

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління

БАКАЛАВРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

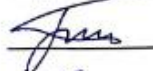
на тему:

**«РОЗРОБКА ІНТЕЛЕКТУАЛЬНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ
УПРАВЛІННЯ НАВЧАЛЬНИМ ПРОЦЕСОМ ІЗ ВИКОРИСТАННЯМ
МЕТОДІВ АВТОМАТИЗАЦІЇ ТА АНАЛІЗУ ДАНИХ.
ЧАСТИНА 2.ПІДСИСТЕМА АВТОМАТИЗАЦІЇ ЕТАПІВ ДОСЛІДНИЦЬКОЇ
РОБОТИ»**

Виконав: студент 4 курсу, групи 2АКІТ-216
спеціальності 151 – Автоматизація та
комп'ютерно-інтегровані технології

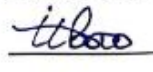
 **Дмитро КОРЧОВСЬКИЙ**
(ім'я та ПРИЗВИЩЕ)

Керівник: к.т.н., професор каф. КСУ

 **Микола БИКОВ**
(ім'я та ПРИЗВИЩЕ)

« 10 » червня 2025 р.

Рецензент: к.т.н., доцент каф. АІТ

 **Юрій ІВАНОВ**
(ім'я та ПРИЗВИЩЕ)

« 11 » червня 2025 р.

Допущено до захисту
Завідувач кафедри КСУ
д.т.н., проф.

 **В'ячеслав КОВТУН**
(ім'я та ПРИЗВИЩЕ)

« 12 » червня 2025 р.

Вінниця ВНТУ - 2025 рік

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління
Рівень вищої освіти I-й (бакалаврський)
Галузь знань – 15 Автоматизація та приладобудування
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
Освітньо-професійна програма Інтелектуальні комп'ютерні системи управління

ЗАТВЕРДЖУЮ

Завідувач кафедри КСУ

д.т.н., проф.

 **Вячеслав КОВТУН**

« 26 » 05 2025 року

ЗАВДАННЯ

НА БАКАЛАВРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Корчовському Дмитру Костянтиновичу

(прізвище, ім'я, по батькові)

1. Тема роботи Розробка інтелектуальної комп'ютерної системи управління навчальним процесом із використанням методів автоматизації та аналізу даних
Частина 2. Підсистема автоматизації етапів дослідницької роботи.
керівник роботи Биков Микола Максимович, професор, к.т.н. каф. КСУ
затверджені наказом вищого навчального закладу від “20” 03 2025 року №97
2. Термін подання студентом роботи «17» 06 2025 року
3. Джерела розробки: Бакалаврська кваліфікаційна робота виконується вперше.
В ході проведення розробки повинні використовуватись такі документи:
 1. Биков М.М., Бикова О.М, Корчовський Д.К., Маняк В.О. Аналіз методів автоматизації навчального процесу з використанням штучного інтелекту. - [Електронний ресурс] – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2025/paper/view/24426/20212>
 2. Lee I., Lee J., Lee K., Lee J., Hwang J. Effectiveness of artificial intelligence in education: A systematic review, *Journal of Educational Technology & Society*, 22(3), 2019, p. 47-63.
 3. Алла Коломієць, Олександра Юріївна Кушнір, “Використання штучного інтелекту в освітній та науковій діяльності: можливості та виклики”, *Modern Information Technologies and Innovation Methodologies of Education in Professional Training Methodology Theory Experience Problems*, Feb 2024. <https://doi.org/10.31652/2412-1142-2023-70-45-57>
4. Зміст текстової частини: Анотація; Зміст; Вступ; Техніко-економічне одґрунтування розробки; Розробка структури системи і вибір сервісів ІІІ; Розробка програмного забезпечення; Машинний експеримент; Висновки.
5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових

креслень): Концептуальна схема системи управління освітнім процесом-плакат демонстраційний; Схема структурна підсистеми автоматизації етапів дослідницької роботи – А4; UML-діаграма послідовності навчального процесу – А4; UML-діаграма послідовності етапів дослідницької роботи – А4; UML-діаграма використання системи – А4; Схема алгоритму роботи системи – А4; Результати машинного експерименту – плакат демонстраційний

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	виконання прийняв

7. Дата видачі завдання «26» 05 2025 року

КАЛЕНДАРНИЙ ПЛАН

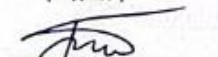
№ з/п	Назва етапів бакалаврської дипломної роботи	Термін виконання етапів роботи	Примітка
1	Розробка технічного завдання	10 травня 2025 р.	
2	Аналіз об'єкта навчального процесу як об'єкта автоматизації	15 травня 2025 р.	
3	Вибір сервісів ШІ і обладнання для інтелектуальної системи управління	18 травня 2025 р.	
4	Розробка структурної схеми системи	26 травня 2025 р.	
5	Розробка алгоритму роботи системи	28 травня 2025 р.	
6	Розробка програмного забезпечення системи	02 червня 2025 р.	
7	Машинний експеримент	9 червня 2025 р.	
8	Оформлення пояснювальної записки	10 червня 2025 р.	
9	Графічні матеріали: Концептуальна схема системи управління дослідницькою роботою – плакат Схема структурна підсистеми автоматизації Схема алгоритму роботи системи UML-діаграми Результати машинного експерименту	03 червня 2025 р. 05 червня 2025 р. 07 червня 2025 р. 09 червня 2025 р. 09 червня 2025 р.	
10	Попередній захист	15 червня 2025 р.	
11	Остаточний захист	18 червня 2025 р.	

Студент


(підпис)

Дмитро КОРЧОВСЬКИЙ

Керівник роботи


(підпис)

Микола БИКОВ

АНОТАЦІЯ

Бакалаврська кваліфікаційна робота містить 75 сторінок формату А4, на яких є 13 рисунків, 8 таблиць та список використаних джерел з 50 найменуваннями.

Виконання даної комплексної бакалаврської кваліфікаційної роботи має за мету підвищити ефективність університетського навчального процесу шляхом побудови інтелектуальної комп'ютерної системи управління для автоматизації етапів навчальної і дослідницької роботи викладача за рахунок використання сучасних інструментів і технологій штучного інтелекту (ШІ). В даній Частині 2 комплексної роботи розроблено підсистему автоматизації етапів науково-дослідницької роботи для підвищення ефективності наукової роботи викладача.

На основі проведеного аналізу університетського навчального процесу як об'єкта автоматизації, визначено сучасні виклики в застосуванні технологій штучного інтелекту для автоматизації завдань науково-дослідної роботи, визначено перелік задач для досягнення поставленої мети.

Запропоновано концептуальну модель схемної побудови ефективної підсистеми автоматизації завдань науково-дослідницької роботи викладача університету за рахунок використання сервісів і моделей штучного інтелекту. Розроблено структуру підсистеми автоматизації з використанням локальних ресурсів на платформах LM Studio і Hugging Face і веб-інтерфейсу для викоистання хмарних сервісів. Розроблено UML-діаграми послідовності етапів науково-дослідної роботи та UML-діаграми варіантів використання розробленої системи, а також програмне забезпечення підсистеми автоматизації дослідницької діяльності.

В ілюстративній частині бакалаврської роботи подані креслення структурної схеми розробленої системи, UML-діаграми, демонстраційні плакати, що пояснюють принципи функціонування підсистеми автоматизації наукової діяльності викладача на основі використання технологій ШІ.

Ключові слова: навчальний процес, автоматизація етапів процесу навчання, машинне навчання, штучний інтелект, інтелектуальні комп'ютерні системи

ABSTRACT

The bachelor's thesis contains 75 pages of A4 format, which contain 13 figures, 8 tables and a list of used sources with 50 names.

The implementation of this comprehensive bachelor's thesis aims to increase the efficiency of the university educational process by building an intelligent computer control system to automate the stages of the teacher's educational and research work through the use of modern tools and technologies of artificial intelligence (AI). In this Part 2 of the comprehensive work, a subsystem for automating the stages of scientific and research work has been developed to increase the efficiency of the teacher's scientific work.

Based on the analysis of the university educational process as an object of automation, modern challenges in the application of artificial intelligence technologies for automating tasks of scientific and research work have been identified, a list of tasks has been defined to achieve the set goal.

A conceptual model of the schematic construction of an effective subsystem for automating tasks of a university teacher's research work through the use of artificial intelligence services and models has been proposed. The structure of the automation subsystem has been developed using local resources on the LM Studio and Hugging Face platforms and a web interface for using cloud services. UML diagrams of the sequence of stages of research work and UML diagrams of the use cases of the developed system have been developed, as well as the software of the research activity automation subsystem.

The illustrative part of the bachelor's thesis presents drawings of the structural diagram of the developed system, UML diagrams, and demonstration posters that explain the principles of the functioning of the subsystem for automating the teacher's research activity based on the use of AI technologies.

Keywords: learning process, automation of learning process stages, machine learning, artificial intelligence, intelligent computer systems.

ЗМІСТ

ВСТУП.....	4
1 ОБГРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ СИСТЕМИ АВТОМАТИЗАЦІЇ ЕТАПІВ ДОСЛІДНИЦЬКОЇ РОБОТИ	7
1.1 Аналіз процесу наукових досліджень і визначення його етапів	7
1.2 Аналіз сучасного стану впровадження технологій ІІІ в процес наукової діяльності	9
1.3 Опис інструментів ІІІ для автоматизації дослідницької роботи	11
1.4 Вибір аналогів для розроблюваної підсистеми автоматизації етапів дослідницької роботи.....	17
1.5 Концептуальні основи створення комп'ютерної системи для автоматизації етапів наукової роботи в університеті	20
1.6 Висновки до розділу	21
2 ВИБІР ТА ІНТЕГРАЦІЯ ІНСТРУМЕНТІВ ІІІ В СИСТЕМУ АВТОМАТИЗАЦІЇ НАУКОВОЇ РОБОТИ	23
2.1 Вибір ефективного інструментарію ІІІ для проведення наукових досліджень	23
2.1.1. Відповідність інструментів ІІІ критеріям ефективності.....	25
2.1.2. Автоматизація етапів наукового дослідження за допомогою обраних інструментів ІІІ.....	28
2.2 Обґрунтування вибору програмних платформ для інтеграції сервісів ІІІ в систему автоматизації.....	30
2.2.1 Інтеграція через хмарні Сервіси та API	31
2.2.2 Вибір платформи з репозиторієм моделей і сервісів ІІІ	32
2.2.3 Вибір локальної платформи для запуску моделей ІІІ на локальному комп'ютері	34
2.3 Розробка структурної схеми підсистеми автоматизації наукової роботи.....	37
2.4 Висновки до розділу	40
3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	41
3.1 Обґрунтування вибору програмного середовища Python.....	41

	3
3.2 Опис розроблених програмних функцій.....	42
3.3 Розробка програми веб-інтерфейсу Open WebUI	46
3.4 Висновки до розділу	50
ВИСНОВКИ	51
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	52
ДОДАТКИ.....	57
Додаток А (обов'язковий) Протокол перевірки кваліфікаційної роботи	58
Додаток Б (обов'язковий) Технічне завдання	59
Додаток В (довідковий) Лістинг програм.....	62
Додаток Г (обов'язковий) Ілюстративна частина.....	65

ВСТУП

Актуальність теми. Застосування штучного інтелекту (ШІ) відкриває нові горизонти в сферах освіти, науки, медицини, бізнесу, адміністративного управління тощо, сприяючи підвищенню ефективності, точності та продуктивності роботи. Тому використання штучного інтелекту в сучасному суспільстві є не просто важливим, а критично необхідним для розвитку технологій, економіки та соціальних процесів. Ступінь використання ШІ може бути одним з індикаторів потенціалу сучасної держави [1].

Одним із ключових напрямів застосування ШІ для модернізації освіти є використання систем управління навчанням (Learning Management Systems, LMS), так званих навчальних платформ, які дозволяють динамічно розв'язувати проблеми, пов'язані з викликами дистанційного навчання, подачі навчального матеріалу, забезпечення взаємної комунікації між викладачами, студентами і адміністрацією, контролю знань студентів, архівуванням навчальної і адміністративної документації [2]. Подальшим розвитком даних навчальних платформ може стати застосування в них інструментів штучного інтелекту для підвищення ефективності наукової роботи викладачів і студентів ЗВО. Очікується, що до 2030 року до 60% рутинної роботи викладача буде автоматизовано, що дозволить зосередитися на креативних, дослідницьких та комунікаційних аспектах навчання [3].

Викладачі університетів стикаються зі зростаючим тиском щодо балансування викладацької, дослідницької та адміністративної відповідальності в умовах дедалі більш конкурентного глобального академічного середовища. Впровадженням інноваційних цифрових інструментів та послуг на основі ШІ допомагає викладачам перенаправити свій час та творчу енергію на високоефективну дослідницьку діяльність [4].

Застосування сервісів штучного інтелекту в функціонування LMS забезпечує основу для прийняття рішень в наукових дослідженнях на основі даних. Використовуючи передові інструменти ШІ в складі LMS викладачі можуть приймати обґрунтовані рішення щодо напрямків досліджень, заявок на гранти та

спільних проектів [4] . Крім того, системи рекомендацій на базі штучного інтелекту можуть допомогти у виборі відповідної літератури, тим самим допомагаючи дослідникам швидко визначати високоякісні джерела та уникати зайвих досліджень, що сприяє більш інноваційним та своєчасним науковим внескам.

На сьогодні одним із викликів для впровадження технологій ШІ в роботу інформаційних систем управління навчально - науковим процесом є відсутність на даний період підсистем автоматизації етапів дослідницької роботи викладачів і студентів з застосуванням комп'ютерних систем управління, що використовують в своїй роботі технології ШІ.

Тому дослідницькі роботи, орієнтовані на впровадження методів і технологій ШІ в інформаційні системи управління університетським освітнім процесом та розробку програмних і апаратних засобів для їхньої реалізації , є на сьогодні актуальною проблемою.

Мета даної роботи - підвищити ефективність університетського освітнього процесу шляхом розробки та впровадження підсистеми управління дослідницькою роботою на основі технологій штучного інтелекту.

Досягнення поставленої мети вимагає вирішення таких **задач**:

- провести аналіз існуючих технологій та інструментів штучного інтелекту, які можна використати в науковій роботі;
- визначити основні завдання дослідницької наукової діяльності викладача університету, які потрібно автоматизувати розроблюваною в бакалаврській роботі системою;
- надати обґрунтування вибору сервісів і моделей ШІ, які потрібно застосувати для автоматизації визначених дослідницьких завдань;
- обґрунтувати способи застосування інструментів ШІ в платформах, на основі яких буде реалізована розроблювана система;
- визначити технічні характеристики апаратного обладнання, потрібного для реалізації розроблюваної системи;
- визначити методи застосування обраних сервісів і моделей штучного

інтелекту в розроблюваній системі для автоматизації дослідницьких завдань;

- розробити прототип системи для автоматизації окремих етапів наукової роботи на основі ШІ та програмне забезпечення для її роботи.

Об'єктом дослідження є процес наукової діяльності у закладі вищої освіти.

Предметом дослідження є моделі і сервіси ШІ та методи їх застосування для автоматизації виконання науково-дослідницьких завдань викладачів і студентів.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалась згідно наукової тематики кафедри КСУ, присвяченої впровадженню інноваційних технологій в навчальний процес.

Наукова новизна одержаних результатів полягає в тому, що в роботі запропоновано підхід до впровадження ШІ в інформаційну систему управління навчальним процесом, який уможлиблює автоматизацію рутинних етапів дослідницької діяльності і підвищує креативність наукової роботи викладачів і студентів.

Практична цінність. Отримані в бакалаврській роботі результати можуть бути використані для впровадження технологій ШІ у процес наукових досліджень, що сприятиме підвищенню їх ефективності, зменшенню навантаження на викладачів та підвищенню їх кваліфікації.

Впровадження. Отримані в роботі результати можуть бути використані в навчальних дисциплінах кафедри КСУ при підготовці фахівців у галузі автоматизації, а також в роботі університетської освітньої платформи JetIq

Апробація. Результати роботи були представлені на науково-технічній конференції підрозділів Вінницького національного технічного університету.

Публікації: Авторами роботи опубліковано тези доповіді: «Аналіз методів автоматизації навчального процесу з використанням штучного інтелекту». – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2025/paper/view/24426/20212> .

1 ОБГРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ СИСТЕМИ АВТОМАТИЗАЦІЇ ЕТАПІВ ДОСЛІДНИЦЬКОЇ РОБОТИ

1.1 Аналіз процесу наукових досліджень і визначення його етапів

Впровадження інструментів штучного інтелекту в процес наукової діяльності викладачів і студентів університету дозволяє автоматизувати рутинну роботу під час його проведення, підвищити ефективність і креативність творчого процесу [1,2]. Проведення наукового дослідження – це багатогранний процес, який передбачає послідовне проходження кількох ключових етапів. Ці етапи забезпечують системність, об'єктивність та відтворюваність отриманих результатів, і можуть мати певну гнучкість та ітеративність. Наприклад, у процесі аналізу даних можуть виникнути нові питання, що потребуватимуть повернення до попередніх етапів, уточнення гіпотези або збору додаткових даних. Однак загальна логіка наукового дослідження залишається незмінною, спрямованою на отримання нового, достовірного знання. Проведений аналіз релевантних літературних джерел [2,3,4] дозволив визначити такі основні етапи в проведенні дослідницької роботи.

1) Етап визначення проблеми та теми дослідження. Цей етап поділяється на таку послідовність робіт

- Вибір проблеми: На цьому етапі дослідник ідентифікує актуальну наукову проблему, яка потребує вирішення. Це може бути прогалина в існуючих знаннях, суперечливі дані або нова суспільна чи практична потреба.
- Формулювання теми: Обрана проблема конкретизується у вигляді теми дослідження, яка чітко окреслює предмет та об'єкт дослідження.
- Обґрунтування актуальності: Дослідник доводить важливість та своєчасність обраної теми, її теоретичне та практичне значення.

2) Огляд літератури та теоретичний аналіз вимагає виконання таких робіт:

- Вивчення джерел полягає в проведенні глибокого аналізу наукової літератури (статей, монографій, дисертацій тощо) і джерел з Інтернету з обраної теми для ознайомлення з існуючими теоріями, концепціями, результатами попередніх досліджень та методологічними підходами.
- Визначення ступеня розробленості проблеми встановлює, які аспекти проблеми

вже досліджені, а які потребують подальшого вивчення.

- Формулювання робочої гіпотези - на основі проведеного теоретичного аналізу висувається наукове припущення (гіпотеза), яке пояснює досліджуване явище або пропонує шляхи вирішення проблеми. Висунута гіпотеза має бути такою, щоб її можна було перевірити. Після цього виконують наступний етап.

3) Планування та розробка методології дослідження. Він складається з таких кроків:

- Визначення мети та завдань - формулюється основна мета дослідження та конкретні завдання, які необхідно вирішити для її досягнення.
- Вибір об'єкта та предмета дослідження дозволяє конкретизувати межі досліджуваного явища.
- Розробка програми та методики дослідження полягає у виборі (або розробці) конкретних методів збору та аналізу даних (наприклад, експеримент, спостереження, опитування, аналіз документів, статистичні методи тощо). Як результат складається детальний план проведення дослідження.

4) Проведення дослідження (збір даних) є головним етапом дослідження, яке проводиться.

- Експериментальна (емпірична частина) цього етапу полягає в безпосередньому збиранні даних відповідно до розробленої методики. Це може включати проведення експериментів, опитувань, спостережень, збір статистичної інформації тощо.
- Фіксація даних реалізує ретельну документацію та систематизацію усіх отриманих даних.

5) На етапі обробки та аналізу результатів виконуються такі роботи:

- Систематизація та обробка даних - зібрані дані піддаються обробці, яка може включати їх класифікацію, групування, статистичний аналіз тощо.
- Інтерпретація результатів - проводиться аналіз оброблених даних у світлі поставлених завдань та висунутої гіпотези. Під час цього виявляються закономірності, тенденції, зв'язки в елементах досліджуваного об'єкта чи явища.
- Перевірка гіпотези передбачає формулювання висновку на основі аналізу про підтвердження або спростування висунутої гіпотези.

6) Формулювання висновків та рекомендацій.

- Узагальнення результатів - підводяться підсумки дослідження, формулюються основні висновки, які відповідають поставленим завданням.
- Розробка рекомендацій - пропонуються практичні рекомендації щодо вирішення досліджуваної проблеми або подальших напрямків досліджень.

7) Оформлення результатів дослідження полягає у виконанні таких робіт:

- Написання наукової роботи, під час якого результати дослідження оформлюються відповідно до встановлених вимог у вигляді наукової статті, дисертації, монографії, звіту тощо.
- Структурування роботи – проводиться для надання науковій роботі чіткої структури, що містить вступ, основну частину, висновки, список використаних джерел, додатки).

На даному етапі процес наукового дослідження може закінчуватися, але в більшості випадків він вимагає виконання ще одного етапу.

8) Апробація та впровадження результатів потребує:

- Представлення результатів дослідження науковій спільноті на конференціях, семінарах, публікації у наукових виданнях.
- Впровадження у практику - для досліджень прикладного характеру їх результати впроваджуються у відповідні сфери діяльності.

1. 2 Аналіз сучасного стану впровадження технологій ШІ в процес наукової діяльності

Останнім часом актуальність застосування штучного інтелекту (ШІ) у різноманітних секторах, включаючи освітній, зокрема в наукову діяльність університетської спільноти, суттєво зросло з 2004 по 2023 рік, з особливим піком після 2016 року (рис. 1.1) [5].

