/blog/2025/01/15/learn-any-language-using-ai-chatbots-for-free/>
25. Mesnard Th. et al. Gemma: Open Models Based on Gemini Research and Technology. *arXiv*. 2024. DOI: <https://doi.org/10.48550/arXiv.2403.08295>
26. Rahman et al. Comparative Analysis Based on DeepSeek, ChatGPT, and Google Gemini: Features, Techniques, Performance, Future Prospects. *ArXiv*. 2025. DOI: <https://arxiv.org/abs/2503.04783>
27. Build apps for Google Chat with Gemini. 18.08.2024. URL: <https://codelabs.developers.google.com/codelabs/chat-apps-gemini>

28. Моделювання програмного забезпечення : навч.-метод. посібник [Електронний ресурс] / уклад.: С.Ю. Манаков, О.Г. Трофименко, Ю.Г. Лобода, А.І. Дика. Одеса: Фенікс, 2023. 145 с. URL: <http://dspace.onua.edu.ua/handle/11300/25952>
29. Getting started with the Gemini API and Web apps. *Google for Developers*. <https://developers.google.com/learn/pathways/solution-ai-gemini-getting-started-web>
30. Gemini. Google DeepMind. Google DeepMind, 2025. URL: <https://deepmind.google/technologies/gemini>
31. Georgiev P. et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv*. 2024. DOI: <https://doi.org/10.48550/arXiv.2403.05530>
32. Трофименко О.Г., Дика А.І., Лобода Ю.Г. Аналіз інструментів тестування вебзастосунків. *Кібербезпека: освіта, наука, техніка*. 2023. № 4(20). С. 62-71. DOI: <https://doi.org/10.28925/2663-4023.2023.20.6271>
33. Gemini API. *Google AI for Developers*. URL: <https://ai.google.dev/gemini-api/docs>
34. Jeon J., Lee S. Large language models in education: A focus on the complementary relationship between human teachers and ChatGPT. *Educ Inf Technol*. 2023. Vol. 28. P. 15873–15892. DOI: <https://doi.org/10.1007/s10639-023-11834-1>
35. Silitonga, L.M et al. The Impact of AI Chatbot-Based Learning on Students' Motivation in English Writing Classroom. *Innovative Technologies and Learning. ICITL. Lecture Notes in Computer Science*. 2023. Vol. 14099. Springer, Cham. DOI: https://doi.org/10.1007/978-3-031-40113-8_53
36. AlSagri, H.S., Farhat, F., Sohail, S.S. et al. ChatGPT or Gemini: Who Makes the Better Scientific Writing Assistant? *J Acad Ethics*. 2025. Vol. 23. P. 1121–1135. DOI: <https://doi.org/10.1007/s10805-024-09549-0>
37. Тестування та забезпечення якості програмних систем : навч. посібник [Електронне видання] / О.Г. Трофименко, А.І. Дика; Нац. ун-т «Одес. юрид. академія». Одеса: Фенікс, 2024. 195 с. URL: <https://hdl.handle.net/11300/27717>

38. Borsci S., Malizia A., Schmettow M., et al. The Chatbot Usability Scale: the Design and Pilot of a Usability Scale for Interaction with AI-Based Conversational Agents. *Pers Ubiquit Comput.* 2022. Vol. 26. P. 95–119. DOI: <https://doi.org/10.1007/s00779-021-01582-9>
39. Тестування та забезпечення якості програмних систем : навч.-метод. посіб. [Електронне видання] / О.Г. Трофименко, А.І. Дика; Нац. ун-т «Одес. юрид. академія». Одеса: Фенікс, 2024. 140 с. URL: <https://hdl.handle.net/11300/27246>
40. Ni'mah I., Fang M., Menkovski V., Pechenizkiy M. NLG Evaluation Metrics Beyond Correlation Analysis: An Empirical Metric Preference Checklist. *arXiv*. 2023. DOI: <https://doi.org/10.48550/arXiv.2305.08566>.
41. Hu T. Unveiling LLM Evaluation Focused on Metrics: Challenges and Solutions. *arXiv*. 2024. <https://arxiv.org/abs/2404.09135>
42. Liu Y., Iter D., Xu Y., Wang Sh., Xu R., Zhu Ch. G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment. *arXiv*. 2023. DOI: <https://doi.org/10.48550/arXiv.2303.16634>.

