

Методичні рекомендації до виконання практичної роботи №3

"Створення додатків для генерації тексту"

Мета роботи

- Ознайомитися з бібліотекою OpenAI та її основними концепціями.
- Навчитися створювати додатки для генерації тексту за допомогою OpenAI.
- Розглянути використання параметрів запитів, таких як температура та токени.
- Навчитися працювати з GitHub Models та Azure AI Inference SDK.

Обладнання

- Комп'ютер з доступом до інтернету
- Встановлений Python 3.x
- GitHub аккаунт
- Visual Studio Code або Jupyter Notebook
- API-ключ для OpenAI або GitHub Models

Порядок виконання роботи

1. Підготовка середовища

- Зареєструйтеся на GitHub та отримайте доступ до GitHub Models.
- Встановіть Python та необхідні бібліотеки (`pip install openai`).
- Отримайте API-ключ OpenAI або GitHub Models та збережіть його у `.env`.

2. Виконання базових вправ

- Ознайомтеся з базовим прикладом коду в `githubmodels-assignment_ua.py`.
- Використовуйте `ChatCompletionsClient` для генерації тексту.
- Виконайте тестування з різними параметрами (температура, кількість токенів).

3. Індивідуальне завдання

Виконайте одне з запропонованих завдань:

1. Розробіть додаток для генерації рецептів за списком інгредієнтів.
2. Створіть чат-бота, що відповідає на запити щодо Python.
3. Розробіть історичного бота, що відтворює манеру мовлення історичних постатей.

Вимоги до звіту

1. Титульна сторінка
2. Мета роботи
3. Хід виконання:
 - Опис налаштування середовища
 - Лістинг створених промптів
 - Аналіз результатів генерації тексту
 - Виконане індивідуальне завдання
4. Висновки
5. Завантаження роботи у Moodle

Контрольні запитання

1. Що таке температура в контексті генерації тексту?
2. Як впливає параметр `max_tokens` на генерацію тексту?
3. Які переваги використання GitHub Models?
4. Як API OpenAI відрізняється від GitHub Models?
5. Які бібліотеки використовуються для роботи з OpenAI?

Варіанти індивідуальних завдань

1. Розробіть додаток для генерації та аналізу алгоритмів сортування (bubble, quick, merge sort), включаючи оцінку складності алгоритмів.
2. Створіть додаток для автоматичної оптимізації SQL-запитів за допомогою AI, враховуючи системний контекст DBA-експерта.
3. Розробіть додаток для автоматичного створення API-специфікацій системи моніторингу серверного обладнання.
4. Створіть додаток для пояснення роботи конвеєра процесора за допомогою AI-моделі та chain-of-thought підходу.
5. Розробіть додаток для аналізу та перевірки конфігурацій мережевого обладнання на предмет безпеки.
6. Створіть додаток для автоматичної генерації тестів периферійних пристроїв, включаючи порівняння форматів.
7. Розробіть додаток для оптимізації асемблерного коду, використовуючи AI та few-shot приклади.
8. Створіть додаток для автоматичної генерації документації драйвера USB-пристрою з різними стилями оформлення.
9. Розробіть додаток для аналізу дамів пам'яті x86/ARM процесорів та виявлення потенційних вразливостей.
10. Створіть додаток для генерації схем на Verilog/VHDL з аналізом та порівнянням стилів написання коду.
11. Розробіть додаток для аналізу та оптимізації мережевих протоколів IoT-пристроїв за допомогою AI.

12. Створіть додаток для автоматизованого генерування тестів проникнення для вбудованих систем.
13. Розробіть додаток для аналізу, рефакторингу та оптимізації коду FPGA-проектів.
14. Створіть додаток для діагностики та налагодження RTOS-систем на основі аналізу логів та даних.
15. Розробіть додаток для оптимізації енергоспоживання вбудованих систем на основі AI-моделювання.
16. Створіть додаток для аналізу та покращення надійності кеш-пам'яті процесорів.
17. Розробіть додаток для автоматизованої генерації драйверів GPIO з підтримкою різних інтерфейсів.
18. Створіть додаток для оптимізації паралельних обчислень на CUDA за допомогою AI.
19. Розробіть додаток для аналізу та покращення якості обслуговування (QoS) в реальному часі.
20. Створіть додаток для автоматичної генерації специфікацій SoC із використанням IP-блоків.

Для кожного варіанту необхідно:

- Розробити базовий додаток для генерації тексту.
- Додати можливість налаштування параметрів генерації (температура, токени).
- Провести тестування роботи додатку на різних варіантах вхідних даних.
- Оцінити якість отриманих результатів.
- Запропонувати покращення у структурі запитів.