Number of AI publications in CS worldwide, 2013–23

Source: AI Index, 2025 | Chart: 2025 AI Index report

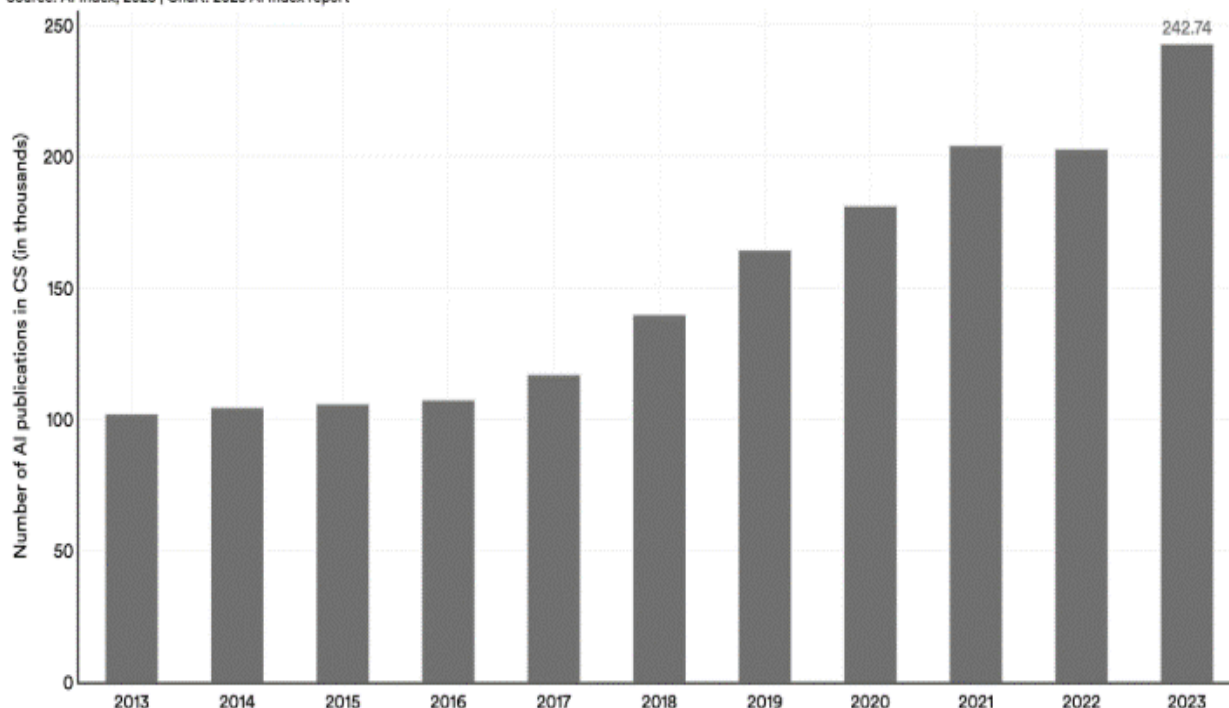


Рисунок 1.1 – Загальна кількість публікацій зі штучного інтелекту за період з 2013 по 2023 рік [5]

Однак в Україні цей процес на сьогодні не отримав достатнього росту, про що свідчать дані прес-релізу досліджень, проведених Київським міжнародним інститутом соціології [1]. Вони показують, що попри активний розвиток технологій на основі ШІ, реальне їх використання серед українців залишається обмеженим. 64% населення не використовують ШІ взагалі. З них видно, що лише кожен восьмий українець активно інтегрував ШІ у своє повсякденне життя, а частка регулярних користувачів (щонайменше раз на тиждень) становить 17% (Таблиця 1.1).

Таблиця 1.1 – Використання ШІ українцями (опитування 2025 рік)

Ступінь використання	Україна	Росія
Регулярно	17%	6%
Іноді	9%	11%
Зараз ні, але знаю про таку можливість	61%	43%
Уперше чую	12%	41%

Ці цифри свідчать про значну різницю у рівнях впровадження та використання ІІІ між Україною та розвиненими країнами. Для того щоб надолужити втрачене та забезпечити стабільне майбутнє, Україна має робити акцент на цифровізації та інтеграції сучасних технологій у всі сфери життя. Важливо активно підтримувати розвиток штучного інтелекту в Україні, інвестуючи у відповідні дослідження та розробки, а також впроваджуючи програми навчання та підвищення кваліфікації для населення. Усвідомлення цього факту робить розробку інтелектуальних систем управління освітнім процесом, зокрема підсистеми автоматизації накової діяльності, своєчасною та актуальною.

Наразі технології штучного інтелекту знаходять застосування в університетах у різних напрямках, демонструючи свій потенціал у покращенні багатьох аспектів навчального процесу. Серед ключових сфер застосування можна виділити персоналізацію навчання через аналіз даних для виявлення прогалин у знаннях та адаптацію навчальних матеріалів [6]. Автоматизовані системи оцінювання та зворотного зв'язку знижують навантаження на викладачів та забезпечують своєчасну інформацію про успішність студентів [7]. ІІІ також використовується для оптимізації адміністративних завдань, таких як складання розкладів, реєстрація на курси та управління фінансовою допомогою [8]. Крім того, інструменти штучного інтелекту допомагають у наукових дослідженнях, аналізуючи великі обсяги академічних публікацій та виявляючи релевантну інформацію [6,9].

1.3 Опис інструментів ІІІ для автоматизації дослідницької роботи

Як було показано в розділі 1.1 роботи, процес наукових досліджень в університеті містить кілька ключових етапів, які можуть бути автоматизовані за допомогою сервісів ІІІ. Проведений авторами аналіз сучасних робіт з дослідження дослідницької діяльності в багатьох університетах [1, 2, 3] дозволив виділити такі основні етапи процесу початкової діяльності викладача (рис. 1.2):



Рисунок 1.2 – Етапи дослідницької роботи

Розширення використання ШІ в наукових дослідженнях сприяє демократизації доступу до передових інструментів, що потенційно вирівнює можливості для дослідників з різних установ та рівнів фінансування [10,11]. Зростаюча інтеграція ШІ в дослідницькі робочі процеси свідчить про потенційну зміну традиційного процесу дослідження, що вимагає від науковців адаптації своїх навичок [12].

На початковому етапі дослідження, ШІ може допомогти генерувати ідеї для дослідницьких задач, уточнити їхню спрямованість та виявити прогалини в сучасних знаннях шляхом аналізу існуючої літератури [13]. У роботі підкреслюється важливість чіткого визначення проблеми, і ШІ може допомогти в цьому процесі, аналізуючи існуючу літературу для визначення областей, що потребують подальшого вивчення [14]. Першим кроком є постановка чіткої та сфокусованої задачі дослідження, і ШІ може допомогти сформулювати такі задачі [15]. Інструменти ШІ, особливо великі мовні моделі (LLM), такі як безкоштовні версії ChatGPT або Claude [10], можуть допомогти в цьому [13]. Початковий етап дослідження, який часто вважається високо креативним і керованим людською інтуїцією, може бути доповнений ШІ для забезпечення більш всебічної основи. Традиційно формулювання задачі значною мірою залежить від досвіду дослідника та огляду літератури. ШІ може розширити цей обсяг, швидко аналізуючи великі

обсяги тексту, потенційно виявляючи пропущені питання або нові напрямки досліджень. Таким чином, допомога ІІІ у формулюванні задач для дослідження може призвести до більш інноваційних та обґрунтованих дослідницьких проєктів [16]. Здатність ІІІ аналізувати існуючий обсяг досліджень [17] ще до формулювання питання підкреслює перехід до більш керованого даними підходу навіть на самому початку наукового процесу.

ІІІ також може допомогти у формулюванні гіпотез, виявляючи взаємозв'язки та закономірності в існуючих даних та літературі. Аналіз попередніх досліджень за допомогою таких інструментів, як Semantic Scholar або Consensus, може виявити потенційні кореляції або причинно-наслідкові зв'язки, які можуть стати основою гіпотези. У роботі [16] наголошується, що хороша гіпотеза призводить до тестованого передбачення, і ІІІ може допомогти сформулювати гіпотези таким чином, щоб полегшити їхнє тестування. У роботі [18] згадується важливість нульової та дослідницької гіпотез, концепцій, які ІІІ може допомогти дослідникам зрозуміти та правильно сформулювати. Здатність ІІІ обробляти та синтезувати інформацію з численних джерел може допомогти дослідникам розробити більш обґрунтовані гіпотези. Використання ІІІ для виявлення потенційних взаємозв'язків для гіпотез може призвести до дослідження менш очевидних або інтуїтивних зв'язків, що потенційно сприятиме більш новаторським дослідженням. Алгоритми ІІІ можуть виявляти закономірності, які можуть пропустити дослідники-люди [18]. Наступний етап передбачає збір доказів або даних для перевірки гіпотези. Хоча повністю автоматизовані безкоштовні інструменти ІІІ для збору первинних даних можуть бути обмежені залежно від галузі, ІІІ може допомогти у виявленні відповідних наборів даних, доступних в Інтернеті [19]. Такі інструменти, як Data Commons, можуть допомогти дослідникам виявити та отримати доступ до загальнодоступних даних. Загальноцільові моделі ІІІ потенційно можуть допомогти в розробці експериментів або опитувань, хоча ретельний контроль з боку людини є важливим для забезпечення методологічної строгості.

Науковці організовують та аналізують зібрані дані для пошуку закономірностей та зв'язків. Кілька безкоштовних інструментів ІІІ можуть

допомогти в аналізі даних [20] . Відкриті мови програмування, такі як R та Python [21] з бібліотеками, такими як Pandas та NumPy, пропонують потужні аналітичні можливості . Безкоштовні або обмежені версії платформ аналізу даних на основі ШІ, такі як Google AI Studio [22], Julius AI [23] та Numerous.ai [24], пропонують функції розмовного аналізу даних та візуалізації. Такі інструменти, як Orange3 та KNIME Analytics Platform, надають візуальні робочі процеси для аналізу даних, навіть для користувачів з обмеженими знаннями програмування [25]. ChatGPT також можна використовувати для базового аналізу даних та генерування фрагментів коду для складнішого аналізу в інших інструментах

На основі аналізу даних науковці визначають, чи підтверджують докази їхню гіпотезу, чи суперечать їй. Інструменти ШІ можуть допомогти на цьому етапі, підсумовуючи результати аналізу даних та допомагаючи інтерпретувати їхнє значення щодо початкової гіпотези [26]. Наприклад, функції звітування на основі ШІ в таких інструментах, як Julius AI, можуть генерувати резюме результатів аналізу. Також їх можна використовувати для обговорення висновків у контексті виявленої на етапі огляду літератури.

Останній етап передбачає поширення результатів дослідження серед наукової спільноти та широкої громадськості через публікації, презентації тощо. Безкоштовні інструменти ШІ можуть допомогти в написанні наукових робіт (розглядається в окремому розділі), створенні презентацій (наприклад, DrLambda, SlideSpeak, MagicSlides.app, згадані в [27,28] і навіть генеруванні резюме для різних аудиторій. Інструмент ШІ SciSpace допомагає в створенні відео для просування результатів дослідження в соціальних мережах [29].

Оскільки зі звіту [5] випливає, що сучасні малі безкоштовні моделі ШІ доганяють за своїми можливостями великі комерційні моделі ШІ, то в даній роботі вибір інструментів ШІ для потенційного використання в розроблюваній інтелектуальній системі управління навчальним процесом університету зорієнтовано на безкоштовні моделі. В таблицях 1.2 і 1.3 наведено приклади безкоштовних інструментів для автоматизації етапів огляду літератури і написання наукових робіт відповідно.

Таблиця 1.2 - Безкоштовні інструменти ШІ для огляду літератури [25-40]

Інструмент	Ключові функції	Доступність безкоштовної версії
Semantic Scholar	Семантичний пошук, аналіз цитування, резюме ШІ, візуалізація мережі	Безкоштовно
Research Rabbit	Візуальні карти літератури, відстеження статей, система рекомендацій	Безкоштовно
Elicit	Синтез досліджень, відповіді на питання, вилучення інформації	Базовий план безкоштовний
Connected Papers	Інтерактивні графіки зв'язків між статтями	Безкоштовно з обмеженнями
Consensus	Пошук на основі ШІ, індикатор консенсусу, інформація на рівні статті	Безкоштовно з обмеженнями
Scholarcy	Резюмування наукових робіт на основі ШІ	Безкоштовно з обмеженнями
SciSpace	Резюме статей, пояснення, інструменти цитування,	Безкоштовно
Undermind	Уточнення дослідницьких питань, виявлення відповідних статей	Безкоштовно з обмеженнями
Інструмент	Ключові функції	Доступність безкоштовної версії
Litmaps	Візуальне відображення цитувань, відстеження літератури	Безкоштовний рівень
OpenAIRE	Відкритий репозиторій наукових публікацій	Безкоштовно
ExplainPaper	Резюмування та візуалізація складних наукових робіт	Безкоштовно
Iris.ai	Концептуальний пошук, міждисциплінарне відображення	Безкоштовна пробна версія
Paper Digest	Генерує рецензії на наукові статті	Обмежене безкоштовне використання
Studyflo	Пошук статей за допомогою запитань	Безкоштовно
PICO Portal	Інструмент ШІ для огляду літератури	Безкоштовно
R Discovery	Великий репозиторій безкоштовних статей, функції прослуховування та перекладу	Безкоштовно
Semantic Scholar	Семантичний пошук, аналіз цитування, резюме ШІ, візуалізація мережі	Безкоштовно
Research Rabbit	Візуальні карти літератури, відстеження статей, система рекомендацій	Безкоштовно
Elicit	Синтез досліджень, відповіді на питання, вилучення інформації	Базовий план безкоштовний
Consensus	Пошук на основі ШІ, індикатор консенсусу, інформація на рівні статті	Безкоштовно з обмеженнями
Scholarcy	Резюмування наукових робіт на основі ШІ	Безкоштовно з обмеженнями

Таблиця 1.2 – (продовження)

SciSpace	Резюме статей, пояснення, інструменти цитування, інтелектуальний пошук	Безкоштовно з обмеженнями
Undermind	Уточнення дослідницьких питань, виявлення відповідних статей	Безкоштовно з обмеженнями
Keenious	Рекомендації академічних статей на основі завантажених документів	Безкоштовно з обмеженнями
Litmaps	Візуальне відображення цитувань, відстеження літератури	Безкоштовно
OpenAIRE	Відкритий репозиторій наукових публікацій	Безкоштовно
ExplainPaper	Резюмування та візуалізація складних наукових робіт	Безкоштовно
Zotero	Керування бібліографією, організація та створення цитувань	Безкоштовно
Paper Digest	Генерує рецензії на наукові статті	Обмежене безкоштовне
Studyflo	Пошук статей за допомогою запитань	Безкоштовно
PICO Portal	Інструмент III для огляду літератури	Безкоштовно
R Discovery	Великий репозиторій безкоштовних статей, функції прослуховування та перекладу	Безкоштовно

Таблиця 1.3 - Безкоштовні інструменти III для написання робіт [21-44]

Інструмент	Ключові функції	Доступність безкоштовної версії	Допомога з цитуванням
Jenni AI	Помічник з написання на основі III, генерація ідей, допомога з цитуванням, автозавершення на основі III	Безкоштовний план	Так
Paperpal	Перевірка граматики, перефразування, перевірка на плагіат, допомога з цитуванням	Безкоштовний план	Так
SciSpace AI Writer	Написання за допомогою III, генерація контенту, редагування, автоматичне цитування	Безкоштовно з обмеженнями	Так
Paperguide AI	Генерація контенту, створення планів, автоматичне цитування в різних стилях	Безкоштовна пробна версія	Так
Scribbr	Інструмент для підсумовування на основі III, програма для перевірки правопису на основі III	Безкоштовні ресурси	Ні
Blainy	Керування цитуваннями, генератор тез, інструмент для підсумовування на основі III, перефразування	Безкоштовний план	Так
QuillBot	Перефразування, перевірка граматики, інструмент для створення цитувань, інструмент для підсумовування	Безкоштовно з обмеженнями	Так
Wordvice AI	Перевірка правопису, перефразування, перевірка на плагіат	Безкоштовно (пробна версія)	Ні
Zotero	Керування бібліографією, створення цитувань	Безкоштовно	Так

Таблиця 1.3 – (продовження)

DeepSeek	Загальноцільова мовна модель для допомоги у написанні	Безкоштовно	Ні
ChatGPT	Генерування ідей, створення планів, написання чернеток, перевірка	Безкоштовна версія	Обмежено
Claude	Генерування ідей, написання чернеток	Безкоштовна версія	Обмежено

1.4 Вибір аналогів для розроблюваної підсистеми автоматизації етапів дослідницької роботи

Як показує проведений аналіз джерел інформації, в Україні на сьогодні розробок систем автоматизації освітнього процесу, зокрема, автоматизації науково-дослідницької роботи, з використання інструментів ШІ не існує. В якості аналога можна обрати систему автоматизації наукових досліджень, представленою дослідницькою командою Google Research в роботах [45,46,47].

Цей колектив представив співнауковця зі штучним інтелектом, багатоагентну систему штучного інтелекту, побудовану за допомогою Gemini 2.0, як віртуального наукового співробітника (co-scientist), який допомагає вченим генерувати нові гіпотези та дослідницькі пропозиції, а також пришвидшує частоту наукових відкриттів. Структура даної системи представлена на рис. 1.3.

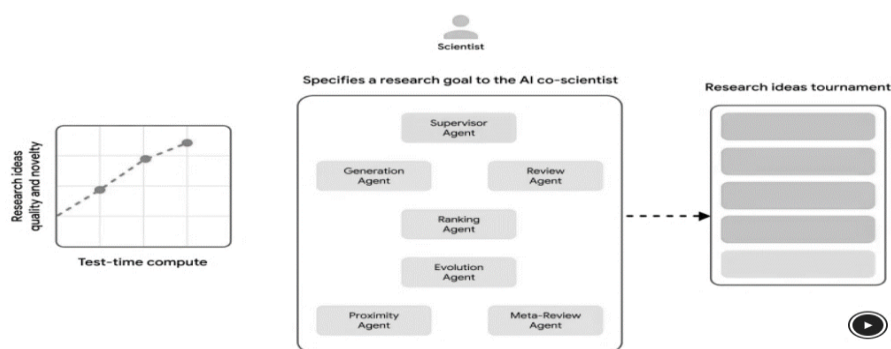


Рисунок 1.3 - Огляд співнауковця зі штучного інтелекту

Детальний опис складових архітектури системи CO-SCIENTIST показано на рис.1.4.

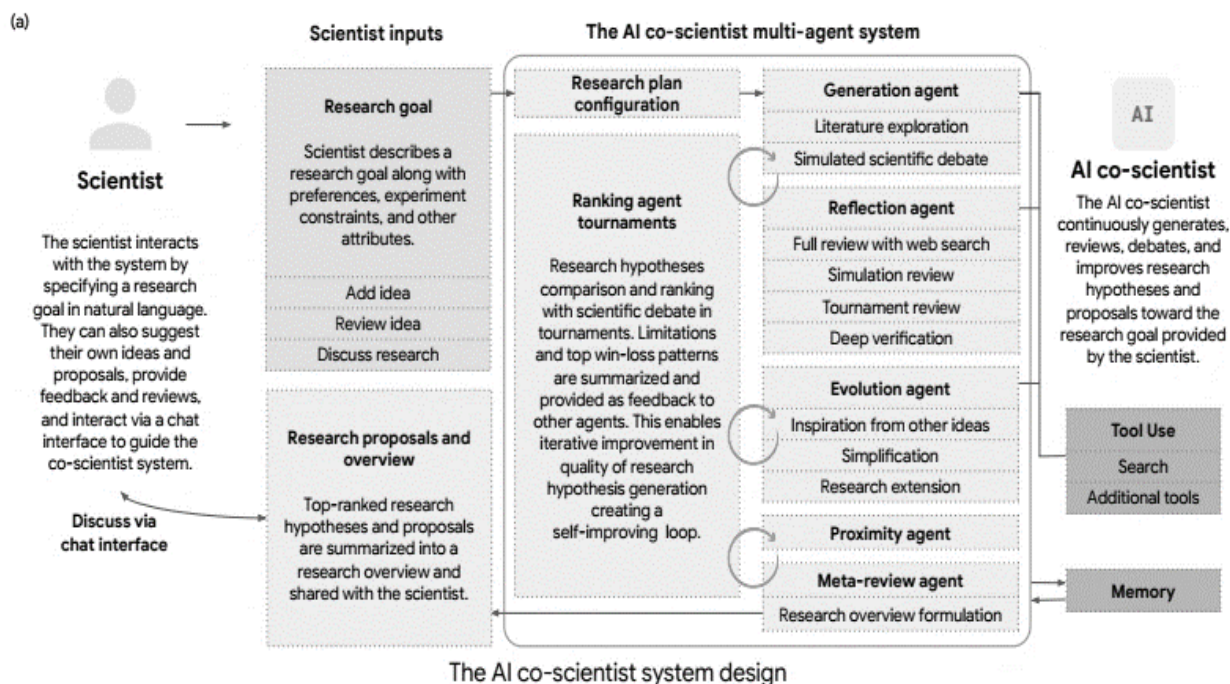


Рисунок 1.4 - Архітектура компонентів багатоагентної системи зі штучним інтелектом та співнауковцем і схема взаємодії між системою та науковцем

Співвчений зі штучним інтелектом – це багатоагентна система штучного інтелекту, призначена для функціонування як інструмент для співпраці вчених. Побудована на базі мовної моделі Gemini 2.0 від Google розроблена для відображення процесу міркування, що лежить в основі наукового методу. Окрім стандартних інструментів огляду літератури, узагальнення та «глибокого дослідження», система співвчених зі штучним інтелектом призначена для виявлення нових, оригінальних знань та формулювання очевидно нових дослідницьких гіпотез і пропозицій, спираючись на попередні дані та адаптованих до конкретних дослідницьких цілей.

Створена людиною спеціально для наукової співпраці, система co-scientist може взаємодіяти з нею різними способами, зокрема, безпосередньо надаючи власні початкові ідеї для дослідження або надаючи відгуки про згенеровані результати природною мовою. Співнауковець зі штучного інтелекту також використовує такі інструменти, як веб-пошук та спеціалізовані моделі штучного інтелекту, для покращення обґрунтованості та якості згенерованих гіпотез.