ДОДАТКИ

Додаток А. Технічна специфікація запитів та відповідей Gemini API для трьох навчальних сценаріїв

Структура JSON-запиту «Словникова вправа»:

```
{
  "contents": [
    {
      "role": "user",
      "parts": [
        {
          "text": "Define the word 'resilient', give an example sentence, and create a fill-in-the-blank exercise."
        }
      ]
    }
  ],
  "generationConfig": {
    "temperature": 0.6,
    "topP": 0.9,
    "topK": 40,
    "maxOutputTokens": 512
  },
  "safetySettings": [
    { "category": "HARM_CATEGORY_HARASSMENT", "threshold": 3 },
    { "category": "HARM_CATEGORY_HATE_SPEECH", "threshold": 3 },
    { "category": "HARM_CATEGORY_SEXUALLY_EXPLICIT", "threshold": 4 },
    { "category": "HARM_CATEGORY_DANGEROUS_CONTENT", "threshold": 3 }
  ]
}
```

Відповідь Gemini API на цей запит у форматі JSON:

```
{
  "candidates": [
    {
      "content": {
        "role": "model",
        "parts": [
          {
            "text": " Word: resilient\n Definition: Able to recover quickly from difficulties or adapt to change.\n Example: Despite the setback, she remained resilient and continued working hard.\n Exercise: Fill in the blank - 'After losing the match, the team stayed _____ and trained even harder.'"

```

Структура JSON-запиту «Граматичне пояснення»:

```
{
  "contents": [
    {
      "role": "user",
      "parts": [
        {
          "text": "He don't like apples."
        }
      ]
    }
  ],
  "generationConfig": {
    "temperature": 0.5,
    "topP": 0.9,
    "topK": 40,
    "maxOutputTokens": 256
  },
  "tools": [
    {
      "functionDeclarations": [
        {
          "name": "grammar_assess",
          "description": "Evaluate grammar and suggest corrections",
          "parameters": {
            "type": "object",
            "properties": {
              "text": {
                "type": "string",
                "description": "Text to be evaluated"
              }
            }
          },
          "required": ["text"]
        }
      ]
    }
  ]
}
```

Відповідь Gemini API на цей запит у форматі JSON:

```
{
  "candidates": [
    {
      "content": {
        "role": "model",
        "parts": [
          {
            "text": " Error: 'He don't like apples.'\n Correction: 'He doesn't like apples.'\n Explanation: In the third person singular (he/she/it), use 'doesn't' instead of 'don't' in Present Simple."
          }
        ]
      }
    }
  ]
}
```

Структура JSON-запиту «Симуляція діалогу»:

```
{
  "contents": [
    {
      "role": "user",
      "parts": [
        {
          "text": "Simulate a short dialogue between a customer and a
waiter in a restaurant."
        }
      ]
    }
  ],
  "generationConfig": {
    "temperature": 0.8,
    "topP": 0.95,
    "topK": 50,
    "maxOutputTokens": 512
  },
  "safetySettings": [
    { "category": "HARM_CATEGORY_HARASSMENT", "threshold": 3 },
    { "category": "HARM_CATEGORY_HATE_SPEECH", "threshold": 3 },
    { "category": "HARM_CATEGORY_SEXUALLY_EXPLICIT", "threshold": 4 },
    { "category": "HARM_CATEGORY_DANGEROUS_CONTENT", "threshold": 3 }
  ]
}
```