Співнауковець зі штучним інтелектом аналізує призначену мету та створює

конфігурацію плану дослідження, якою керує агент-супервізор. Агент-супервізор призначає спеціалізованих агентів до черги робітників та розподіляє ресурси. Така конструкція дозволяє системі гнучко масштабувати обчислення та ітеративно вдосконалювати свої наукові міркування для досягнення заданої мети дослідження.

На рис. 1.5 представлена схема взаємодії компонентів системи під час виконання досліджень.

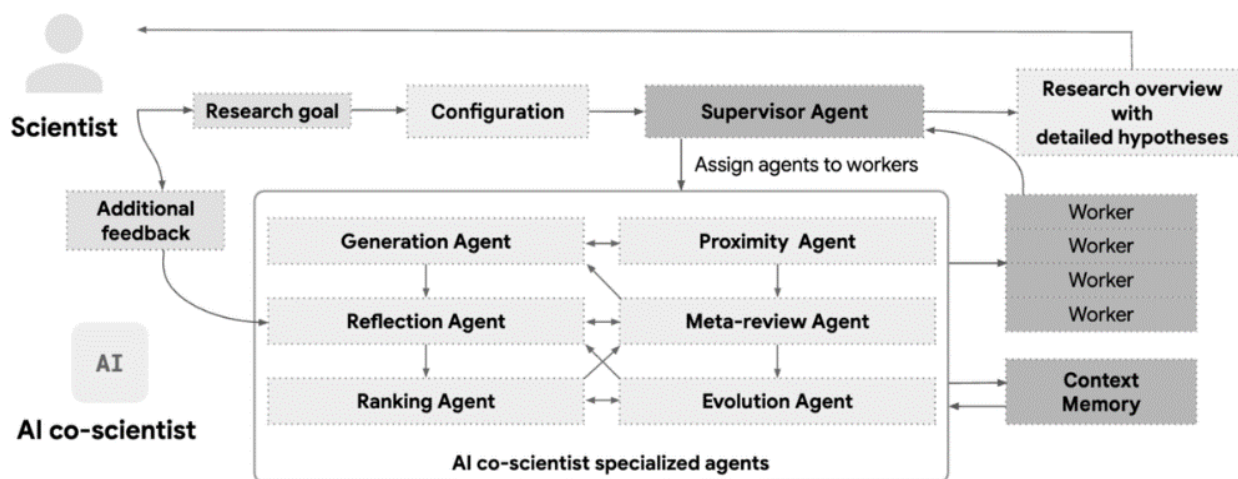


Рисунок 1.5 - Схема взаємодії компонентів системи під час виконання досліджень

Співнауковець зі штучним інтелектом використовує масштабування обчислень під час тестування для ітеративного міркування, розвитку та покращення результатів. Ключові кроки міркування включають наукові дебати на основі самостійної гри для генерації нових гіпотез, рейтингові турніри для порівняння гіпотез та процес «еволюції» для покращення якості. Агентна природа системи сприяє рекурсивній самокритиці, включаючи використання інструментів для зворотного зв'язку з метою уточнення гіпотез та пропозицій.

На разі представлена система автоматизації наукової роботи є найбільш передовою і розвинутою. Однак вона на сьогодні є недоступною, досить складна для засвоєння, має закритий код і використовується для наукових досліджень тільки всередині команди Google Research, Тому актуальною залишається задача розробки в Україні інтелектуальної комп'ютерної системи автоматизації дослідницької роботи в університеті, розв'язанню якої присвячена дана бакалаврська кваліфікаційна робота.

1.5 Концептуальні основи створення комп'ютерної системи для автоматизації етапів наукової роботи в університеті

Для автоматизації окремих етапів навчального процесу в університеті пропонується модульна структура інтелектуальної комп'ютерної системи на основі ІІІ, що включає наступні компоненти:

- Студентський портал:
- Веб-інтерфейс для студентів, що забезпечує доступ до навчальних матеріалів, персоналізованих рекомендацій, можливості подання завдань, отримання зворотного зв'язку, взаємодії з віртуальними асистентами та відстеження власного прогресу.
- Панель керування для викладачів:
- Інтерфейс для викладачів, що дозволяє керувати курсами, створювати та оцінювати завдання, отримувати доступ до аналітики успішності студентів, спілкуватися зі студентами та налаштовувати інструменти ІІІ.
- ІІІ-двигун:

Основний компонент системи, що слугує її ядром, міститиме обрані моделі та сервіси штучного інтелекту. Функцію цього двигуна виконуватиме інтелектуальна комп'ютерна система управління навчально-дослідницькою роботою, що є предметом розробки у цій бакалаврській роботі.

Структура цього компонента передбачає такі модулі:

Модуль персоналізації - призначений для впровадження систем рекомендацій, що забезпечують адресну доставку навчальних матеріалів.

Модуль оцінювання - інтегрує передові моделі обробки природної мови (NLP) та машинного навчання (ML) для автоматизації процесів оцінювання та надання ефективного зворотного зв'язку.

Модуль підтримки та тьюторства, який керує інтелектуальними чат-ботами та тьюторськими системами.

Модуль прогнозування, що запускає ML-моделі для визначення студентів, які перебувають в зоні ризику.

- Модуль комп'ютерного зору призначений для обробки даних з камер для

моніторингу освітнього середовища і аналізу залученості.

- Система управління даними:

Забезпечує безпечне зберігання та управління даними студентів, навчальними матеріалами, даними оцінювання та системними журналами, дотримуючись вимог щодо конфіденційності [33].

- Підсистема автоматизації етапів науково-дослідницької роботи з використанням моделей і інструментів ШІ, визначених в Розділі 1.3.

- Інтеграційний рівень:

Забезпечує безперебійну інтеграцію з існуючими університетськими системами, такими як інформаційна система студентів (SIS), система управління навчанням (LMS) та інші академічні платформи.

1.6 Висновки до розділу

Аналіз об'єкта автоматизації засвідчив значний потенціал використання штучного інтелекту для модернізації освітнього процесу у вищих навчальних закладах. Запропонована концепція системи керування навчальним процесом із застосуванням ШІ дозволяє автоматизувати основні етапи навчання, впроваджувати індивідуальний підхід до здобувачів освіти, підвищувати якість оцінювання та забезпечувати комплексну підтримку як студентам, так і викладачам. Крім того, система сприяє автоматизації рутинних завдань під час науково-дослідної діяльності.

Для ефективного впровадження цієї системи в університетське середовище пропонується виконати такі ключові кроки:

- Пілотний запуск - започаткування тестової програми, щоб оцінити ефективність системи на обмеженій кількості курсів і користувачів.
- Створення робочої групи - сформування цільової групи з питань ШІ в освіті, залучення викладачів, адміністративний персонал, ІТ-спеціалістів та студентів, для координації розробки й впровадження.
- Забезпечення ресурсами - виділення необхідних коштів для оновлення інфраструктури, закупівлі програмного забезпечення та підвищення кваліфікації

персоналу.

- Система управління даними - розробка всеосяжної системи для відповідального використання даних та захисту конфіденційності.
- Культура інновацій – стимулювання середовища для експериментів та безперервного вдосконалення, регулярно аналізуючи вплив ІІІ на навчальні досягнення студентів та навантаження викладачів.

Надалі важливо зосередитися на дослідженнях, що вивчають довгостроковий вплив ІІІ на залученість студентів та розвиток їхнього критичного мислення. Також пріоритетом має бути розробка просунутих моделей ІІІ для персоналізованого навчання та формування етичних норм, специфічних для застосування ІІІ в університетській освіті.

Ця частина бакалаврської кваліфікаційної роботи зосередиться на розробці ключового компонента університетської системи управління освітнім процесом - інтелектуальної комп'ютерної системи для автоматизації етапів дослідницької роботи.

2 ВИБІР ТА ІНТЕГРАЦІЯ ІНСТРУМЕНТІВ ІІІ В СИСТЕМУ АВТОМАТИЗАЦІЇ НАУКОВОЇ РОБОТИ

2.1 Вибір ефективного інструментарію ІІІ для проведення наукових досліджень

Як показав проведений в Розділі 1.3 даної частини комплексної роботи огляд інструментів і моделей ІІІ, призначених для автоматизації виконання окремих етапів дослідницької роботи, їх на сьогодні існує велика кількість. Наприклад, тільки сервісів для пошуку джерел інформації і написання наукових робіт, знайдених в Інтернеті і представлених в таблицях 1.2 і 1.3, нараховується близько 45. В зв'язку цим під час вибору інструментів ІІІ для їх використання в розроблюваній комп'ютерній системі автоматизації етапів дослідницької роботи виникає проблема вибору серед них таких, які роблять систему ефективною з точки зору затрат і продуктивності.

Під час вирішення цієї проблеми потрібно враховувати, що серед існуючих моделей і сервісів ІІІ є інструменти, призначені для реалізації різного обсягу дослідницької роботи. Є сервіси, які виконують окремі кроки дослідницької роботи (наприклад, пошук джерел, аналіз джерел на відповідність темі дослідження і резюмування наукових робіт, аналіз даних, генерація ідей, графічне оформлення статей, графічне оформлення результатів, написання статей тощо). Більшість таких сервісів міститься на безкоштовній платформі Hugging Face, і для їх повноцінного використання користувачу достатньо підписатися на них під час реєстрації на даній платформі і завантажити для локального використання в локальну платформу LM Studio, завантажену попередньо на комп'ютер користувача. Є також великі мовні моделі (LLM – Large Language Models), такі, наприклад, як Deep Research від Google Gemini, Chat GPT від Open AI або Future House, здатні проводити комплексне дослідження – здійснюють пошук релевантних до теми дослідження джерел, проводять міркування, приймають рішення на досягнення поставленої мети і оформлюють результат. Деякі з них можуть надавати програмний код для реалізації прийнятого рішення. Однак деякі етапи з оформлення результатів дослідження, наприклад, візуалізацію даних, формування графічних схем чи

презентацій вони спроможні реалізувати в обмеженому вигляді – або у вигляді текстового опису, або у вигляді незручних для демонстрації графічних діаграм. Більшість з моделей даного типу можна використовувати через веб-інтерфейси або через хмарні сервіси. Це також вимагає окремого схемного рішення з огляду на прийняту концепцію побудови розроблюваної системи управління навчальною і дослідницькою роботою на локальних платформах.

З огляду на наведені міркування під час вибору конкретних сервісів і моделей ІІІ для їх роботи в розроблюваній інтелектуальній комп'ютерній системі управління навчальним процесом потрібно орієнтуватися, на якому кроці дослідження і який інструмент потрібно використовувати. Для отримання рішення в даному напрямку потрібно виконати такі кроки:

I - рейтингування інструментів на основі вибраних критеріїв ефективності;

II - установлення послідовності їх використання у відповідності до послідовності виконання етапів дослідницької роботи.

Для виконання першого кроку виберемо такі критерії, що забезпечать мінімізацію затрат і максимізують продуктивність системи:

- 1) Наявність україномовного інтерфейсу;
- 2) Відповідність принципу Парето 80/20 – 20 відсотків сервісів ІІІ можуть забезпечити 80 відсотків потреб дослідника;
- 3) Висота рівня інтелекту відповідно до таксономії Блума;
- 4) Можливість виконати комплексне дослідження чи тільки окремі його етапи;
- 5) Локальне використання з Hugging Face чи через веб-інтерфейси;
- 6) Перевага безкоштовному перед платним.

Для виконання першого і другого кроків було проведено аналіз низки літературних робіт [2,3,4,5,10,18,19], автори яких аналізували відповідність існуючих сервісів ІІІ вказаним критеріям.

Процес вибору керувався конкретними критеріями ефективності та був спрямований на побудову гнучкої до перебудови ефективної локальної системи ІІІ, здатної задовольнити більшість потреб дослідника, підтримуючи при цьому

високі рівні інтелектуальної діяльності. Основними компонентами, визначеними як фундаментальні для цієї системи, є екосистема Hugging Face, LM Studio, Ollama та Open WebUI. Ці компоненти спільно забезпечують потужну та гнучку основу для автоматизації значної частини академічних та дослідницьких завдань викладача, з акцентом на локальному розгортанні та економічній ефективності.

2.1.1. Відповідність інструментів ШІ критеріям ефективності

Обрані інструменти ШІ були оцінені за ключовими критеріями ефективності, що є важливими для їх практичного застосування в академічному середовищі.

Відповідність принципу Парето 80/20: Цей принцип передбачає, що 20% сервісів ШІ мають задовольняти 80% потреб дослідника.

Екосистема Hugging Face слугує фундаментальним джерелом інтелекту, надаючи доступ до тисяч попередньо навчених великих мовних моделей (LMM), які охоплюють широкий спектр завдань, задовольняючи більшість основних можливостей ШІ, необхідних досліднику.

LM Studio є основним локальним середовищем виконання для LMM з Hugging Face. Він забезпечує конфіденційність даних та відсутність постійних витрат на API, що є критично важливим для 80% потреб обробки даних, пов'язаних з безпекою та економією.

Ollama — це фреймворк для оркестрації, який перетворює окремі можливості LMM на комплексні робочі процеси за рахунок можливості інтеграції веб-інтерфейсу Open WebUI з платформою LM Studio+Hugging Face. Він автоматизує 80% складних завдань, таких як аналіз літератури, планування досліджень та генерація звітів, шляхом інтеграції LMM з іншими інструментами та базами даних (RAG).

Open WebUI, хоча сам по собі не є моделлю ШІ, є критично важливим для забезпечення 80% зручності використання системи, пропонуючи інтуїтивно зрозумілий, самостійно розміщений інтерфейс для взаємодії з локальними LMM, роблячи систему доступною навіть для викладачів, які не є експертами в командному рядку.

Висота рівня інтелекту відповідно до таксономії Блума:

Екосистема Hugging Face підтримує всі рівні таксономії Блума. Від «Розуміння» (через узагальнення та відповіді на питання) до «Аналізу», «Оцінки» та «Створення» (через тонке налаштування моделей, генерацію нового контенту та допомогу у формулюванні гіпотез та висновків). Можливість тонкого налаштування дозволяє адаптувати моделі до академічних особливостей та специфічних дослідницьких доменів, підвищуючи їх інтелектуальний рівень у конкретному контексті.

Рівень інтелекту LM Studio залежить від завантаженої LMM. Проте, він дозволяє запускати моделі, здатні до «Аналізу» (наприклад, узагальнення документів), «Застосування» (відповіді на питання) та «Створення» (генерація тексту). Його функція «чат з документами» безпосередньо підтримує «Аналіз» та «Розуміння».

Ollama значно підвищує рівень інтелекту системи, дозволяючи реалізацію «агентних систем» через веб-інтерфейс, де LMM можуть приймати рішення, формулювати плани та виконувати завдання на основі конкретних цілей. Це безпосередньо підтримує вищі рівні таксономії Блума: «Аналіз», «Оцінка» та «Створення» через складні міркування та використання інструментів.

Open WebUI сам по собі не має інтелекту, але він фасилітує взаємодію, дозволяючи викладачам використовувати ШІ для завдань, що вимагають «Розуміння», «Застосування» та «Створення». Він покращує «Застосування» ШІ-інструментів.

Можливість виконати комплексне дослідження чи тільки окремі його етапи:

Екосистема Hugging Face надає інструменти та моделі для виконання окремих етапів (наприклад, вивчення джерел, написання наукової роботи, аналіз даних) і є основою для комплексного дослідження при інтеграції з іншими компонентами через свої бібліотеки Transformers та Datasets.

LM Studio сам по собі виконує окремі етапи (виведення LMM, чат з документами). Проте, його локальний API, сумісний з OpenAI, дозволяє інтегрувати його в ширші програмні рішення (наприклад, з Ollama), що робить

його частиною комплексного дослідження.

Ollama розроблена для підключення комплексних додатків, які автоматизують значні частини дослідницького процесу, від збору інформації до генерації звітів, координуючи виконання кількох етапів дослідження.

Open WebUI є інтерфейсом для всієї системи, що дозволяє викладачам керувати та взаємодіяти з ШІ на всіх етапах дослідження, де потрібна взаємодія з користувачем.

Локальне використання з Hugging Face чи через веб-інтерфейси:

Hugging Face Model Hub є веб-платформою, але її ключові бібліотеки (Transformers, Datasets) призначені для локального використання та тонкого налаштування на власному обладнанні, ідеально відповідаючи вимозі локального розгортання.

LM Studio є настільним додатком, призначеним для локального запуску LMM, повністю відповідаючи вимозі локального використання.

Ollama є фреймворком, що дозволяє локально інтегрувати LMM на LM Studio.

Open WebUI є самостійно розміщеним веб-інтерфейсом, який працює повністю офлайн, ідеально відповідаючи вимозі локального використання.

Перевага безкоштовному перед платним:

Екосистема Hugging Face переважно пропонує моделі та бібліотеки з відкритим вихідним кодом та безкоштовні для використання, що робить її надзвичайно економічно ефективним рішенням для академічних установ.

LM Studio пропонує безкоштовну версію зі значною функціональністю, ідеальну для індивідуальних розробників та невеликих проєктів, що підходить для академічного використання.

Ollama є відкритим вихідним кодом та безкоштовним для використання.

Open WebUI є відкритим вихідним кодом та безкоштовним.

Наявність україномовного інтерфейсу: У наданих джерелах не міститься інформація щодо наявності україномовного інтерфейсу для Hugging Face, LM Studio, Ollama або Open WebUI.

2.1.2. Автоматизація етапів наукового дослідження за допомогою обраних інструментів III

Інтегрована система, що включає Hugging Face, LM Studio, Ollama та Open WebUI, значно сприяє автоматизації різних етапів наукового дослідження за рахунок можливостей самих платформ і спеціалізованих моделей і сервісів, які можна до них підключити з допомогою обраних інтерфейсів..

Огляд літератури та теоретичний аналіз: Моделі Hugging Face можуть бути використані для вивчення джерел та узагальнення. Функція "чат з документами" LM Studio безпосередньо підтримує розуміння та аналіз документів. Ollama, завдяки своїм можливостям RAG та оркестрації, дозволяє проводити глибокий аналіз літератури та інтеграцію з базами даних.

Формулювання робочої гіпотези: Моделі Hugging Face можуть допомагати у формулюванні гіпотез та висновків. Агентні системи Ollama можуть здійснювати складні міркування для підтримки цього.

Планування та розробка методології дослідження: Ollama є інструментальним у автоматизації планування досліджень.

Обробка та аналіз результатів: Хоча конкретні інструменти для якісного аналізу даних (NVivo) або створення моделей машинного навчання (Google AutoML) перераховані як допоміжні¹⁶, здатність Ollama інтегрувати LMM з іншими інструментами дозволяє здійснювати складну обробку та інтерпретацію даних⁶. LM Studio підтримує аналіз через узагальнення документів.

Формулювання висновків та рекомендацій: Моделі Hugging Face можуть допомогти у формулюванні висновків. Розширені можливості міркування Ollama можуть сприяти цьому етапу.

Оформлення результатів дослідження: Моделі Hugging Face здатні до генерації тексту, що є важливим для написання наукових робіт. Ollama може автоматизувати генерацію звітів.

Загальна автоматизація: Вся система призначена для автоматизації значної частини наукової роботи викладача. Open WebUI надає зручний інтерфейс для управління та взаємодії з III на всіх етапах дослідження, що вимагають взаємодії з

користувачем.

Підсумовуючи, чотири обрані основні компоненти (Hugging Face, LM Studio, Ollama, Open WebUI) формують надійну та гнучку основу для проєкту, дозволяючи автоматизувати значну частину навчальної та дослідницької роботи викладача університету з сильним акцентом на локальне використання та економічну ефективність.

Також було проаналізовано конкретні спеціалізовані сервіси на відповідність вказаним критеріям було обрано ряд спеціалізованих сервісів, які можуть бути інтегровані на дані платформи. Як результат в таблиці 2.1 представлені сервіси і моделі III відповідно до їх рейтингу, які найкраще підходять для автоматизації дослідження.

Таблиця 2.1 - Рейтинг топ-7 AI-сервісів для наукового дослідження

№	Сервіс / Модель	Критерії (1–6)	Обґрунтування ефективності
1	ChatGPT 4o (chat.openai.com)	+	Найкраще покриває всі етапи дослідження. Працює у браузері, доступна безкоштовна версія, підтримує аналітику, реферування, гіпотези, письмове оформлення.
2	Perplexity AI	+	Інформаційний пошук, формування проблеми, гіпотези. Дає джерела. Відмінний інструмент 80/20.
3	LM Studio + Hugging Face (локально)	+	Повністю безкоштовне локальне середовище, з моделями типу Mistral, LLaMA, Phi, Gemma. Підходить для генерації текстів, резюме, завдань, висновків.
4	Elicit (elicit.org)	+	Формулювання гіпотез, аналіз джерел, обґрунтування актуальності. Не всі функції доступні безплатно.
5	SciSpace Copilot	+	Зручний для аналізу PDF статей, формулювання висновків. Безкоштовна версія обмежена.
6	Hugging Face Spaces (наприклад, summarizers, translators, Q&A)	+	Багато моделей на одній платформі. Працюють як окремі утиліти або через API.
7	Google Bard / Gemini	+	Альтернатива ChatGPT. Відмінно справляється з ідеями, планами, текстами, але гірша точність

В таблиці 2.2 наведено відповідність застосування обраних інструментів III до заданих завдань дослідницької роботи

Таблиця 2.2 - Відповідність сервісів етапам наукового дослідження

Етап дослідження	Найкращі інструменти для автоматизації
1. Визначення проблеми і теми	Perplexity, ChatGPT, Elicit
2. Теоретичний огляд, гіпотеза	ChatGPT, Elicit, SciSpace
3. Планування та методологія	ChatGPT, LM Studio, Hugging Face
4. Збір даних	ChatGPT (анкети), LM Studio, Q&A-боти
5. Аналіз і обробка даних	LM Studio, Hugging Face, Google Sheets (AI)
6. Висновки та рекомендації	ChatGPT, Gemini, LM Studio
7. Написання тексту	ChatGPT, LM Studio
8. Апробація, публікації	ChatGPT, Perplexity, SciSpace

На рис. 2.1 зображено рекомендовану траєкторію використання сервісів за етапами дослідження, що ґрунтується на висновках проведеного аналізу.

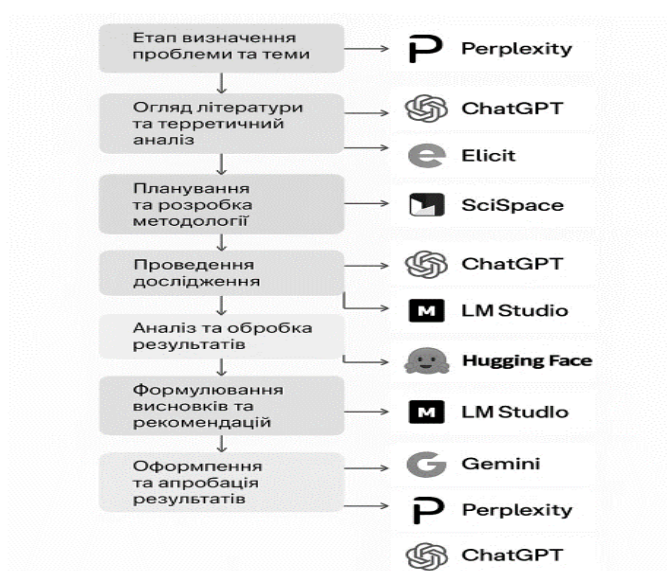


Рисунок 2.1 – Послідовність використання сервісів і моделей ШІ під час виконання етапів навчальної роботи

2.2 Обґрунтування вибору програмних платформ для інтеграції сервісів ШІ в систему автоматизації

Інтеграція сервісів штучного інтелекту (ШІ) в комп'ютерні системи, зокрема в розроблювану в даній Частині 2 комплексної бакалаврської роботи автоматизовану систему для науково-дослідницької діяльності викладачів і студентів, може бути реалізована різними способами. Двома основними підходами є використання хмарних сервісів та розгортання моделей ШІ на локальних платформах.

2.2.1 Інтеграція через хмарні Сервіси та API

Більшість провідних постачальників ІІІ-сервісів, таких як OpenAI (GPT-4), Google Gemini, Microsoft Copilot, Anthropic Claude, пропонують доступ до своїх моделей через API (інтерфейси прикладного програмування - ППІ) або веб-інтерфейси. В цьому випадку обчислення виконуються на серверах постачальника послуг, що розміщені у хмарі. Інтеграція зазвичай здійснюється шляхом реєстрації акаунтів на відповідних сервісах та використання їхніх API, або через готові інтеграції/боти для платформ типу Moodle, Google Classroom, Microsoft Teams.

Розглянемо основні переваги хмарних сервісів. До них належать:

- Легкість використання дозволяє витратити мінімум часу для виконання процесів реєстрації та підключення.
- локальний комп'ютер не потребує високої обчислювальної потужності, оскільки обчислення виконуються у хмарі,.
- автоматичні оновлення та підтримка відбуваються на стороні постачальника, при цьому всі покращення та оновлення моделей відбуваються автоматично.
- легка інтеграція до освітніх платформ, таких як Moodle, Google Classroom, Microsoft Teams, оскільки сервіси часто мають готові рішення для інтеграції..

До недоліків хмарних сервісів потібно віднести

- відсутність конфіденційності, оскільки дані передаються через інтернет, що може становити проблему для закритих освітніх установ, конфіденційної наукової роботи та досліджень.
- Обмеження безкоштовного використання, оскільки безкоштовно доступні лише менш потужні моделі (наприклад, OpenAI надає GPT-3.5 безкоштовно, тоді як GPT-4 є платним). Використання потужніших моделей або більших обсягів обчислень вимагає оплати.
- залежність від сервісу, тому що існує ризик припинення роботи сервісу або зміни політики доступу до нього.

Хмарні сервіси рекомендується використовувати для інтеграції ІІІ в системи автоматизації науково-дослідницької діяльності, коли відсутнє потужне локальне обладнання або коли є потреба у швидкому розгортанні рішення без складних

налаштувань. Зважаючи на відсутність конфіденційності інформації під час використання веб-сервісів в наукових дослідженнях, що є критичним для сучасної ситуації в Україні, розроблювану в бакалаврській роботі систему автоматизації етапів дослідницької діяльності вирішено будувати на основі локальних платформ.

2.2.2 Вибір платформи з репозиторієм моделей і сервісів ШІ

На поточному етапі розробки інтелектуальної комп'ютерної системи для автоматизації етапів навчального процесу в Розділі 1 даної роботи в якості оптимального вибрано варіант використання тільки локальних моделей ШІ через локальну платформу Hugging Face для максимальної конфіденційності [31]. Інтеграція Hugging Face в підсистему автоматизації науково-дослідницької роботи в університеті може значно прискорити та спростити процес досліджень для викладачів і студентів. Вона надає доступ до передових ШІ-сервісів, сприяє ефективному використанню ресурсів, стимулює спільну роботу та дозволяє зосередитися на креативних аспектах наукової роботи, мінімізуючи рутинні та технічні завдання.

Hugging Face є найбільшим репозиторієм попередньо навчених моделей машинного навчання (ML) і глибокого навчання (DL) для обробки природної мови (NLP), комп'ютерного зору, генерації зображень та інших завдань. Це дозволяє викладачам і студентам швидко знаходити та використовувати моделі, які вже були навчені на величезних обсягах даних, замість того, щоб навчати їх з нуля.

Програмна платформа Hugging Face включає три ключові компоненти:

- 1) Inference API,
- 2) Transformers,
- 3) Spaces.

Inference API дає змогу використовувати моделі без потреби в локальному розгортанні.

Модуль Transformers забезпечує доступ до широкого спектра відкритих моделей, таких як GPT, BERT, LLaMA, Whisper та інші.

Spaces дозволяє створювати та запускати AI-додатки для освітніх і науково-

дослідницьких цілей. що робить платформу зручною для викладачів і для студентів, які хочуть проводити експерименти, будувати прототипи систем аналізу даних або обробки текстів, розробляти власні наукові інструменти.

Використання Hugging Face у розроблюваній системі дозволить значно підсилити науково-дослідницьку діяльність викладачів та студентів університету за допомогою своїх сервісів штучного інтелекту. Реалізація цих можливостей здійснюється за рахунок таких переваги її використання:

1) Оптимізація аналізу наукових текстів - Платформа надає доступ до моделей NLP, які можуть автоматично аналізувати, узагальнювати та класифікувати наукові статті, прискорюючи обробку великого масиву текстових даних.

2) Автоматичне реферування та огляди літератури - Hugging Face дозволяє створювати автоматизовані огляди літератури, що допомагає швидко отримувати стислі виклади основних ідей з великого обсягу матеріалів.

3) Допомога у написанні наукових статей - використання мовних моделей для перевірки стилю, граматики, чіткості та структури тексту значно спрощує підготовку публікацій для наукових журналів.

4) Інтелектуальний пошук і систематизація даних - моделі Hugging Face можуть допомогти шукати релевантні дослідження, наукові роботи, патенти або дані, зводячи до мінімуму ручну працю.

5) Автоматизація перекладу та адаптації текстів - ШІ-платформа підтримує багатомовні моделі, що дозволяє швидко перекладати дослідницькі матеріали та адаптувати їх до міжнародних вимог.

6) Аналіз та прогнозування результатів досліджень - Hugging Face має моделі для машинного навчання, які можуть аналізувати експериментальні дані, знаходити закономірності та передбачати результати на основі трендів.

7) Розпізнавання та аналіз зображень і графіків - вбудовані моделі для роботи з комп'ютерним зором можуть розпізнавати та аналізувати графіки, схеми та візуальні матеріали, що є корисним для технічних наук.

8) Колаборація та інтеграція - Hugging Face підтримує інтеграцію з

іншими інструментами наукових досліджень, такими як Jupyter Notebook, Google Scholar та внутрішні репозиторії університету.

9) Автоматизовані рекомендації для наукових проектів - системи штучного інтелекту можуть допомагати визначати перспективні напрями досліджень на основі світових трендів і рекомендацій.

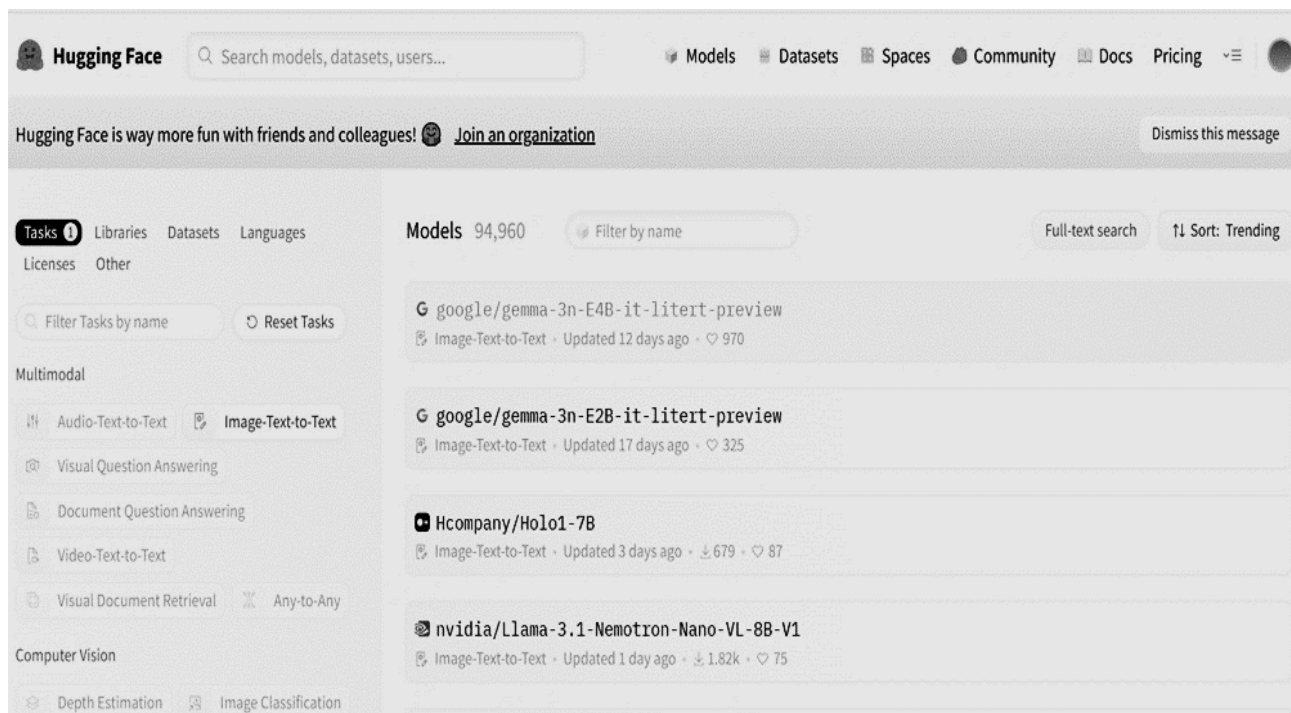


Рисунок 2.2 - Інтерфейс програмного середовища Hugging Face

До недоліків можна віднести те, що не всі моделі доступні безкоштовно у хмарному режимі, крім того, потрібно мати базові навички роботи з API. Інтерфейс програмного середовища Hugging Face представлений на рис. 2.2.

Дана платформа надає можливість реалізувати за допомогою наявних в ній безкоштовних інструментів ШІ велику кількість функцій з автоматизації окремих етапів науково-дослідницької роботи.

2.2.3 Вибір локальної платформи для запуску моделей ШІ на локальному комп'ютері

Розглянемо три популярні платформи для локального запуску мовних моделей (LLM) та сервісів ШІ. До них відносяться Ollama, Koboldcpp та LM Studio.

Платформа Ollama призначена для локального запуску та управління моделями III на персональних пристроях, що дозволяє розгортати LLM без хмарних сервісів. Вона підтримує популярні LLM, зокрема Mistral, LLaMA, Gemma, Code Llama та надає можливість створювати кастомні моделі за допомогою Modelfile.

Переваги: Забезпечує високу продуктивність та конфіденційність. Відрізняється простотою встановлення та використання завдяки інтуїтивно зрозумілому інтерфейсу командного рядка. Пропонує гнучкість та кастомізацію моделей. Використовує апаратне прискорення (GPU/CPU). Має легку інтеграцію через API. Інтеграція здійснює шляхом взаємодії через API, що дозволяє вбудувати завантажені з Hugging Face III в різні застосунки.

Основний недолік полягає в тому, що платформа офіційно підтримує macOS та Linux, а для користувачів Windows потребує встановлення через WSL (Windows Subsystem for Linux).

Платформа KoboldCpp є високопродуктивною, оптимізованою для CPU/GPU версія KoboldAI і призначена для легкого запуску локальних LLM на основі GGUF-бібліотек. Підтримує локальні LLM-моделі, зокрема LLaMA, Mistral, Gemma, використовуючи GGUF-моделі. Інтегрується в систему за допомогою локального API.

Переваги: Відрізняється легкою установкою та запуском, не потребує складного налаштування. Підтримує як Windows, так і Linux. Забезпечує швидку роботу на CPU та GPU завдяки оптимізації (використовує допрацьований llama.cpp) та підтримує CUDA, ROCm, Metal. Має вбудований локальний сервер та дружній інтерфейс у браузері. Може запускатися як локальний API.

Основним недоліком є те, що дана платформа має меншу гнучкість у виборі джерел моделей порівняно з LM Studio, оскільки сфокусований на форматі моделей GGUF, а також Вимагає самостійного оновлення та підтримки.

Платформа LM Studio призначена для запуску мовних моделей локально на комп'ютері без підключення до інтернету. Підтримує всі моделі, доступні на Hugging Face, включаючи GPT-2, LLaMA, Mistral, BERT та інші.

До переваг даної платформи відносяться високий рівень конфіденційності, безкоштовність використання, гнучкість у виборі моделей, можливість автономної роботи, може інтегруватися через API або локальний сервер у такі локальні системи управління навчальним процесом, як Moodle, Google Classroom, JetIq, Microsoft Teams, Notion

До недоліків можна віднести трохи складніше налаштування, ніж у Koboldcpp та необхідність базових процесів розгортання моделей. Потребує самостійного оновлення та підтримки.

Описані характеристики розглянутих платформ для інтеграції сервісів ШІ в локальній комп'ютерній системі управління навчальним процесом зведені в порівняльну таблицю 2.3.

Таблиця 2.3 – Порівняльна таблиця платформ для запуску сервісів ШІ

Властивість / Платформа	LM Studio	Ollama	KoboldCpp
Локальний запуск	Так	Так	Так
Без інтернету	Так	Так	Так
Конфіденційність	Висока	Висока	Висока
Вимоги до обладнання	Бюджетний комп'ютер для моделей ШІ середньої складності	Потребує апаратне прискорення (GPU/CPU)	Оптимізовано для CPU/GPU
Складність налаштув.	Проста установка	Складніша, через WSL	Низька (не потребує складного налаштув.)
Підтримка ОС	Windows, Linux	macOS, Linux (Windows через WSL)	Windows, Linux
Підтримувані моделі	Hugging Face (всі формати)	Різні LLM (Mistral, LLaMA, Gemma), Custom (Modelfile)	GGUF-моделі (LLaMA, Mistral, Gemma)
Інтерфейс	Вбудований API	Командний рядок, API	Вбудований сервер, браузерний інтерфейс, API
Оновлення/Підтримка	Самостійно	Автоматично	Самостійно

На основі аналізу наведених в таблиці характеристик для використання в розроблюваній інтелектуальній комп'ютерній системі управління освітнім процесом була обрана платформа LM Studio.

2.3 Розробка структурної схеми підсистеми автоматизації наукової роботи

Відповідно до описаної в попередніх розділах концепції інтеграції сервісів ІІІ в розроблювану інтелектуальну комп'ютерну систему управління освітнім процесом, структурна схема підсистеми автоматизації наукової роботи викладача побудована у вигляді, представленому на рис. 2.3. В доповнення до концептуальної схеми ІКСУ, представленої в Частині 1 даної комплексної бакалаврської роботи, в систему введено веб-інтерфейс Open WebUI, який забезпечує локальну роботу з моделями ІІІ як з локальної платформи LM Studio, так і моделями з веб-ресурсів. Така схема забезпечує необхідну конфіденційність діяльності дослідника.

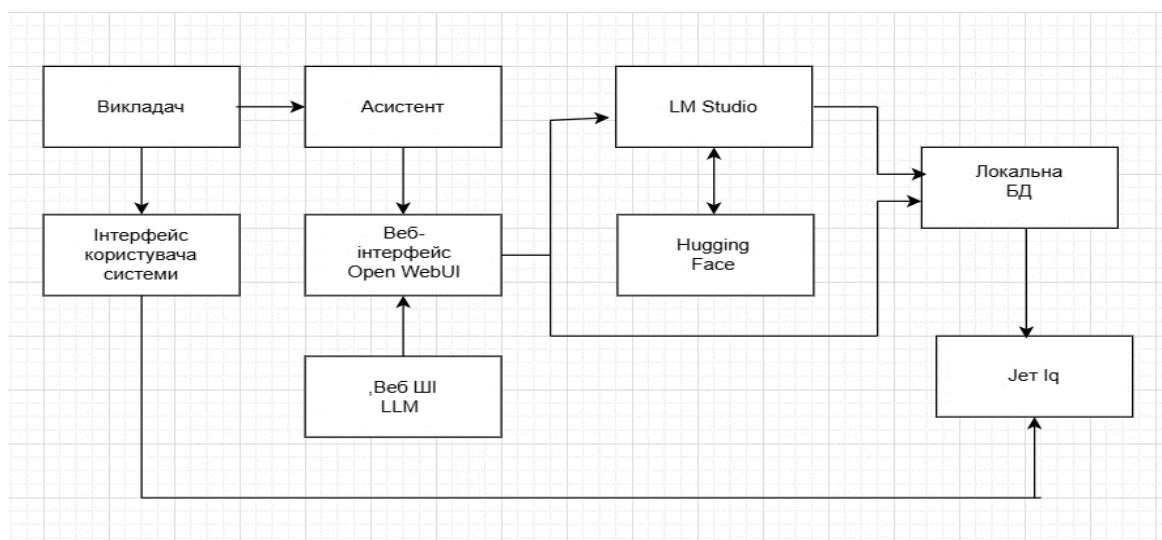


Рисунок 2.3 – Структурна схема підсистеми автоматизації наукової роботи викладача

Порядок роботи користувача в цій системі має такий вигляд. Викладач для виконання потрібного з етапів наукової роботи з допомогою асистента вводить запит з запрограмованого кейсу промптів у вікно веб-інтерфейсу Open WebUI, інтерфейс завантажує потрібну модель ІІІ з LM Studio або з веб-ресурсу, модель виконує потрібно роботу і результат завантажується в локальну базу даних на комп'ютері користувача. При необхідності отримані результати дослідження з допомогою індивідуального інтерфейсу користувача з університетською системою Jet Iq можуть бути передані в один із відповідних інформаційних модулів цієї системи, наприклад, в репозиторій.

UML- діаграма послідовності роботи викладача в описаній структурі представлена на рис. 2.4.

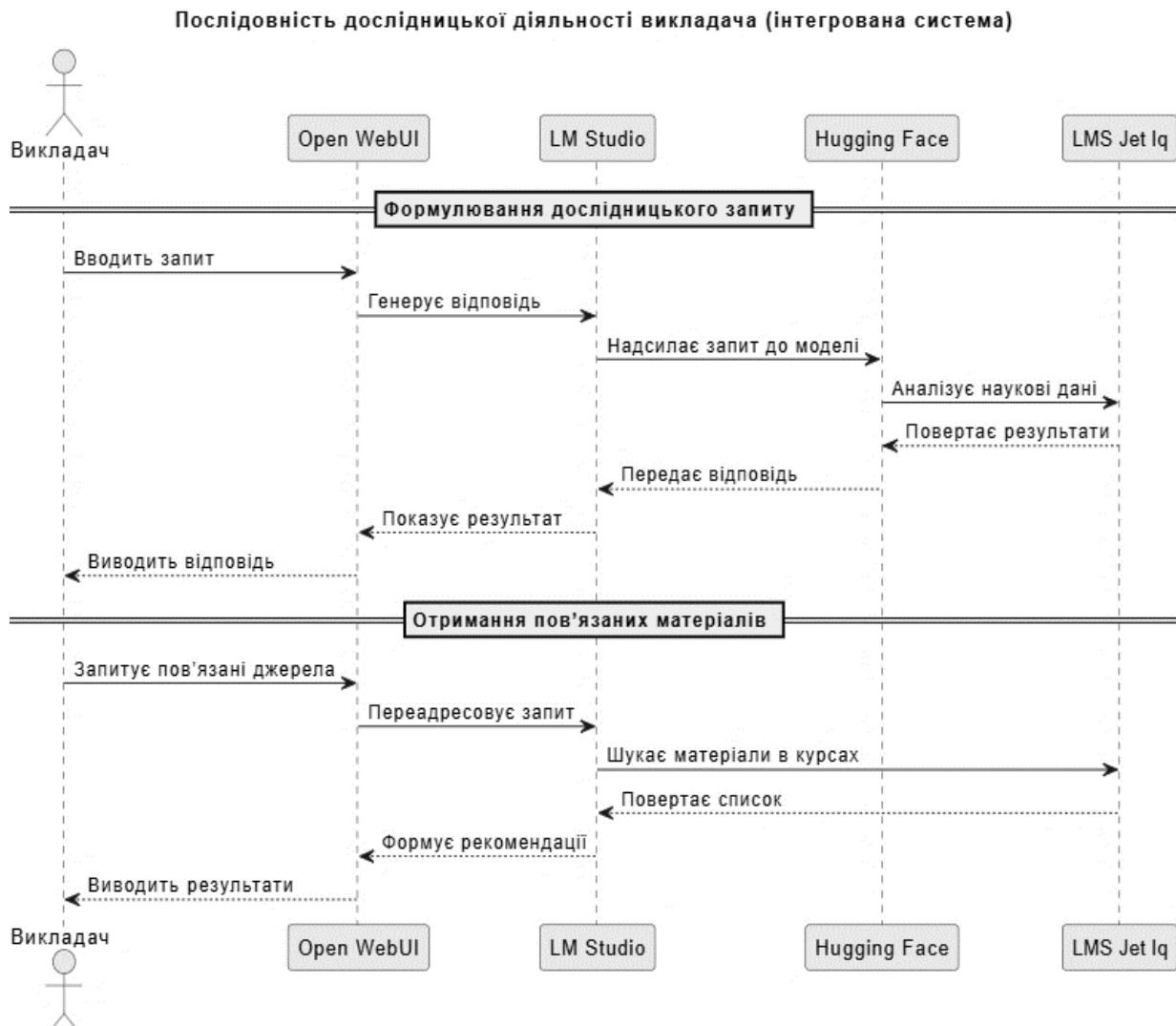


Рисунок 2.4 – UML-діаграма послідовності дослідницької роботи

Принцип інтеграції веб-інтерфейсу Open WebUI в систему. Для забезпечення можливості використання в системі автоматизації наукової роботи веб моделей III використовує інтерфейс Open WebUI- графічний інтерфейс, що інтегрує в роботу зовнішні моделі, такі як Chat GPT чи Gemini, і локальні LLM, встановлені через LM Studio або Hugging Face Transformers. Він дає змогу працювати з усіма моделями локально, що є важливим з точки зору забезпечення конфіденційності дослідницької роботи.