Відповідь Gemini API на цей запит у форматі JSON:

```
{
  "candidates": [
    {
      "content": {
        "role": "model",
        "parts": [
          {
            "text": " Customer: Hello, I'd like to order a coffee,
please.\n Waiter: Sure! Would you like milk or sugar with that?\n
Customer: Just milk, thank you.\n Waiter: Coming right up!"
          }
        ]
      }
    }
  ]
}
```

Додаток Б. Python-код генерації JSON-запиту до Gemini API

```

import json

def build_gemini_request(scenario, user_input):
    base_request = {
        "contents": [
            {
                "role": "user",
                "parts": [
                    {"text": user_input}
                ]
            }
        ],
        "generationConfig": {
            "temperature": 0.6,
            "topP": 0.9,
            "topK": 40,
            "maxOutputTokens": 512
        },
        {
            "category": "HARM_CATEGORY_HARASSMENT", "threshold": 3},
        {"category": "HARM_CATEGORY_HATE_SPEECH", "threshold": 3},
        {"category": "HARM_CATEGORY_SEXUALLY_EXPLICIT", "threshold": 4},
        {"category": "HARM_CATEGORY_DANGEROUS_CONTENT", "threshold": 3}
    ]
}

if scenario == "Граматичне пояснення":
    base_request["generationConfig"]["temperature"] = 0.5
    base_request["generationConfig"]["maxOutputTokens"] = 256
    base_request["tools"] = [
        {
            "functionDeclarations": [
                {
                    "name": "grammar_assess",
                    "description": "Evaluate grammar and suggest
corrections",
                    "parameters": {
                        "type": "object",
                        "properties": {
                            "text": {
                                "type": "string",
                                "description": "Text to be
evaluated"
                            }
                        }
                    },
                    "required": ["text"]
                }
            ]
        }
    ]

elif scenario == "Симуляція діалогу":

```

```

        base_request["generationConfig"]["temperature"] = 0.7
        base_request["generationConfig"]["topP"] = 0.95
        base_request["generationConfig"]["maxOutputTokens"] = 640

    elif scenario == "Словникова вправа":
        base_request["generationConfig"]["temperature"] = 0.6
        base_request["generationConfig"]["maxOutputTokens"] = 512

    return json.dumps(base_request, indent=2, ensure_ascii=False)

# Приклад використання:
print(build_gemini_request("Словникова вправа", "Create a
vocabulary exercise using the word 'resilient'. Include
definition, example sentence, and a fill-in-the-blank task."))

```

Додаток В. Фрагменти Python-коду модуля використання Gemini API у чат-боті для вивчення англійської мови

models.py – типи запитів:

```
from pydantic import BaseModel

class TranslationRequest(BaseModel):
    source_text: str
    source_lang: str = "uk"
    target_lang: str = "en"
    level: str = "intermediate"

class VocabularyRequest(BaseModel):
    word: str

class GrammarRequest(BaseModel):
    sentence: str

class DialogueRequest(BaseModel):
    scenario: str
```