Open WebUI забезпечує наступні переваги:

- Локальна робота — без відправки даних в інтернет;
- Підтримка декількох моделей одночасно;
- Збереження історії запитів (сесій);
- Розширення через власні промпти та плагіни;
- Інтеграція з PDF, зображеннями, кодом.

Запуск інтерфейсу в роботу системи виконано в такій послідовності:

після встановлення сервера LM Studio з завантаженими LLM (наприклад, mistral, phi, gemma, llama2) встановлюється Open WebUI системними командами:

```
docker run -d -p 3000:3000 \
  -e OLLAMA_BASE_URL=http://host.docker.internal:11434 \
  ghcr.io/open-webui/open-webui:main
```

і здійснюється перехід у браузері на <http://localhost:3000>.

Порядок роботи з даним інтерфейсом під час наукового дослідження має такий вигляд.

На етапі 1 «Визначення проблеми та теми» асистент вводить запит, наприклад “Які актуальні проблеми досліджуються в галузі освітньої аналітики у 2024–2025 роках?”. Open WebUI згенерує перелік тем, а також допоможе конкретизувати формулювання.

На етапі 2 «Теоретичний аналіз і гіпотеза» формулюється запит: “Зроби огляд літератури на тему [тема] із висновком про теоретичні підходи”, додайте PDF-файл через вікно завантаження або вставляється уривок тексту — модель зробить узагальнення. Після цього можна сформулювати робочу гіпотезу за запитом:

“На основі аналізу сформулюй перевірну гіпотезу для дослідження”

На етапі 3 «Планування методології» наприклад по запити: “Склади план дослідження для емпіричної перевірки гіпотези про [зміст]” модель надає структуровану методику, включно з методами збору та аналізу.

На етапі 4 «Збір даних», наприклад, якщо плануються анкети, опитування або скрипти інтерв’ю вводиться промпт: “Згенеруй анкету для опитування студентів про [тема]”

На етапі 5 «Обробка результатів» асистентом вводяться запити типу: “Склади таблицю для обробки даних з опитування”, “Поясни, як провести кореляційний аналіз у Python”. Пр цьому можна завантажити текстові масиви або вставити вручну, і модель надасть класифікацію та узагальнення.

На етапі 6 «Висновки та рекомендації» запит має, наприклад, такий вигляд: “Сформулуй висновки та рекомендації за результатами дослідження у сфері...”. Також тут можна окремо згенерувати фрази для обґрунтування актуальності, новизни, цінності.

На етапі 7 «Оформлення та структура» асистент може ввести промпт “Склади структуру статті з цієї теми відповідно до вимог ВАК”, або “Структуруй текст кваліфікаційної роботи за шаблоном”.

На практиці, виходячи з тематики і досвіду дослідника, для асистента веб-інтерфейсу можна запрограмувати кейс промптів, які будуть викликатися на відповідному етапі роботи дослідника. Приклади таких промптів запрограмовані в графічному інтерфейсі, розробленому в програмній частині бакалаврської роботи. Програма розробленого графічного веб-інтерфейсу “AI Teacher GUI” представлена в Додатку В пояснювальної записки.

2.4 Висновки до розділу

В цьому розділі проведено рейтингування існуючих сервісів ШІ і їх вибір для інтеграції в комп’ютерну систему автоматизації викладацької діяльності, розроблено структуру підсистеми автоматизації з використанням локальних ресурсів на платформах LM Studio і Hugging Face і веб-інтерфейсу для викоистання хмарних сервісів. Розроблено UML-діаграми послідовності етапів науково-дослідної роботи та діаграму послідовності використання сервісів ШІ на різних етапах дослідницької роботи.

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Обґрунтування вибору програмного середовища Python

Для розробки локальної комп'ютерної системи автоматизації дослідницьких завдань на платформі LM Studio та Hugging Face, інтегрованої з веб-інтерфейсом Open WebUI, в роботі обрано програмне середовище Python 11.3. Оптимальність даного вибору продиктована такими характеристиками даного середовища, необхідними в розробці даної системи:

Гнучкість і універсальність - Python дозволяє підтримувати широкий набір бібліотек для машинного навчання, обробки природної мови NLP, автоматизації та веб-розробки, а також легко інтегрувати локальні моделі (Hugging Face, LM Studio) та створювати веб-інтерфейси (OpenWebUI, Django, Flask та інші);

Підтримка роботи з AI та ML - Hugging Face надає готові моделі для моделей обробки природної мови NLP, які легко завантажуються та керуються за допомогою Python, а бібліотеки Python, такі як PyTorch і TensorFlow, добре інтегруються з Hugging Face та LM Studio, дозволяючи оптимізувати роботу моделей штучного інтелекту.

Легка інтеграція з Open WebUI - Python підтримує інтеграцію через REST API або WebSocket, що дозволяє взаємодіяти з Open WebUi для створення динамічних інтерфейсів. Крім того, використання OpenWebUI або Flask забезпечує ефективне розгортання веб-сервісів.

Автоматизація та скриптинг - Python дозволяє автоматизувати завдання дослідження: збір даних, їх обробку та аналіз, інтеграцію моделей обробки природної мови NLP. Потужні бібліотеки (pandas, numpy, asyncio) дозволяють ефективно обробляти великі масиви даних;

Підтримка локальних рішень - Python добре адаптований для локального використання, що знижує залежність від хмарних сервісів та покращує конфіденційність, а можливість роботи з контейнерами Docker та віртуальними середовищами забезпечує гнучкість розгортання.

Окрім вже названих, Python має ще багато переваг для автоматизації досліджень. До них відносяться:

Велика кількість бібліотек для наукових обчислень - бібліотеки SciPy, NumPy, Pandas дозволяють ефективно працювати з великими масивами даних, проводити статистичний аналіз та моделювання. Такі бібліотеки як Matplotlib, Seaborn, Plotly забезпечують гнучку візуалізацію результатів досліджень.

Автоматизація обробки тексту та NLP – компоненти NLTK, spaCy, Transformers (Hugging Face) забезпечують аналіз тексту, класифікацію, витягування сутностей, машинний переклад. Ці можливості Python дозволяє легко інтегрувати у дослідницькі завдання.

Паралельні та розподілені обчислення - бібліотеки Multiprocessing, Dask, Ray допомагають ефективно використовувати багатоядерні процесори, а Apache Spark, PySpark дозволяють виконувати розподілену обробку великих даних.

Автоматизація API-запитів та веб-скрейпінг - бібліотеки Requests, BeautifulSoup, Scrapy дозволяють витягати дані з веб-ресурсів, що дозволяє Python спростувати збір інформації для подальшого аналізу.

Інтеграція з платформами для машинного навчання - бібліотеки TensorFlow, PyTorch, Hugging Face Transformers дозволяють використовувати готові AI-моделі, з якими Python легко працює як з локальними, так і з хмарними AI-рішеннями.

Гнучкість у роботі з базами даних - бібліотеки SQLite, PostgreSQL, MongoDB, SQLAlchemy підтримують ефективне зберігання та аналіз даних, завдяки чому Python добре інтегрується з системами управління базами даних.

З огляду на ці можливості, програмне середовище Python є потужним інструментом для автоматизації дослідницьких завдань, особливо у контексті інтеграції AI в освітні технології.

3.2 Опис розроблених програмних функцій

Розроблене програмне забезпечення для інтелектуальної комп'ютерної системи управління реалізує низку функцій для автоматизації робіт викладача і студента під час проходження навчального процесу. Нижче наводяться тексти програм мовами Python і Java, орієнтованих на їх використання прикладними програмними інтерфейсами платформ LM Studio і Hugging Face.

Програма автоматичного підключення до LM Studio потрібної моделі з Hugging Face

```
import streamlit as st

# Визначення списку моделей
task_options = {
    "Генерація тексту": "TheBloke/Mistral-7B-Instruct-v0.1-GGUF",
    "Транскрипція лекцій": "openai/whisper-large-v3",
    "Переклад тексту": "Helsinki-NLP/opus-mt-en-uk",
    "Аналіз тексту": "bert-base-uncased"
}

st.title("Автоматизований вибір ШІ-моделі для викладачів")

# Викладач обирає завдання
task = st.selectbox("Оберіть завдання", list(task_options.keys()))

# Відображення вибраної моделі
selected_model = task_options[task]
st.write(f"**Рекомендована модель:** {selected_model}")
```

З допомогою даного коду викладач відкриває веб-інтерфейс, обирає завдання, наприклад, здійснення генерації тексту для статті, і система автоматично пропонує оптимальну модель.

Після підключення потрібної моделі за допомогою цієї програми її можна запустити через API LM Studio

```
import requests
def run_model(model_name, prompt):
    url = "http://localhost:1234/v1/completions"
    headers = {"Content-Type": "application/json"}
    data = {
        "model": model_name,
        "prompt": prompt,
        "max_tokens": 500
    }
    response = requests.post(url, headers=headers, json=data)
    return response.json().get("choices", [{}])[0].get("text", "Помилка генерації")
from gtts import gTTS
import os
def speak_text(text):
    tts = gTTS(text=text, lang="uk")
    tts.save("response.mp3")
    os.system("start response.mp3") # Для Windows (на Linux/Mac замінити на 'afplay' або 'mpg321')
```

Розроблена програма створює HTTP-запит (<http://localhost:1234/v1/completions>) до LM Studio API, передає назва моделі, яку потрібно запустити, і отримує відповідь та повертає результат. В кінець програми вставлено код, який озвучує відповідь.

Програма інтеграції в LM Studio API простого асистента:

```

import requests
def chat_with_model(prompt):
    url = "http://localhost:1234/v1/completions"
    headers = {"Content-Type": "application/json"}
    data = {
        "model": "Mistral-7B-Instruct-v0.1-GGUF",
        "prompt": prompt,
        "max_tokens": 150
    }
    response = requests.post(url, headers=headers, json=data)
    return response.json().get("choices", [{}])[0].get("text", "Помилка")
while True:
    user_input = input("👤 Викладач: ")
    if user_input.lower() in ["вихід", "exit"]:
        break
    response = chat_with_model(user_input)
    print(f"Асистент: {response}")

```

Код об'єднання всіх компонентів в єдину програму асистента автоматизації

```

import speech_recognition as sr
import requests
from gtts import gTTS
import os
# Функція розпізнавання голосу
def recognize_speech():
    recognizer = sr.Recognizer()
    with sr.Microphone() as source:
        print("Говоріть...")
        recognizer.adjust_for_ambient_noise(source)
        audio = recognizer.listen(source)
    try:
        text = recognizer.recognize_google(audio, language="uk-UA")
        print(f"🗣️ Ви сказали: {text}")
        return text
    except sr.UnknownValueError:
        print("Не вдалося розпізнати голос")
        return None
    except sr.RequestError:
        print("⚠️ Помилка сервера розпізнавання")
        return None

# Функція вибору моделі
def select_model(task):
    models = {
        "генерація тексту": "TheBloke/Mistral-7B-Instruct-v0.1-GGUF",
        "транскрипція лекцій": "openai/whisper-large-v3",
        "перевірка завдань": "bert-base-uncased",
        "чат-бот для студентів": "TheBloke/Llama-2-13B-Chat-GGUF"
    }
    for key in models.keys():
        if key in task.lower():
            return models[key]
    return None

# Функція виклику моделі через API
def run_model(model_name, prompt):
    url = "http://localhost:1234/v1/completions"
    headers = {"Content-Type": "application/json"}
    data = {
        "model": model_name,
        "prompt": prompt,

```

```

    "max_tokens": 500
}
response = requests.post(url, headers=headers, json=data)
return response.json().get("choices", [{}])[0].get("text", "Помилка генерації")
# Функція озвучення тексту
def speak_text(text):
    tts = gTTS(text=text, lang="uk")
    tts.save("response.mp3")
    os.system("start response.mp3")
# Основна логіка роботи асистента
print(" Вітаю! Я ваш голосовий асистент для автоматизації навчального процесу.")
task = recognize_speech()
if task:
    model = select_model(task)
    if model:
        print(f"Обрана модель: {model}")
        prompt = "Створи підсумковий звіт з цього предмету."
        response = run_model(model, prompt)
        print(" Відповідь ШІ:", response)
        speak_text(response)
    else:
        print(" Не знайдено відповідну модель.")

```

Програма запуску веб-інтерфейсу Open WebUI у локальній мережі університету

```

env
# URL до локального LM Studio, який використовує Ollama-сумісний API
OLLAMA_BASE_URL=http://192.168.1.100:1234

# Hugging Face API токен (не обов'язково, якщо всі моделі локальні)
HF_TOKEN=hf_XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

# IP-адреса інтерфейсу в локальній мережі для доступу з інших ПК
WEBUI_HOST=0.0.0.0
WEBUI_PORT=3000

# (Необов'язково) Назва моделі за замовчуванням
DEFAULT_MODEL=mistral

# Увімкнути / вимкнути реєстрацію користувачів (для відкритого доступу в аудиторіях)
WEBUI_ALLOW_REGISTRATION=true
# Директорія для зберігання історій та промптів
DATA_PATH=/app/backend/data\

```

Надає можливість одночасного підключення до локального LM Studio, моделей Hugging Face та інтеграції з проксі або іншими інструментами.

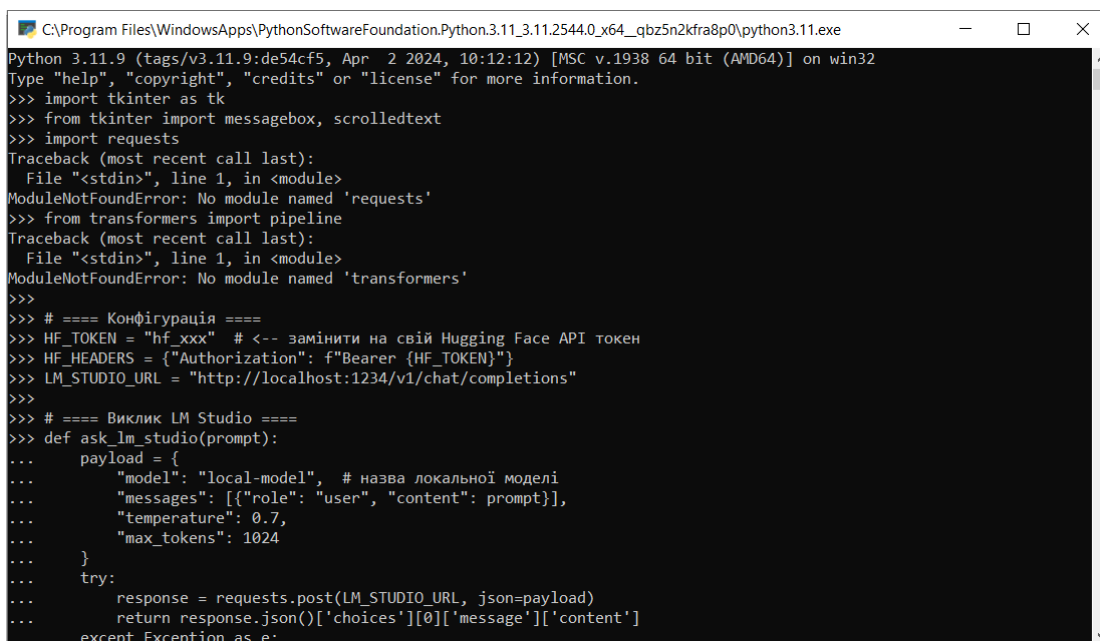
Пояснення інструкцій програми надано в таблиці 3.1.

Таблиця 3.1 – Призначення інструкцій програми

Інструкція	Призначення
OLLAMA_BASE_URL	IP-адреса сервера LM Studio у локальній мережі (уточніть IP хоста)
HF_TOKEN	Для використання зовнішніх API Hugging Face (наприклад, для перекладу)
Інструкція	Призначення
WEBUI_HOST	0.0.0.0 дозволяє доступ до інтерфейсу з інших комп'ютерів у мережі
WEBUI_PORT	Порт, на якому працюватиме Open WebUI
DEFAULT_MODEL	Ім'я локальної моделі (має збігатися з тим, що активовано в LM Studio)
WEBUI_ALLOW_REGISTRATION	Якщо true — дозволить іншим користувачам створювати акаунти

3.3 Розробка програми веб-інтерфейсу Open WebUI

Для зручності роботи користувача в системі автоматизації наукової роботи був розроблений графічний інтерфейс “ AI Teachures GUI” мовою Python. Лістинг програми наведений в Додатку В пояснювальної записки. На рис. 3.1 показано екран середовища Python 11.3 з частиною тексту програми.



```

C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.2544.0_x64__qbz5n2kfra8p0\python3.11.exe
Python 3.11.9 (tags/v3.11.9:de54cf5, Apr 2 2024, 10:12:12) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import tkinter as tk
>>> from tkinter import messagebox, scrolledtext
>>> import requests
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'requests'
>>> from transformers import pipeline
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'transformers'
>>>
>>> # ==== Конфігурація ====
>>> HF_TOKEN = "hf_xxx" # <-- замінити на свій Hugging Face API токен
>>> HF_HEADERS = {"Authorization": f"Bearer {HF_TOKEN}"}
>>> LM_STUDIO_URL = "http://localhost:1234/v1/chat/completions"
>>>
>>> # ==== Виклик LM Studio ====
>>> def ask_lm_studio(prompt):
...     payload = {
...         "model": "local-model", # назва локальної моделі
...         "messages": [{"role": "user", "content": prompt}],
...         "temperature": 0.7,
...         "max_tokens": 1024
...     }
...     try:
...         response = requests.post(LM_STUDIO_URL, json=payload)
...         return response.json()[0]['choices'][0]['message']['content']
...     except Exception as e:

```

Рисунок 3.1- Екран програмного середовища з текстом програми

В таблиці 3.2 наведені пояснення головних функцій програми.

Таблиця 3.2 – Пояснення основних функцій програми

Функція	Призначення
ask_lm_studio(prompt)	Надсилає запит до локальної LLM-моделі, запущеної через LM Studio, та повертає відповідь
ask_huggingface(model_id, prompt)	Використовує зовнішні моделі Hugging Face через API (наприклад, перекладачі чи генератори тексту).
summarize_text(text) Функції	Стискає великі тексти (лекції, статті) до короткого резюме.
handle_action(action)	Обробляє дії користувача (натискання кнопок “План курсу”, “Тестові завдання” тощо).
insert_prompt(prompt_text)	Автоматично вставляє шаблон дослідницького промпту у поле введення
tkinter	Інтерфейсні елементи для взаємодії з користувачем.

На рис. 3.2 наведено екран графічного інтерфейсу при натиснутій кнопці «Актуальність теми»..

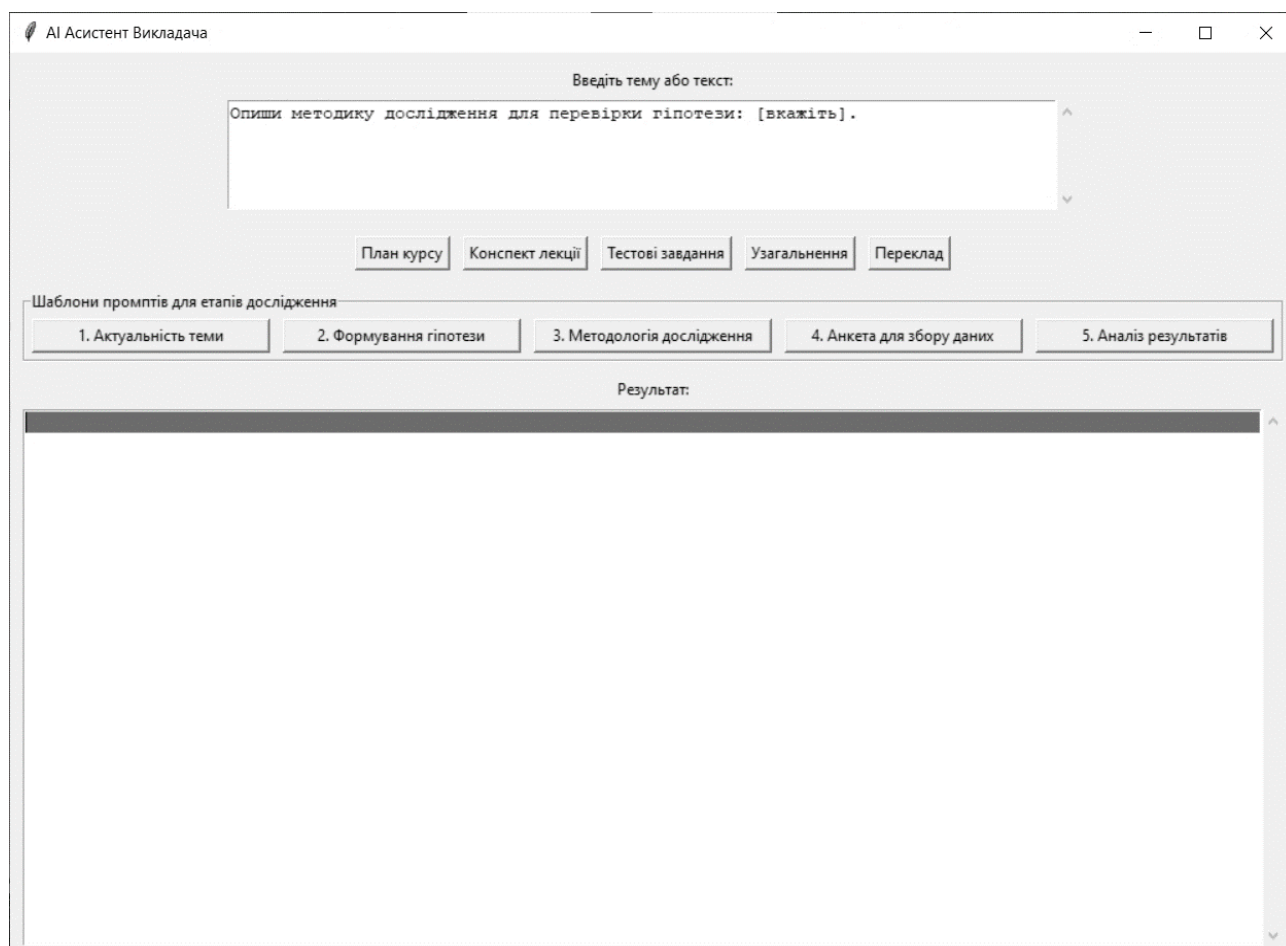


Рисунок 3.2 - Екран графічного інтерфейсу при натиснутій кнопці «Актуальність теми»

Приклад використання інтерфейсу Open WebUI:

- Користувач натискає кнопку “2. Формування гіпотези”;
- Шаблон промпту вставляється в поле вводу;
- Користувач доповнює його, натискає “Конспект лекції”
- Програма надсилає запит до моделі;
- Результат з’являється в нижньому полі виводу.

Інструкція користувачу

Інструкція користувача: Teacher AI GUI

1. Призначення програми

Програма Teacher AI GUI призначена для автоматизації навчальної та дослідницької роботи викладача університету з використанням штучного інтелекту. Вона інтегрується з локальною моделлю LM Studio та сервісом Hugging Face.

2. Основні функції

- Генерація навчального плану (силлабусу)
- Написання конспектів лекцій
- Створення тестових завдань
- Узагальнення текстів
- Переклад наукових текстів
- Використання шаблонів промтів для кожного етапу наукового дослідження

3. Запуск програми

Для запуску програми потрібно мати встановлений Python, бібліотеки tkinter, requests та transformers. Запустіть файл teacher_ai_gui.py в середовищі Python.

4. Інтерфейс користувача

- Поле введення: сюди вводиться тема або текст запиту
- Кнопки дій: запускають відповідні функції ШІ
- Кнопки шаблонів: вставляють підготовлені шаблони запитів
- Поле виводу: відображає відповідь ШІ

5. Робота з шаблонами

Натискаючи кнопку шаблону (наприклад, '1. Актуальність теми'), користувач вставляє рекомендований запит. Його можна редагувати та надіслати у модель, натиснувши одну з кнопок дій.

Роботу програми було протестовано на використанні моделі ШІ з веб-ресурсу Notebook LM для резюмування опублікованої статті на науково-технічній конференції студентів і співробітників підрозділів ВНТУ. Оскільки в даному випадку веб-інтерфейс передає управління на інтерфейс самої моделі, то результат роботи має вигляд, представлений на рис.3.3 і рис. 3.4.

Запит:

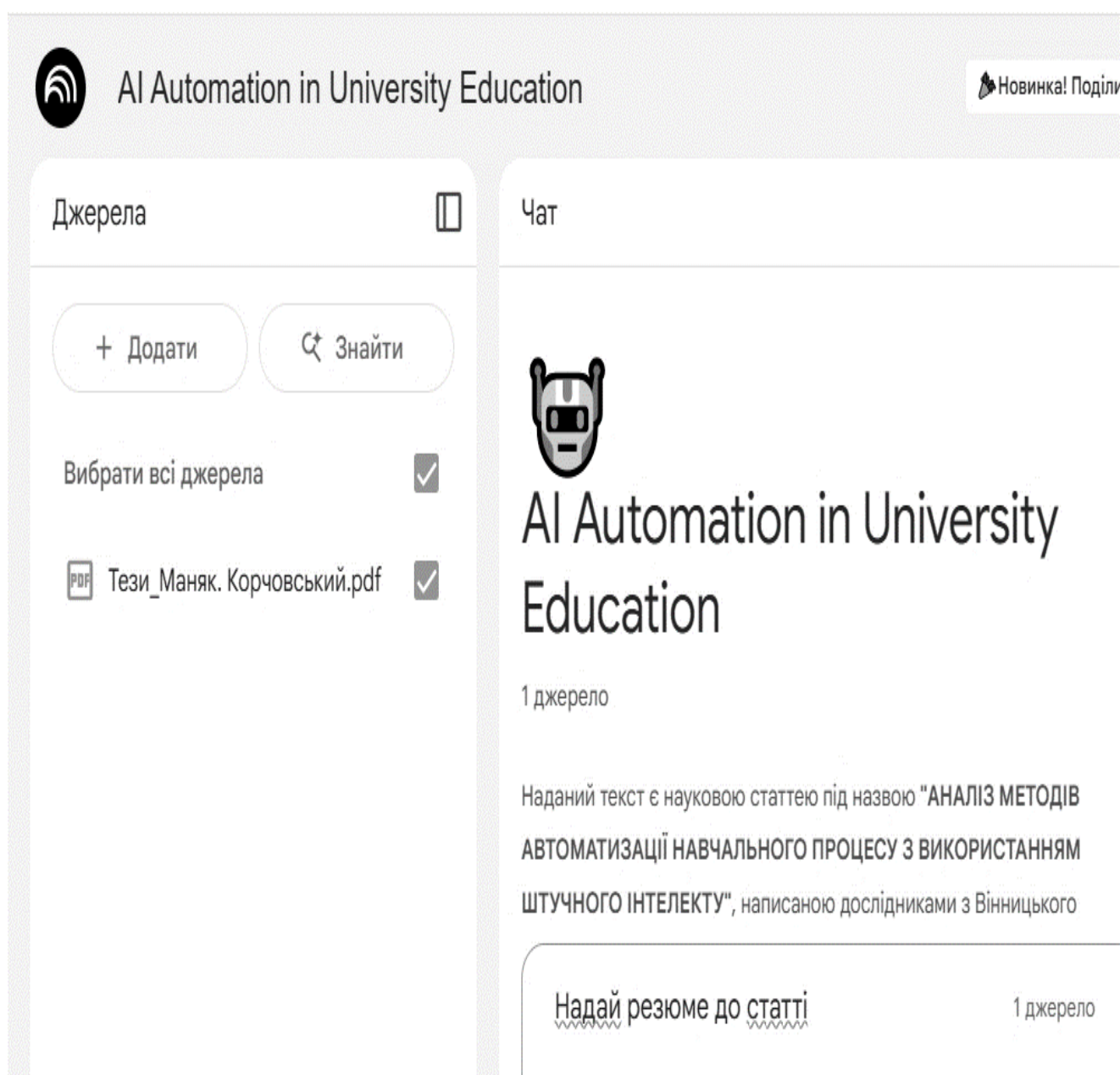


Рисунок 3.3 – Екран інтерфейсу з промптом до ШІ Notebook LM

Відповідь:

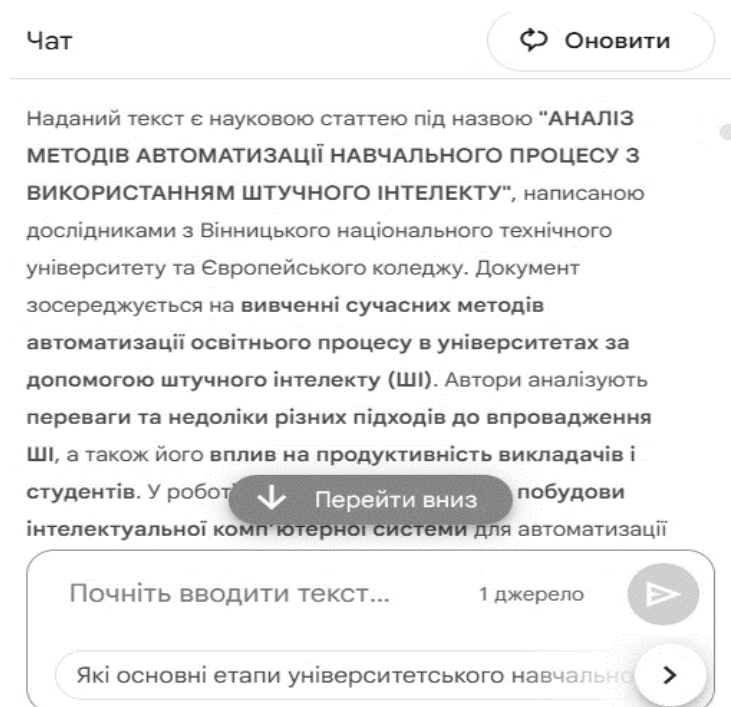


Рисунок 3.4 – Результат роботи системи

3.4 Висновки до розділу

У даному розділі було обґрунтовано вибір програмної платформи для реалізації роботи підсистеми автоматизації окремих етапів науково-дослідницької роботи викладача і розроблено програмне забезпечення для неї. Розроблено графічний веб-інтерфейс системи користувача, який дозволяє використовувати для автоматизації наукової роботи викладача як локальні моделі ШІ, так і веб-ресурси в локальному режимі. Розроблено інструкцію користувача програми.. Розроблено інструкцію користувача програми. Проведене тестування підтвердило адекватність прийнятих в роботі рішень.

ВИСНОВКИ

В даній Частині 2 комплексної роботи розроблено підсистему автоматизації етапів науково-дослідницької роботи для підвищення ефективності наукової роботи викладача.

В розділі 1 роботи на основі проведеного аналізу університетського навчального процесу як об'єкта автоматизації, визначено сучасні виклики в застосуванні технологій штучного інтелекту для автоматизації завдань науково-дослідної роботи, визначено перелік задач для досягнення поставленої мети.

Запропоновано концептуальну модель схемної побудови ефективної підсистеми автоматизації завдань науково-дослідницької роботи викладача університету за рахунок використання сервісів і моделей штучного інтелекту.

В розділі 2 проведено рейтингування існуючих сервісів ІІІ і їх вибір для інтеграції в комп'ютерну систему автоматизації викладацької діяльності, розроблено структуру підсистеми автоматизації з використанням локальних ресурсів на платформах LM Studio і Hugging Face і веб-інтерфейсу для використання хмарних сервісів. Розроблено UML-діаграми послідовності етапів науково-дослідної роботи та діаграму послідовності використання сервісів ІІІ на різних етапах дослідницької роботи.

В розділі 3 було обґрунтовано вибір програмної платформи для реалізації роботи підсистеми автоматизації окремих етапів науково-дослідницької роботи викладача і розроблено програмне забезпечення для неї. Розроблено графічний веб-інтерфейс системи користувача, який дозволяє використовувати для автоматизації наукової роботи викладача як локальні моделі ІІІ, так і веб-ресурси в локальному режимі. Розроблено інструкцію користувача програми.

Проведене тестування програмного забезпечення підтвердило адекватність прийнятих в роботі рішень.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Володимир Паніотто, Використання штучного інтелекту населенням України, Прес-реліз, Київський міжнародний інститут соціології, 12 травня 2025 р. Доступно: <https://kiis.com.ua/?lang=ukr&cat=reports&id=1528&page=1>
2. Lee I., Lee J., Lee K., Lee J., Hwang J.. “Effectiveness of artificial intelligence in education: A systematic review”, *Journal of Educational Technology & Society*, vol. 22(3), 2019. p. 47-63.
3. С.В Толочко, Н.С. Бордюг, Л.П. Міронець, Академічна доброчесність та штучний інтелект в освітній і науковій діяльності, 2023
4. Алла Коломієць, Олександра Юріївна Кушнір, “Використання штучного інтелекту в освітній та науковій діяльності: можливості та виклики”, *Modern Information Technologies and Innovation Methodologies of Education in Professional Training Methodology Theory Experience Problems*, Feb 2024. <https://doi.org/10.31652/2412-1142-2023-70-45-57>
5. Nestor Maslej. AI Index 2025, Stanford University, May, 2025. [Online], Available: <https://hai.stanford.edu/news/ai-index-2025-state-of-ai-in-10-charts>.
6. The Impact of Artificial Intelligence (AI) on Students' Academic Development - MDPI, Available: <https://www.mdpi.com/2227-7102/15/3/343> . [Accessed: 16-May-2025].
7. Top 10 AI Trends Reshaping the World of Education in 2024 - HurixDigital, <https://www.hurix.com/blogs/top-trends-reshaping-the-world-of-education-in/> . [Accessed: 16-May-2025].
8. Artificial intelligence in higher education institutions - *Frontiers*, <https://www.frontiersin.org/journals/education/articles/10.3389/feduc.2025.1530247/full> . [Accessed: 16-May-2025]
9. Lijia Chen, Pingping Chen, Zhijian Lin, “Artificial Intelligence in Education: A Review”, *IEEE Access*, vol. 8, 2020, p. 75264 – 7575268. AI Tools for Academic Research & Writing - Library Guides - Oklahoma State University, доступ отримано травня 12, 2025, <https://info.library.okstate.edu/AI/tools>
10. The Scientific Method: 5 Steps for Investigating Our World | AMNH, доступ

отримано травня 12, 2025, <https://www.amnh.org/explore/videos/the-scientific-process>

- 11.WWW: The Scientific Method - PMC - PubMed Central, доступ отримано травня 12, 2025, <https://pmc.ncbi.nlm.nih.gov/articles/PMC1635141/>
- 12.The scientific method (article) - Khan Academy, доступ отримано травня 12, 2025, <https://www.khanacademy.org/science/biology/intro-to-biology/science-of-biology/a/the-science-of-biology>
- 13.The Principles of Four Basic Steps of Scientific Stage: Problem, Hypothesis, Trial, Report, доступ отримано травня 12, 2025, https://www.researchgate.net/publication/371752957_The_Principles_of_Four_Basic_Steps_of_Scientific_Stage_Problem_Hypothesis_Trial_Report
- 14.Six Steps of the Scientific Method - ThoughtCo, доступ отримано травня 12, 2025, <https://www.thoughtco.com/steps-of-the-scientific-method-p2-606045>
- 15.Steps of the Scientific Method - Science Buddies, доступ отримано травня 12, 2025, <https://www.sciencebuddies.org/science-fair-projects/science-fair/steps-of-the-scientific-method>
- 16.Scientific Method Steps in Psychology Research - Verywell Mind, доступ отримано травня 12, 2025, <https://www.verywellmind.com/steps-of-the-scientific-method-2795782>
- 17.What Are The Steps Of The Scientific Method? - Simply Psychology, доступ отримано травня 12, 2025, <https://www.simplypsychology.org/steps-of-the-scientific-method.html>
- 18.AI Tools for Research: Working with data - Research Guides - Temple University, доступ отримано травня 12, 2025, <https://guides.temple.edu/ai-research-tools/data>
- 19.Steps in the Scientific Process - GLOBE.gov, доступ отримано травня 12, 2025, <https://www.globe.gov/do-globe/resources/student-resources/be-a-scientist/steps-in-the-scientific-process>
- 20.Best AI Research Tools for Academics and Researchers - Litmaps, доступ отримано травня 12, 2025, <https://www.litmaps.com/learn/best-ai-research-tools>

- 21.10+ AI Tools You Can Start Using For Free | Google Cloud, доступ отримано травня 12, 2025, <https://cloud.google.com/use-cases/free-ai-tools>
- 22.Julius AI | Your AI Data Analyst, доступ отримано травня 12, 2025, <https://julius.ai/>
- 23.12 Powerful Free AI Tools for Research In 2024 - Numerous.ai, доступ отримано травня 12, 2024, <https://numerous.ai/blog/free-ai-tools-for-research>
- 24.10 Free AI Tools for Data Analysis - Numerous.ai, доступ отримано травня 12, 2025, <https://numerous.ai/blog/free-ai-tools-for-data-analysis>
- 25.Artificial Intelligence (Generative) Resources: AI Tools for Research - Guides, доступ отримано травня 12, 2025, <https://guides.library.georgetown.edu/ai/tools>
- 26.10 free online tools for scientific research - SciNote, доступ отримано травня 12, 2025, <https://www.scinote.net/blog/10-free-online-tools-for-scientific-research/>
- 27.What are the best AI tools for research? : r/PhD - Reddit, доступ отримано травня 12, 2025, https://www.reddit.com/r/PhD/comments/1fad9ry/what_are_the_best_ai_tools_for_research/
- 28.scienceOS - The AI research tool for scientists - Try for free!, доступ отримано травня 12, 2025, <https://www.scienceos.ai/>
- 29.AI-Based Literature Review Tools - Research Guides at Texas A&M University, доступ отримано травня 12, 2025, <https://tamu.libguides.com/c.php?g=1289555>
- 30.I tested every ai literature review tool so you don't have to (8 best options for 2025), доступ отримано травня 12, 2025, <https://techpoint.africa/guide/best-ai-tools-for-literature-reviews/>
- 31.Wellness Wednesday - Some Useful AI Literature Review Tools : r/PhD - Reddit, доступ отримано травня 12, 2025, https://www.reddit.com/r/PhD/comments/13r3u0n/wellness_wednesday_some_useful_ai_literature/
- 32.Elicit: The AI Research Assistant, доступ отримано травня 12, 2025, <https://elicit.com/>
- 33.11 Best AI Tools for Research Paper Writing in 2025 - Blainy, доступ отримано

- травня 12, 2025, <https://blainy.com/best-ai-tools-for-research-paper-writing/>
- 34.Consensus: AI-powered Academic Search Engine, доступ отримано травня 12, 2025, <https://consensus.app/>
- 35.AI Writer Tool & Content Generator | Trusted by 1M+ Researchers - SciSpace, доступ отримано травня 12, 2025, <https://scispace.com/ai-writer>
- 36.5 Unbelievably Useful AI Tools For Research in 2025 (better than ChatGPT) - YouTube, доступ отримано травня 12, 2025, <https://www.youtube.com/watch?v=wmQVdzBRnN1>
- 37.AI Paper Writer - Write Paper & Essay with Sources for Free - Paperguide, доступ отримано травня 12, 2025, <https://paperguide.ai/writer/>
- 38.Free AI Writing Resources | Scribbr, доступ отримано травня 12, 2025, <https://www.scribbr.com/ai-writing/>
- 39.AI Literature Review Generator | Free Account - Jenni AI, доступ отримано травня 12, 2025, <https://jenni.ai/literature-review-generator>
- 40.Jenni AI, доступ отримано травня 12, 2025, <https://jenni.ai/>
- 41.Paperpal: AI Academic Writing Tool - Online English Language Check, доступ отримано травня 12, 2025, <https://paperpal.com/>
- 42.7 Free AI Tools So Good, Academics Can't Believe They're Free - YouTube, доступ отримано травня 12, 2025, <https://www.youtube.com/watch?v=vL12MBT1JZw>
- 43.AI tools that have actually been helpful with my research : r/academia - Reddit, доступ отримано травня 12, 2025, https://www.reddit.com/r/academia/comments/1ayadir/ai_tools_that_have_actually_been_helpful_with_my/
44. AI tools for literature Review: r/PhD - Reddit, доступ отримано травня 12, 2025, https://www.reddit.com/r/PhD/comments/17nqf59/ai_tools_for_literature_review/
- 45.Yossi Matias, How we're using AI to drive scientific research with greater real-world benefit, 2025. <https://blog.google/technology/research/google-research-scientific-discovery/>
- 46.Juraj Gottweis, Vivek Natarajan, "Accelerating scientific breakthroughs with an AI

co-scientist”, February, 2025. <https://research.google/blog/accelerating-scientific-breakthroughs-with-an-ai-co-scientist/>

47. Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, and other, “Towards an AI co-scientist”, 2025.

https://storage.googleapis.com/coscientist_paper/ai_coscientist.pdf

48. Formula Bot: AI for Data Analysis, доступ отримано травня 12, 2025, <https://www.formulabot.com/>

49. Best Free Data Analysis Programs for Beginners - Insight7, доступ отримано травня 12, 2025, <https://insight7.io/best-free-data-analysis-programs-for-beginners/>

50. Here are the Top AI Tools for Research Data Analysis - YouTube, доступ отримано травня 12, 2025, https://www.youtube.com/watch?v=_7JyuA1nAKk

ДОДАТКИ

ДОДАТОК А

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: Розробка інтелектуальної комп'ютерної системи управління навчальним процесом із використанням методів автоматизації та аналізу даних. Ч2: Підсистема автоматизації етапів дослідницької роботи.