main.py – FastAPI-сервер:

```
from fastapi import FastAPI
from models import VocabularyRequest, GrammarRequest, DialogueRequest
import requests

app = FastAPI(title="LLM Chatbot for English Learning")

GEMINI_API_URL =
"https://generativelanguage.googleapis.com/v1beta/models/gemini-1.5-
pro:generateContent"
GEMINI_API_KEY = "YOUR_API_KEY" # Замінити на реальний ключ

headers = {
    "Content-Type": "application/json"
}

safety_settings = [
    {"category": "HARM_CATEGORY_HARASSMENT", "threshold": 3},
    {"category": "HARM_CATEGORY_HATE_SPEECH", "threshold": 3},
    {"category": "HARM_CATEGORY_SEXUALLY_EXPLICIT", "threshold": 4},
    {"category": "HARM_CATEGORY_DANGEROUS_CONTENT", "threshold": 3}
]

@app.post("/vocabulary")
def generate_vocabulary_exercise(req: VocabularyRequest):
    prompt = (
        f"Create a vocabulary exercise using the word '{req.word}' for
intermediate English learners. "
        f"Include definition, example sentence, and a fill-in-the-blank task."
    )
```

```

    payload = {
        "contents": [{"role": "user", "parts": [{"text": prompt}]}],
        "generationConfig": {"temperature": 0.6, "topP": 0.9, "topK": 40,
"maxOutputTokens": 512},
        "safetySettings": safety_settings
    }
    response = requests.post(f"{GEMINI_API_URL}?key={GEMINI_API_KEY}",
headers=headers, json=payload)
    return response.json()

@app.post("/grammar")
def grammar_explanation(req: GrammarRequest):
    prompt = (
        f"Please explain the grammatical mistake in the sentence:
'{req.sentence}'. "
        f"Suggest a corrected version and explain the rule."
    )
    payload = {
        "contents": [{"role": "user", "parts": [{"text": prompt}]}],
        "generationConfig": {"temperature": 0.5, "topP": 0.85, "topK": 40,
"maxOutputTokens": 256},
        "safetySettings": safety_settings,
        "tools": [
            {
                "functionDeclarations": [
                    {
                        "name": "grammar_assess",
                        "description": "Evaluate grammar and suggest
corrections",
                        "parameters": {
                            "type": "object",
                            "properties": {
                                "text": {
                                    "type": "string",
                                    "description": "Text to be evaluated"
                                }
                            }
                        },
                        "required": ["text"]
                    }
                ]
            }
        ]
    }
    response = requests.post(f"{GEMINI_API_URL}?key={GEMINI_API_KEY}",
headers=headers, json=payload)
    return response.json()

@app.post("/dialogue")
def simulate_dialogue(req: DialogueRequest):
    prompt = (
        f"Simulate a short dialogue: {req.scenario}. "
        f"Use polite and natural expressions suitable for A2-B1 learners."
    )

```

```

payload = {
    "contents": [{"role": "user", "parts": [{"text": prompt}]}],
    "generationConfig": {"temperature": 0.7, "topP": 0.95, "topK": 50,
"maxOutputTokens": 640},
    "safetySettings": safety_settings
}
response = requests.post(f"{GEMINI_API_URL}?key={GEMINI_API_KEY}",
headers=headers, json=payload)
return response.json()

```

utils.py – генерація запиту до Gemini API:

```

def build_gemini_payload(text: str, source_lang: str, target_lang: str,
level: str):
    prompt = (
        f"Translate the following {source_lang}-language sentence into
{target_lang}, "
        f"providing a version suitable for {level} learners. "
        f"Include a brief explanation of grammar and vocabulary used.\n\n"
        f"Text: {text}"
    )
    return {
        "contents": [{"role": "user", "parts": [{"text": prompt}]}],
        "generationConfig": {"temperature": 0.6, "topP": 0.9, "topK": 40,
"maxOutputTokens": 512},
        "safetySettings": [
            {"category": "HARM_CATEGORY_HARASSMENT", "threshold": 3},
            {"category": "HARM_CATEGORY_HATE_SPEECH", "threshold": 3},
            {"category": "HARM_CATEGORY_SEXUALLY_EXPLICIT", "threshold": 4},
            {"category": "HARM_CATEGORY_DANGEROUS_CONTENT", "threshold": 3}
        ]
    }

```

Додаток Г. Копії документів про апробацію результатів роботи



Інформаційні технології: моделі, алгоритми, системи – 2025

VI Міжнародна науково-практична інтернет конференція

Головна Інформаційне повідомлення Публікації Мої публікації Вийти



ПЕРЕВАГИ І РИЗИКИ ІНТЕГРАЦІЇ LLM В ОСВІТНІ ПЛАТФОРМИ

Автори: Трофименко О.Г., Алі А.Ф.