Тип роботи: бакалаврська кваліфікаційна робота
(бакалаврська кваліфікаційна робота / магістерська кваліфікаційна робота)

Підрозділ кафедра КСУ
(кафедра, факультет, навчальна група)

Коефіцієнт подібності текстових запозичень, виявлених у роботі системою StrikePlagiarism 2,9 %

Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне)

- ☒ Запозичення, виявлені у роботі, є законними і не містять ознак плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту
- ☐ У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але надмірна кількість текстових запозичень та/або наявність типових розрахунків не дозволяють прийняти рішення про оригінальність та самостійність її виконання. Роботу направити на доопрацювання.
- ☐ У роботі виявлено ознаки плагіату та/або текстових маніпуляцій як спроб укриття плагіату, фабрикації, фальсифікації, що суперечить вимогам законодавства та нормам академічної доброчесності. Робота до захисту не приймається.

Експертна комісія:

Завідувач кафедри КСУ
(прізвище, ініціали, посада)

(підпис)

В'ячеслав КОВТУН

Гарант ОПП
(прізвище, ініціали, посада)

(підпис)

Олег КОВАЛЮК

Особа, відповідальна за перевірку

(підпис)

Володимир ДУБОВОЙ
(прізвище, ініціали)

З висновком експертної комісії ознайомлений(-на)

Керівник

(підпис)

Микола БИКОВ

(прізвище, ініціали, посада)

Здобувач

(підпис)

Дмитро КОРЧОВСЬКИЙ

(прізвище, ініціали)

ДОДАТОК Б
(обов'язковий)
ВНТУ

ЗАТВЕРДЖЕНО
Завідувач кафедри КСУ
д.т.н., проф.

 В'ячеслав КОВТУН

“ 23 ” травня 2025 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання бакалаврської кваліфікаційної роботи

РОЗРОБКА ІНТЕЛЕКТУАЛЬНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ УПРАВЛІННЯ
НАВЧАЛЬНИМ ПРОЦЕСОМ ІЗ ВИКОРИСТАННЯМ МЕТОДІВ
АВТОМАТИЗАЦІЇ ТА АНАЛІЗУ ДАНИХ.

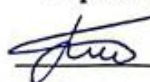
ЧАСТИНА 2.ПІДСИСТЕМА АВТОМАТИЗАЦІЇ ЕТАПІВ ДОСЛІДНИЦЬКОЇ
РОБОТИ

08-33.БДР.010.00.000 ТЗ

Студент групи 2 АКІТ-216

 Дмитро КОРЧОВСЬКИЙ

Керівник: к.т.н., доцент

 Микола БИКОВ.

Вінниця 2025

1. Назва та галузь застосування

1.1. Назва – Розробка інтелектуальної комп'ютерної системи управління навчальним процесом із використанням методів автоматизації та аналізу даних. Частина 2. Підсистема автоматизації етапів дослідницької роботи.

1.2. Галузь застосування – Автоматизація, комп'ютерно-інтегровані технології та робототехніка.

2. Підстава для проведення розробки.

Тема бакалаврської кваліфікаційної роботи затверджена наказом по ВНТУ № 97 від 20 березня 2025 р.

3. Мета та призначення розробки.

Метою бакалаврської кваліфікаційної роботи є підвищення ефективності навчального процесу в університеті за рахунок розробки та впровадження підсистеми управління дослідницькою роботою на основі технологій штучного інтелекту.

4. Джерела розробки.

Бакалаврська кваліфікаційна робота виконується вперше. В ході проведення розробки повинні використовуватись такі документи:

1. Биков М.М., Бикова О.М, Корчовський Д.К., Маняк В.О. Аналіз методів автоматизації навчального процесу з використанням штучного інтелекту. - [Електронний ресурс] – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2025/paper/view/24426/20212>

2. Lee I., Lee J., Lee K., Lee J., Hwang J. Effectiveness of artificial intelligence in education: A systematic review, *Journal of Educational Technology & Society*, 22(3), 2019, p. 47-63.

3. Алла Коломієць, Олександра Юріївна Кушнір, “Використання штучного інтелекту в освітній та науковій діяльності: можливості та виклики”, *Modern Information Technologies and Innovation Methodologies of Education in Professional Training Methodology Theory Experience Problems*, Feb 2024. <https://doi.org/10.31652/2412-1142-2023-70-45-57>

5. Вимоги до розробки.

5.1. Перелік головних функцій:

- формування запиту до агента ІІІ системи про виконання навчального завдання;
- автоматичний вибір потрібного сервісу ІІІ;
- оформлення отриманої відповіді у вигляді потрібного навчального продукту.

5.2. Основні технічні вимоги до розробки.

5.2.1. Вимоги до програмної платформи:

- WINDOWS 2007\10\11;
- Hugging Face, LM Studio;
- Python, Java, Plantum ML, Streamlit.

5.2.2. Умови експлуатації системи:

- робота на стандартних ПК в приміщеннях зі стандартними умовами;
- можливість цілодобового функціонування системи;
- текст програмного забезпечення системи є цілком закритим.

6. Стадії та етапи розробки.

6.1 Пояснювальна записка:

№ з/п	Назва етапів бакалаврської кваліфіційної роботи	Термін виконання етапів роботи
1	Розробка технічного завдання	10 травня 2025 р.
2	Аналіз об'єкта навчального процесу як об'єкта автоматизації	15 травня 2025 р.
3	Вибір сервісів ІІІ і обладнання для підсистеми автоматизації дослідницької роботи	18 травня 2025 р.
4	Розробка структурної схеми системи	26 травня 2025 р.
5	Розробка алгоритму роботи системи	28 травня 2025 р.
6	Розробка програмного забезпечення системи	02 червня 2025 р.
7	Машинний експеримент	9 червня 2025 р.
8	Оформлення пояснювальної записки	10 червня 2025 р.

6.2 Графічні матеріали:

- Концептуальна схема системи управління навчальним процесом – плакат - Схема структурна ІКСУ - Схема алгоритму роботи підсистеми - UML-діаграми - Результати машинного експерименту	03 червня 2025 р. 05 червня 2025 р. 07 червня 2025 р. 09 червня 2025 р. 09 червня 2025 р..
--	--

7. Порядок контролю і приймання.

- 7.1. Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до «08» червня 2025 р.
- 7.2. Атестація роботи здійснюється на попередньому захисті. Попередній захист бакалаврської кваліфікаційної роботи провести до «10» червня 2025 р.
- 7.3. Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ЕК.
- 7.4. Захист бакалаврської кваліфікаційної роботи провести до «18» червня 2025р.

Додаток В

(ДОВІДКОВИЙ)

Лістинг програми інтерфейсу AI Teachures GUI

```
import tkinter as tk
from tkinter import messagebox, scrolledtext
import requests
from transformers import pipeline
# ===== Конфігурація =====
HF_TOKEN = "hf_xxx" # <-- замінити на свій Hugging Face API токен
HF_HEADERS = {"Authorization": f"Bearer {HF_TOKEN}"}
LM_STUDIO_URL = "http://localhost:1234/v1/chat/completions"
# ===== Виклик LM Studio =====
def ask_lm_studio(prompt):
    payload = {
        "model": "local-model", # назва локальної моделі
        "messages": [{"role": "user", "content": prompt}],
        "temperature": 0.7,
        "max_tokens": 1024
    }
    try:
        response = requests.post(LM_STUDIO_URL, json=payload)
        return response.json()['choices'][0]['message']['content']
    except Exception as e:
        return f"[LM Studio Error] {e}"
# ===== Виклик Hugging Face =====
def ask_huggingface(model_id, prompt):
    url = f"https://api-inference.huggingface.co/models/{model_id}"
    data = {"inputs": prompt}
    try:
        response = requests.post(url, headers=HF_HEADERS, json=data)
        return response.json()[0]['generated_text'] if isinstance(response.json(), list) else
str(response.json())
    except Exception as e:
        return f"[HF Error] {e}"
# ===== Узагальнення тексту через pipeline =====
def summarize_text(text):
    try:
        summarizer = pipeline("summarization", model="facebook/bart-large-cnn")
        summary = summarizer(text, max_length=150, min_length=50, do_sample=False)
        return summary[0]['summary_text']
    except Exception as e:
        return f"[Summarization Error] {e}"
# ===== Промпт шаблони =====
def insert_prompt(prompt_text):
    input_text.delete("1.0", tk.END)
    input_text.insert(tk.END, prompt_text)
# ===== Основна GUI логіка =====
def handle_action(action):
    prompt = input_text.get("1.0", tk.END).strip()
```



```

if not prompt:
    messagebox.showwarning("Помилка", "Введіть текст запиту!")
    return
if action == "plan":
    result = ask_lm_studio(f"Сформулюй силабус і план курсу для теми: {prompt}")
elif action == "lecture":
    result = ask_lm_studio(f"Напиши конспект лекції на тему: {prompt}")
elif action == "test":
    result = ask_lm_studio(f"Створи тестові запитання на тему: {prompt}")
elif action == "summarize":
    result = summarize_text(prompt)
elif action == "translate":
    result = ask_huggingface("Helsinki-NLP/opus-mt-uk-en", prompt)
else:
    result = "[Невідома дія]"
output_text.delete("1.0", tk.END)
output_text.insert(tk.END, result)
# ===== Побудова GUI =====
window = tk.Tk()
window.title("AI Асистент Викладача")
window.geometry("1000x700")
frame = tk.Frame(window)
frame.pack(pady=10)
label = tk.Label(frame, text="Введіть тему або текст:")
label.pack()
input_text = scrolledtext.ScrolledText(frame, height=5)
input_text.pack(padx=10, pady=5, fill=tk.X)
# ===== Кнопки дій =====
btn_frame = tk.Frame(window)
btn_frame.pack(pady=5)
btn1 = tk.Button(btn_frame, text="План курсу", command=lambda: handle_action("plan"))
btn2 = tk.Button(btn_frame, text="Конспект лекції", command=lambda: handle_action("lecture"))
btn3 = tk.Button(btn_frame, text="Тестові завдання", command=lambda: handle_action("test"))
btn4 = tk.Button(btn_frame, text="Узагальнення", command=lambda: handle_action("summarize"))
btn5 = tk.Button(btn_frame, text="Переклад", command=lambda: handle_action("translate"))
for b in (btn1, btn2, btn3, btn4, btn5):
    b.pack(side=tk.LEFT, padx=5)
# ===== Кнопки шаблонів =====
prompt_frame = tk.LabelFrame(window, text="Шаблони промптів для етапів дослідження")
prompt_frame.pack(padx=10, pady=10, fill=tk.X)
tk.Button(prompt_frame, text="1. Актуальність теми", width=25, command=lambda:
insert_prompt("Обґрунтуй актуальність теми: [вказіть тему].")) .pack(side=tk.LEFT, padx=5,
pady=5)
tk.Button(prompt_frame, text="2. Формування гіпотези", width=25, command=lambda:
insert_prompt("Сформулюй гіпотезу дослідження на тему: [вказіть тему].")) .pack(side=tk.LEFT,
padx=5, pady=5)
tk.Button(prompt_frame, text="3. Методологія дослідження", width=25, command=lambda:
insert_prompt("Опиши методику дослідження для перевірки гіпотези: [вказіть]."))
.pack(side=tk.LEFT, padx=5, pady=5)
tk.Button(prompt_frame, text="4. Анкета для збору даних", width=25, command=lambda:
insert_prompt("Створи анкету з 5-7 питань для вивчення теми: [вказіть].")) .pack(side=tk.LEFT,
padx=5, pady=5)

```

```

tk.Button(prompt_frame, text="5. Аналіз результатів", width=25, command=lambda:
insert_prompt("Поясни результати дослідження з таблиці: [вставте дані].")) .pack(side=tk.LEFT,
padx=5, pady=5)
tk.Button(prompt_frame, text="6. Висновки дослідження", width=25, command=lambda:
insert_prompt("Сформулюй висновки та рекомендації щодо теми: [вказіть]."))
.pack(side=tk.LEFT, padx=5, pady=5)
tk.Button(prompt_frame, text="7. Структура статті", width=25, command=lambda:
insert_prompt("Запропонуй структуру наукової статті для теми: [вказіть].")) .pack(side=tk.LEFT,
padx=5, pady=5)
tk.Button(prompt_frame, text="8. Тези доповіді", width=25, command=lambda:
insert_prompt("Склади тези для наукової доповіді з теми: [вказіть тему].")) .pack(side=tk.LEFT,
padx=5, pady=5)
# ==== Вивід результату ====
output_label = tk.Label(window, text="Результат:")
output_label.pack()
output_text = scrolledtext.ScrolledText(window, height=20)
output_text.pack(padx=10, pady=5, fill=tk.BOTH, expand=True)
window.mainloop()

```

Додаток Г
(обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА
РОЗРОБКА ІНТЕЛЕКТУАЛЬНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ УПРАВЛІННЯ
НАВЧАЛЬНИМ ПРОЦЕСОМ ІЗ ВИКОРИСТАННЯМ МЕТОДІВ
АВТОМАТИЗАЦІЇ ТА АНАЛІЗУ ДАНИХ.
ЧАСТИНА 2. ПІДСИСТЕМА АВТОМАТИЗАЦІЇ ЕТАПІВ
ДОСЛІДНИЦЬКОЇ РОБОТИ

Студент групи

2 АКІТ-216


Підпис

Дмитро КОРЧОВСЬКИЙ
Ім'я ПРІЗВИЩЕ

Керівник к.т.н., доцент, проф. кафедри КСУ


Підпис

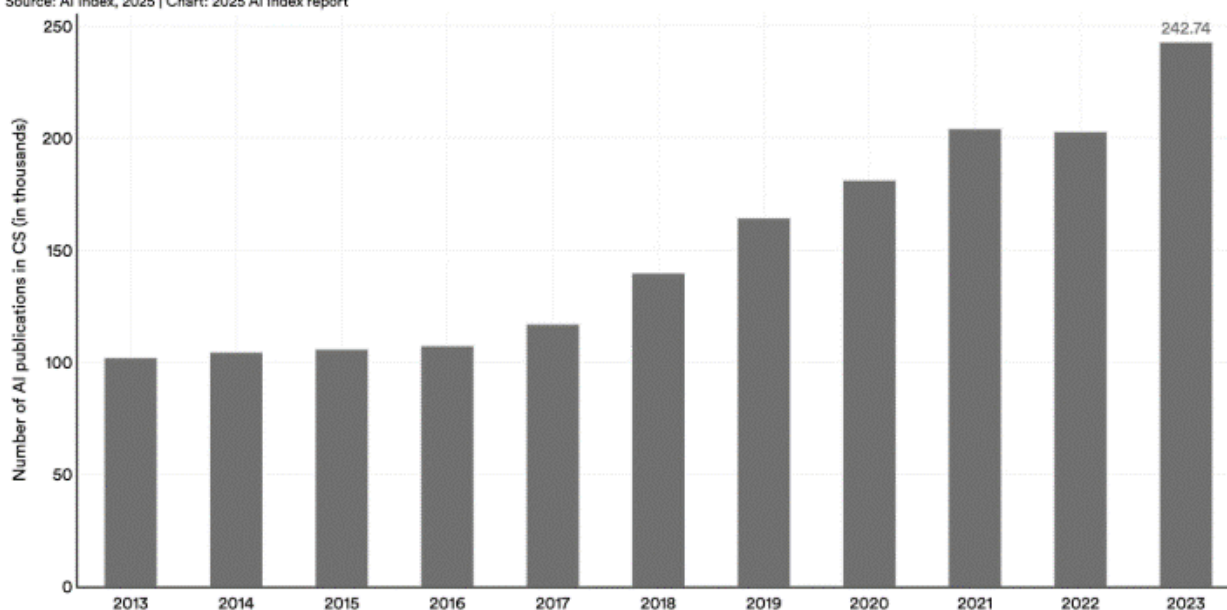
Микола БИКОВ
Ім'я ПРІЗВИЩЕ

Вінниця 2025

АКТУАЛЬНІСТЬ ПРОБЛЕМИ ВПРОВАДЖЕННЯ ШТУЧНОГО ІНТЕЛЕКТУ В ОСВІТНІЙ ПРОЦЕС УНІВЕРСИТЕТУ

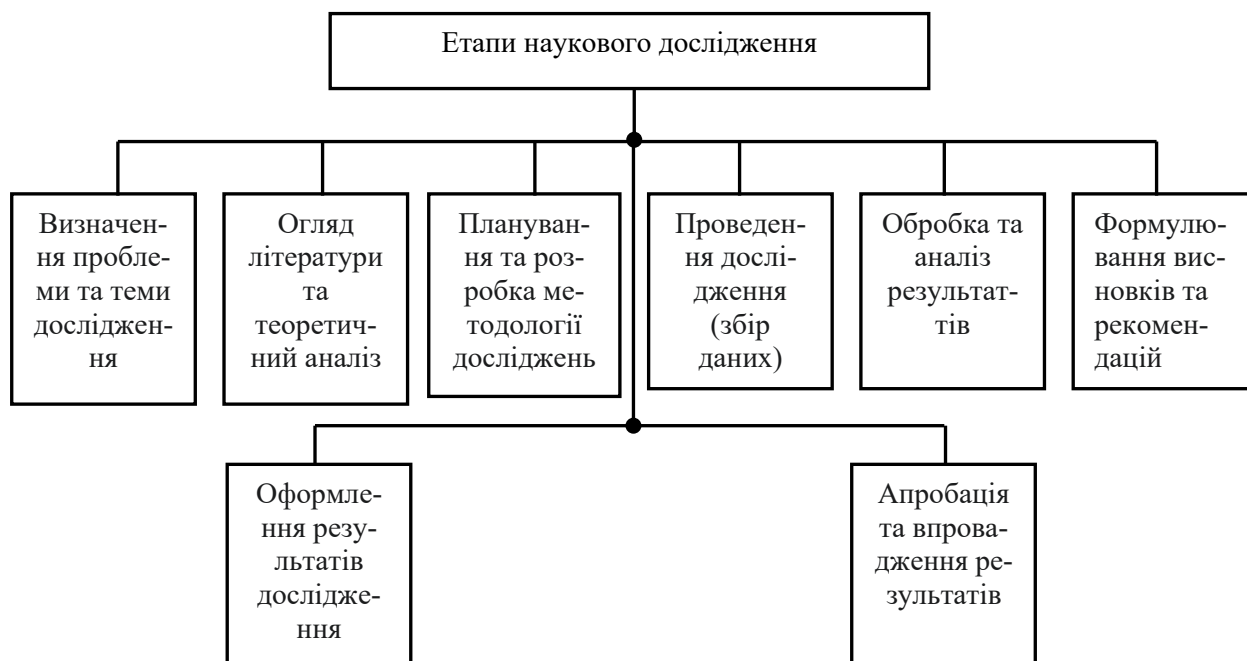
Number of AI publications in CS worldwide, 2013–23

Source: AI Index, 2025 | Chart: 2025 AI Index report

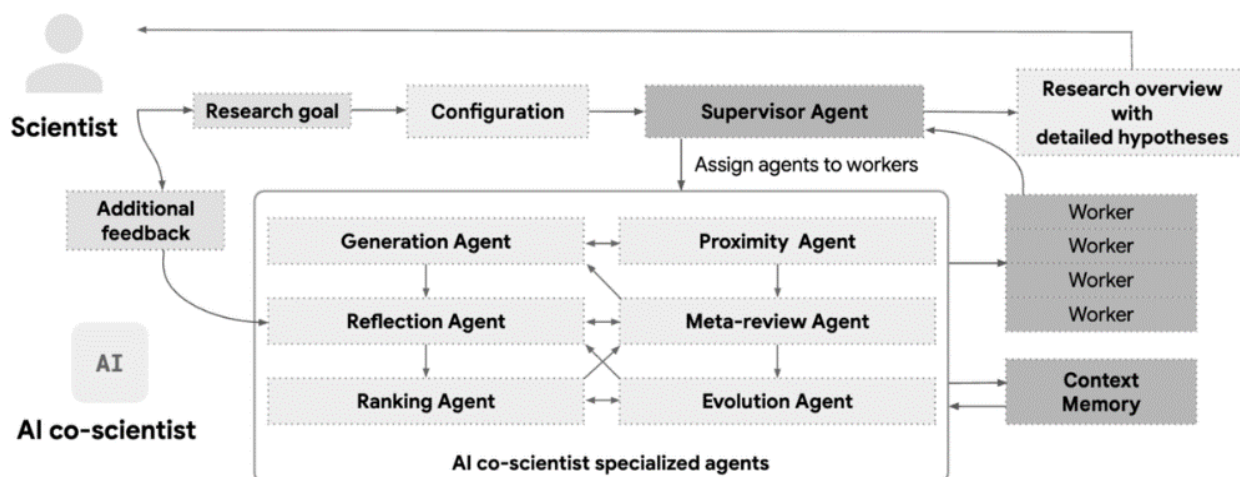


Діаграма зростання інтересу до ШІ з огляду на кількість опублікованих робіт по темі

ЕТАПИ ДОСЛІДНИЦЬКОЇ РОБОТИ ВИКЛАДАЧА УНІВЕРСИТЕТУ



АРХІТЕКТУРА ПЕРЕДОВОЇ ДОСЛІДНИЦЬКОЇ СИСТЕМИ ВІД GOOGLE RESEARCH



ОБРАНИ СЕРВІСИ ДЛЯ ІНТЕГРАЦІЇ В ПІДСИСТЕМУ АВТОМАТИЗАЦІЇ ДОСЛІДНИЦЬКОЇ РОБОТИ

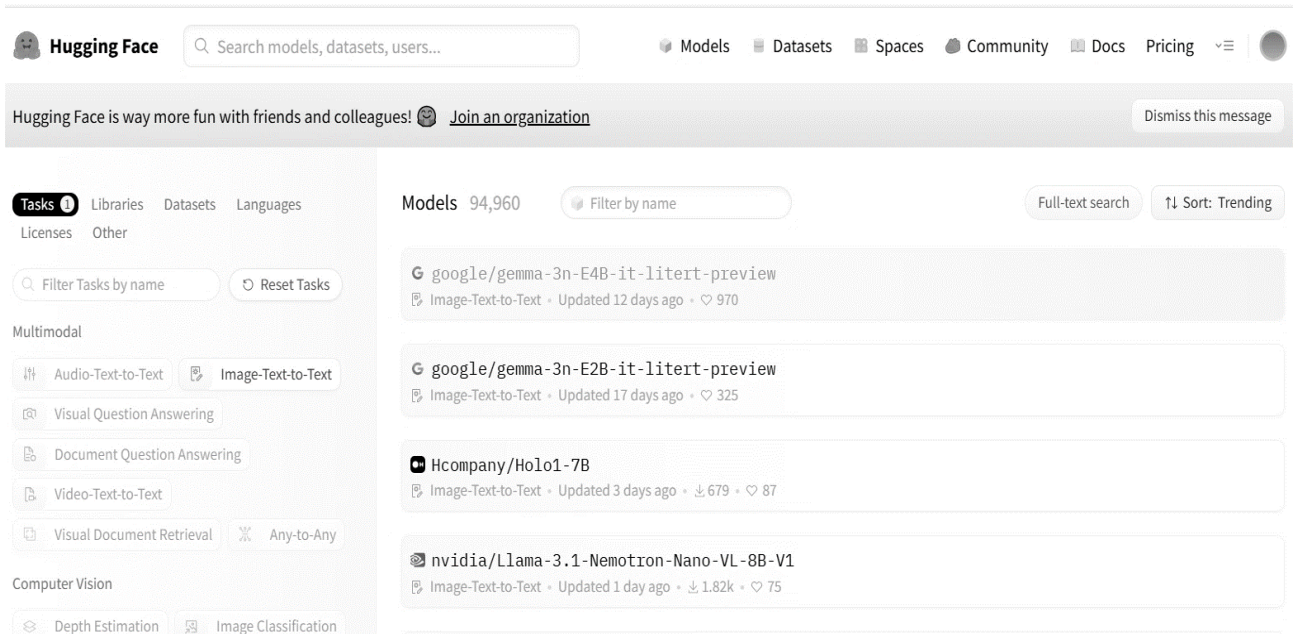
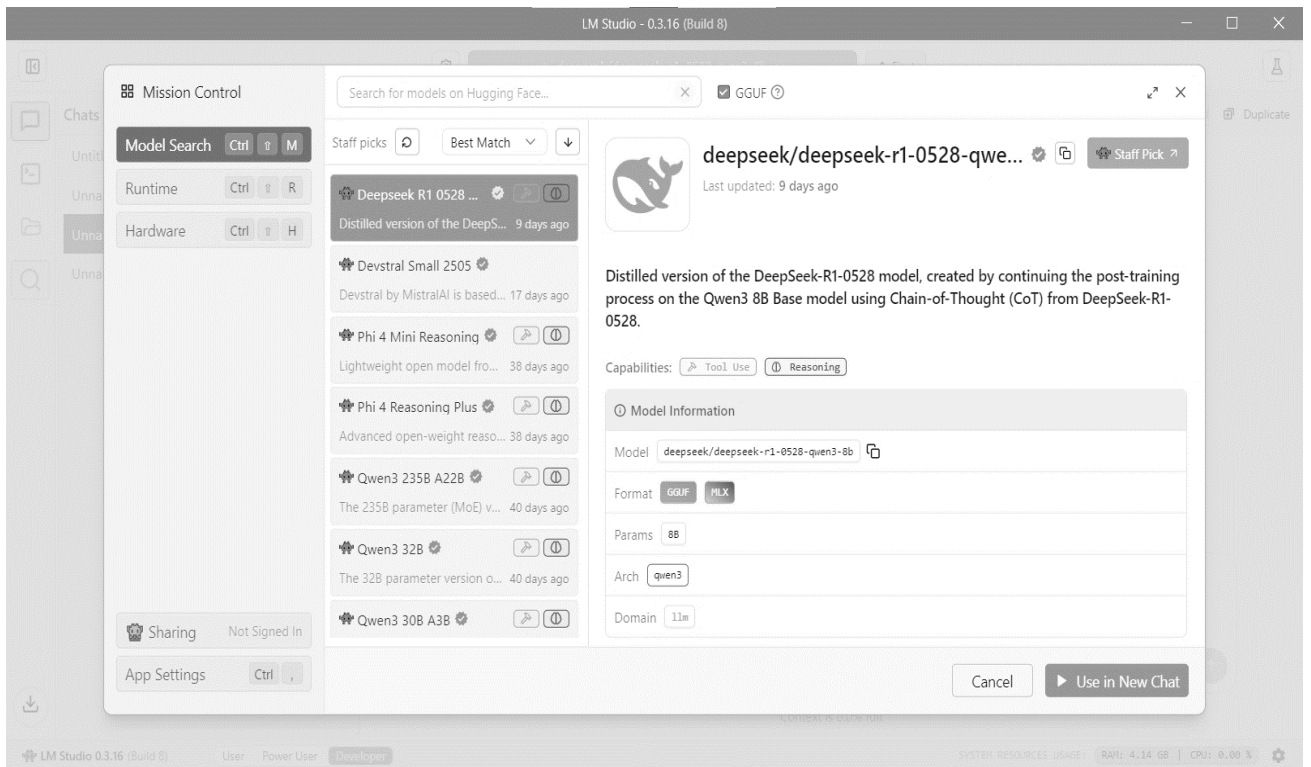
№	Сервіс / Модель	Критерії (1–6)	Обґрунтування ефективності
1	ChatGPT 4o (chat.openai.com)	+	Найкраще покриває всі етапи дослідження. Працює у браузері, доступна безкоштовна версія, підтримує аналітику, реферування, гіпотези, письмове оформлення.
2	Perplexity AI	+	Інформаційний пошук, формування проблеми, гіпотези. Дає джерела. Відмінний інструмент 80/20.
3	LM Studio + Hugging Face (локально)	+	Повністю безкоштовне локальне середовище, з моделями типу Mistral, LLaMA, Phi, Gemma. Підходить для генерації текстів, резюме, завдань, висновків.
4	Elicit (elicit.org)	+	Формулювання гіпотез, аналіз джерел, обґрунтування актуальності. Не всі функції доступні безплатно.
5	SciSpace Copilot	+	Зручний для аналізу PDF статей, формулювання висновків. Безкоштовна версія обмежена.
6	Hugging Face Spaces (наприклад, <i>summarizers</i> , <i>translators</i> , <i>Q&A</i>)	+	Багато моделей на одній платформі. Працюють як окремі утиліти або через API.
7	Google Bard / Gemini	+	Альтернатива ChatGPT. Відмінно справляється з ідеями, планами, текстами, але гірша точність джерел.

Етап дослідження	Найкращі інструменти для автоматизації
1. Визначення проблеми і теми	Perplexity, ChatGPT, Elicit
2. Теоретичний огляд, гіпотеза	ChatGPT, Elicit, SciSpace
3. Планування та методологія	ChatGPT, LM Studio, Hugging Face
4. Збір даних	ChatGPT (анкети), LM Studio, Q&A-боти
5. Аналіз і обробка даних	LM Studio, Hugging Face, Google Sheets (AI)
6. Висновки та рекомендації	ChatGPT, Gemini, LM Studio
7. Написання тексту	ChatGPT, LM Studio
8. Апробація, публікації	ChatGPT, Perplexity, SciSpace

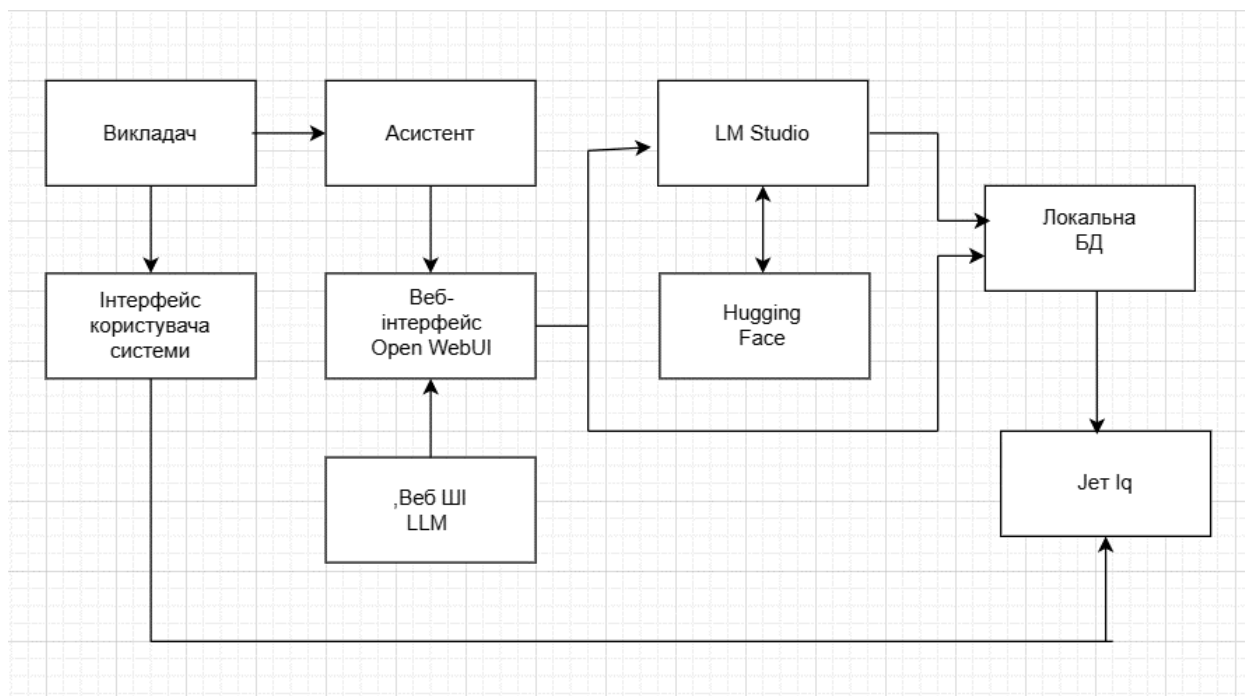
ПОСЛІДОВНІСТЬ ВИКОРИСТАННЯ СЕРВІСІВ І МОДЕЛЕЙ ШІ ПІД ЧАС ВИКОНАННЯ ЕТАПІВ НАВЧАЛЬНОЇ РОБОТИ



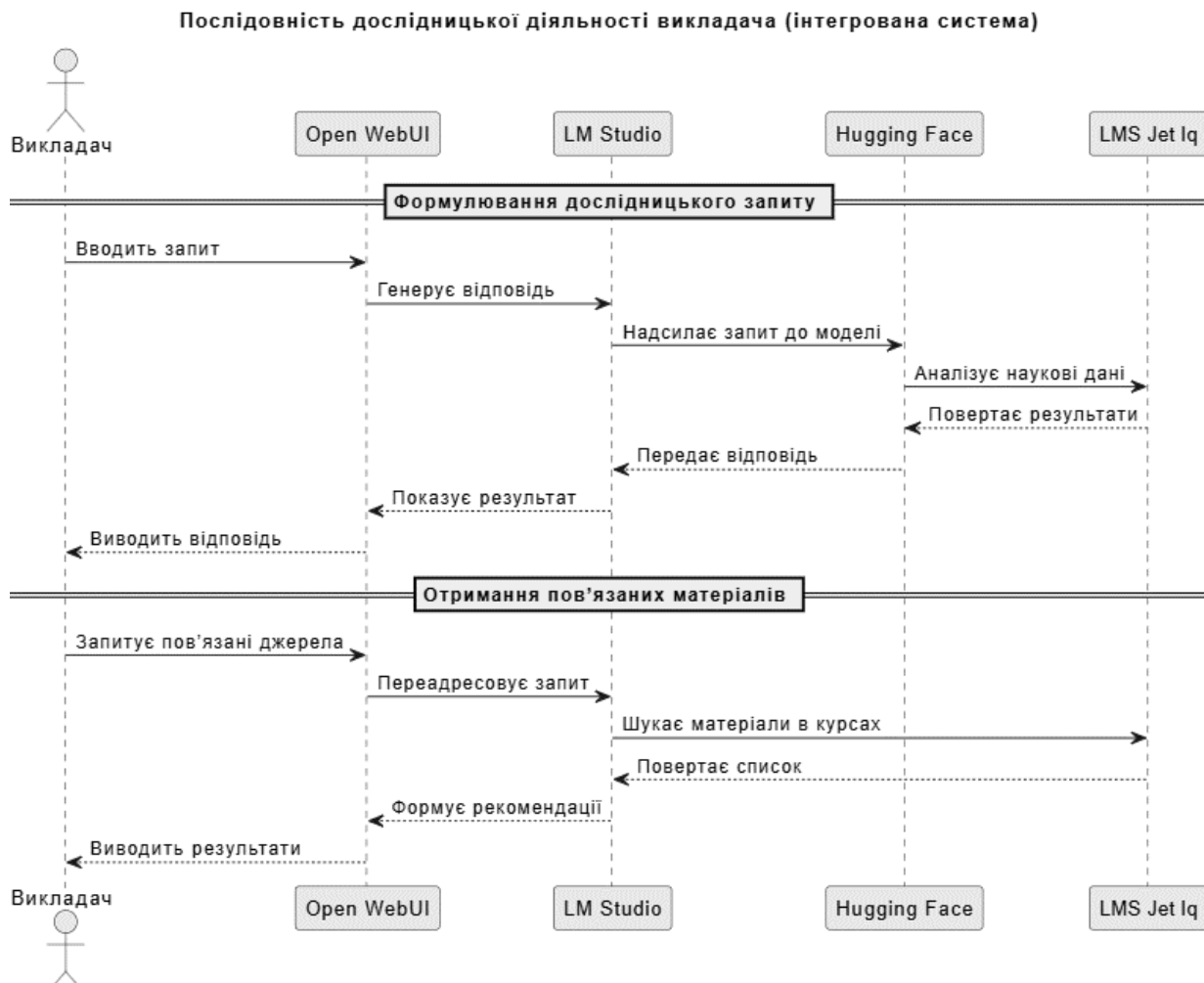
LM STUDIO I HUGGING FACE



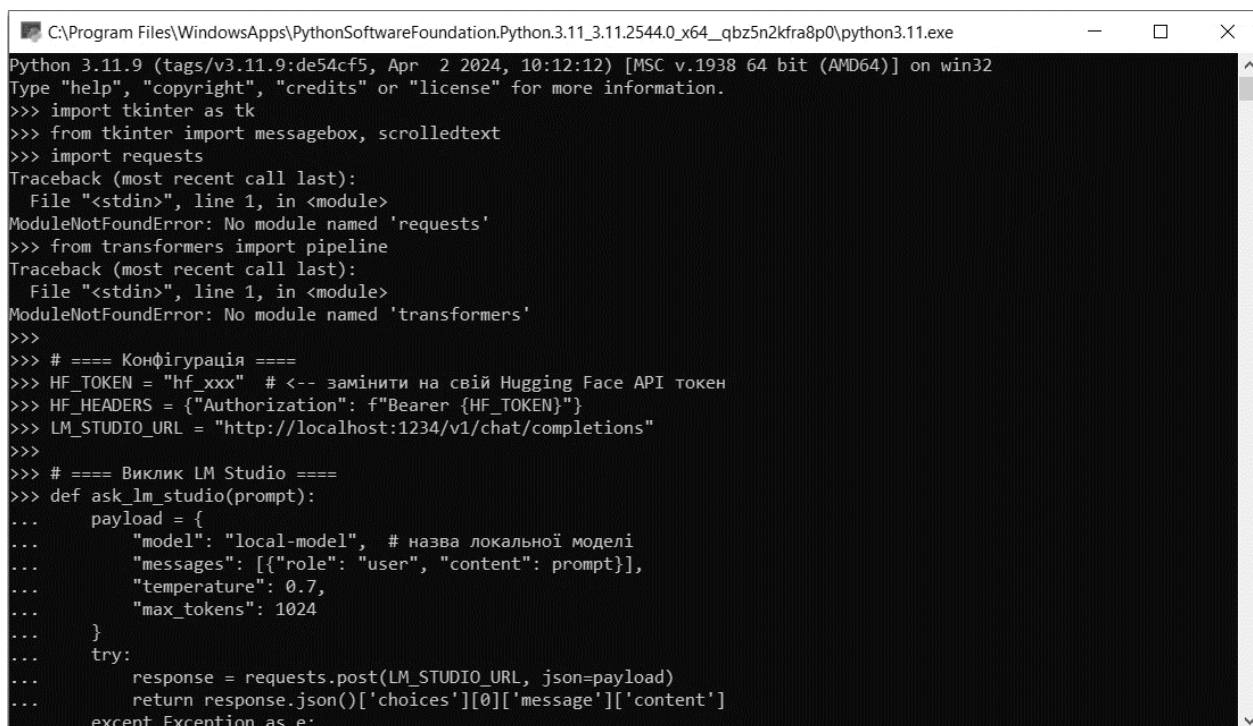
СТРУКТУРНА СХЕМА ПІДСИСТЕМИ АВТОМАТИЗАЦІЇ НАУКОВОЇ РОБОТИ ВИКЛАДАЧА



UML-ДІАГРАМА ПОСЛІДОВНОСТІ ЕТАПІВ ДОСЛІДНИЦЬКОЇ ДІЯЛЬНОСТІ ВИКЛАДАЧА УНІВЕРСИТЕТУ



ЕКРАН ПРОГРАМНОГО СЕРЕДОВИЩА З ТЕКСТОМ ПРОГРАМИ

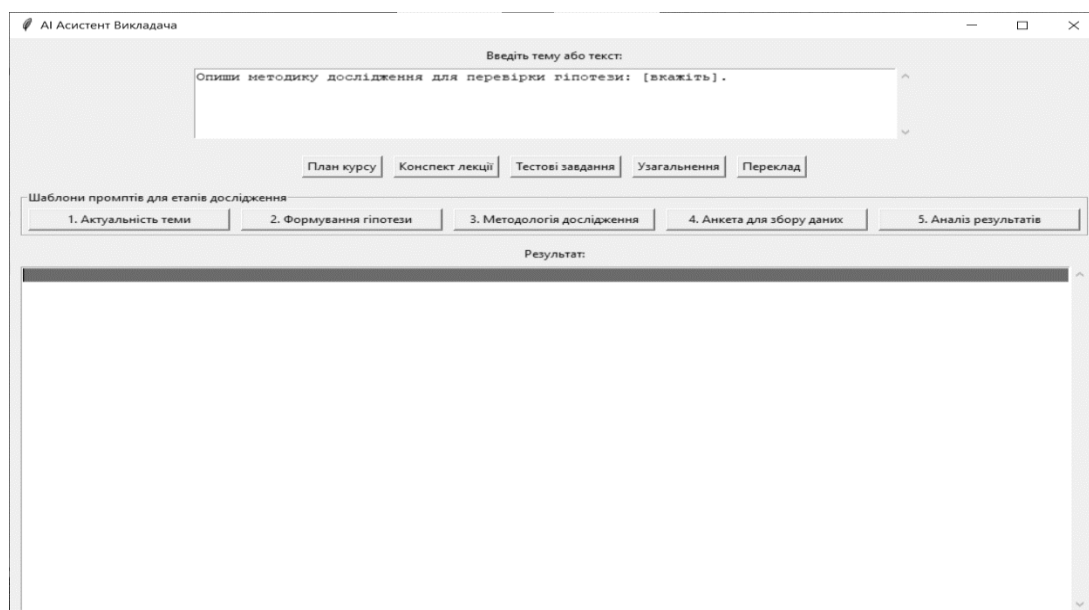


```

C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.2544.0_x64_qbz5n2kfra8p0\python3.11.exe
Python 3.11.9 (tags/v3.11.9:de54cf5, Apr  2 2024, 10:12:12) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import tkinter as tk
>>> from tkinter import messagebox, scrolledtext
>>> import requests
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'requests'
>>> from transformers import pipeline
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'transformers'
>>>
>>> # ==== Конфігурація ====
>>> HF_TOKEN = "hf_xxx" # <-- замінити на свій Hugging Face API токен
>>> HF_HEADERS = {"Authorization": f"Bearer {HF_TOKEN}"}
>>> LM_STUDIO_URL = "http://localhost:1234/v1/chat/completions"
>>>
>>> # ==== Виклик LM Studio ====
>>> def ask_lm_studio(prompt):
...     payload = {
...         "model": "local-model", # назва локальної моделі
...         "messages": [{"role": "user", "content": prompt}],
...         "temperature": 0.7,
...         "max_tokens": 1024
...     }
...     try:
...         response = requests.post(LM_STUDIO_URL, json=payload)
...         return response.json()['choices'][0]['message']['content']
...     except Exception as e:

```

ЕКРАН ГРАФІЧНОГО ІНТЕРФЕЙСУ OPEN WEBUI



ПРИКЛАД ВИКОРИСТАННЯ СИСТЕМИ ДЛЯ РЕЗЮМУВАННЯ СТАТТІ

Запит


AI Automation in University Education
Новинка! Поділі

Джерела

+ Додати
Знайти

Вибрати всі джерела ☒

PDF

Тези_Маняк. Корчовський.pdf ☒

Чат



AI Automation in University Education

1 джерело

Наданий текст є науковою статтею під назвою "АНАЛІЗ МЕТОДІВ АВТОМАТИЗАЦІЇ НАВЧАЛЬНОГО ПРОЦЕСУ З ВИКОРИСТАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ", написаною дослідниками з Вінницького

Надай резюме до статті
1 джерело

Відповідь

Чат

Оновити

Наданий текст є науковою статтею під назвою "АНАЛІЗ МЕТОДІВ АВТОМАТИЗАЦІЇ НАВЧАЛЬНОГО ПРОЦЕСУ З ВИКОРИСТАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ", написаною дослідниками з Вінницького національного технічного університету та Європейського коледжу. Документ зосереджується на вивченні сучасних методів автоматизації освітнього процесу в університетах за допомогою штучного інтелекту (ШІ). Автори аналізують переваги та недоліки різних підходів до впровадження ШІ, а також його вплив на продуктивність викладачів і студентів. У роботі

↓

Перейти вниз

побудови інтелектуальної комп'ютерної системи для автоматизації

Почніть вводити текст...
1 джерело

▶

Які основні етапи університетського навчально

▶