Ключові слова: LLM; освітні платформи, Gemini; GPT-4; цифрова освіта

УДК 004.8:37.018.43

ПЕРЕВАГИ І РИЗИКИ ІНТЕГРАЦІЇ LLM В ОСВІТНІ ПЛАТФОРМИ

Трофименко О.Г. к.т.н., доц.¹; Алі А.Ф.²

^{1,2} Національний університет «Одеська юридична академія»

^{1,2} Україна, Одеса

¹ORCID: 0000-0001-7626-0886; ²aliamir122@gmail.com

Анотація. Розглянуто потенціал інтеграції великих мовних моделей (LLM) у сучасні освітні платформи. На основі емпіричних досліджень проаналізовано переваги та окреслено ризики LLM у персоналізації навчання.

Ключові слова: LLM; освітні платформи, Gemini; GPT-4; цифрова освіта.

Вступна частина
делей (Large Language
трансформації навчаль
ципи персоналізоване
римку студентів у реалі
зкою ризиків: від етич
академічній автономії
використання LLM у

обчислювальних ресурсів, стабільного інтернет-з'єднання та захищених каналів передачі даних. У контексті освітніх платформ це означає додаткові витрати на інфраструктуру та кібербезпеку. Крім того, питання конфіденційності студентських даних залишається відкритим: інтеграція сторонніх API може порушувати принципи GDPR та локальні освітні регламенти.

Рекомендації щодо інтеграції LLM в освітні платформи мають ґрунтуватися на педагогічній доцільності, етичній відповідальності та технічній обґрунтованості. Передусім, застосування LLM не має розглядатися як заміна викладача, а як інструмент підтримки навчального процесу. Так, студенти, які навчаються у змішаному форматі, тобто з використанням LLM у поєднанні з викладацьким супроводом, демонструють вищі результати у відкритих завданнях, ніж ті, хто працює виключно з моделями [1]. Це підтверджує потребу педагогічного контролю над використанням штучного інтелекту в навчанні. Крім того, критично важливо забезпечити прозорість щодо джерел знань, на яких базується модель, а також механізмів генерації контенту. В освітньому середовищі довіра до джерела інформації є фундаментальною, особливо коли йдеться про дисципліни, де точність має вирішальне значення.

Етична складова інтеграції LLM також не може бути проігнорована. Моделі повинні відповідати нормам конфіденційності [4], зокрема вимогам GDPR, FERPA та локальних освітніх регламентів. Збір і обробка персональних даних студентів мають здійснюватися лише за умови явної згоди, а використання сторонніх API – проходити етичний аудит, адже впровадження LLM у навчальні середовища без належної етичної оцінки створює ризики порушення приватності та академічної автономії.

Технічна реалізація LLM в освітніх платформах має враховувати обмеження інфраструктури. Оптимізовані версії моделей, такі як Gemini Nano, можуть бути корисними для мобільних платформ, а офлайн-режими або кешування контенту – критичними для регіонів з нестабільним інтернетом. Водночас, ефективне використання LLM потребує розробки методичних протоколів, які допоможуть формувувати запити та інтерпретувати відповіді. Це сприятиме розвитку критичного мислення і зменшить ризик поверхневого засвоєння знань.

Висновки. Загалом LLM-моделі мають потенціал радикально покращити якість освіти, але їхня інтеграція має супроводжуватися нормативно-етичним контролем, педагогічною адаптацією та технічною верифікацією. Рекомендовано враховувати стандартизовані протоколи інтеграції LLM у навчальні середовища з урахуванням можливих ризиків. Реалізація LLM з урахуванням зазначених рекомендацій дозволить максимально використати потенціал моделей, мінімізувати ризики і забезпечити якісне, безпечне й ефективне освітнє середовище.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Sharma, S., Mittal, P., Kumar, M. et al. (2025). The role of large language models in personalized learning: a systematic review of educational impact. *Discov Sustain*, 6, 243. <https://doi.org/10.1007/s43621-025-01094-z>
- [2] Kumar, P. (2024). Large language models (LLMs): survey, technical frameworks, and future challenges. *Artif Intell Rev*, 57, 260. <https://doi.org/10.1007/s10462-024-10888-y>