

Інтерактивна апаратно-програмна система інтелектуального асистента на базі Arduino та виділеного серверу

Стан проблеми. Сучасні великі мовні моделі (LLM) все глибше інтегруються в цифрові помічники, дозволяючи вести діалог з користувачем на природному рівні. Однак більшість таких систем існують виключно як програмні продукти — веб-сайти або мобільні застосунки, що позбавляє взаємодію фізичного втілення та невербальних елементів спілкування [4].

Водночас, завдяки розвитку Інтернету речей (IoT) та доступності таких платформ, як ESP32, з'явилася можливість створювати апаратні пристрої, що виходять за межі простої обробки команд. Такі системи здатні надавати емоційний відгук через світлові ефекти, звук чи анімацію, роблячи взаємодію з технологією більш інтуїтивною та "живою" [2, 5]. Таким чином, актуальною є задача поєднання потужності хмарних мовних моделей із фізичним інтерфейсом, здатним демонструвати емоційні реакції.

Постановка задачі. Метою роботи є створення інтерактивного інтелектуального асистента, який використовує апаратні засоби для відображення емоцій та здійснює обробку користувацьких запитів через віддалений обчислювальний модуль. Для цього система повинна забезпечувати повний цикл взаємодії: від введення запитів голосом чи текстом, зокрема з мобільного пристрою, до їх аналізу та формування відповіді на основі мовної моделі. Сформовані результати мають передаватися на апаратну частину для відображення тексту, звуків чи зображень, а також для візуалізації емоційного стану асистента через світлову індикацію, анімації або відповідні звукові ефекти.

Розв'язання задачі. Архітектура системи є дворівневою. Серверна частина (VPS) відповідає за обчислювально складні завдання: приймає запит від апаратного модуля, обробляє його за допомогою мовної моделі (наприклад, через Google's Gemini AP), визначає емоційний тон відповіді та відправляє назад структурований JSON-об'єкт.

Апаратна частина на базі мікроконтролера ESP32 [1] виступає як фізичний інтерфейс. Вона приймає голосові команди, надсилає їх на сервер для обробки, отримує відповідь і візуалізує її. Програмна частина розроблена в середовищі Arduino IDE з використанням бібліотек `WiFi.h`, `ArduinoJson.h` та `ESPAsyncWebServer.h` для організації стабільної комунікації.

Таблиця 1.

Основні компоненти системи та їх функції

Компонент	Призначення	Особливості
ESP32	Центральний модуль керування	Вбудований Wi-Fi, обробка HTTP-запитів, керування периферією [1]
Мікрофонний модуль	Ввід голосових команд	Дозволяє користувачу ставити питання голосом
DFPlayer Mini	Відтворення звукових ефектів	Озвучення емоційних реакцій, системні сповіщення
RGB-світлодіоди	Візуальна індикація емоцій	Динамічна зміна кольору відповідно до контексту відповіді
OLED-дисплей	Виведення тексту та анімацій	Відображення відповідей та стилізованого "обличчя" асистента
Віддалений сервер (VPS)	Обчислювальний модуль	Інтеграція з LLM API, аналіз тональності, REST API бекенд
Мобільний інтерфейс	Допоміжний канал вводу	Проста веб-сторінка для надсилання текстових запитів з телефону

Інноваційність проєкту полягає у реалізації контекстної емоційної реакції, яка забезпечує якісний зворотний зв'язок з користувачем [4] на основі принципів афективних обчислень [2, 5].

Висновки. У ході роботи було розроблено концепцію та архітектуру апаратно-програмного асистента, що поєднує обчислювальні можливості хмарних мовних моделей та інтерактивність фізичних IoT-пристроїв. Таке рішення дозволяє створити більш природний та емоційно забарвлений досвід взаємодії людини з ШІ.

Подальший розвиток проєкту може включати реалізацію локального синтезу мовлення (Text-to-Speech) на пристрої, вдосконалення алгоритмів розпізнавання емоцій.

Література

- [1] Espressif Systems. (2023). ESP32 Technical Reference Manual.
- [2] Picard, R. W. (1997). Affective Computing. MIT Press.
- [3] Google. (2024). Gemini API Documentation.
- [4] Norman, D. A. (2013). The Design of Everyday Things.
- [5] B. Mutlu, J. Forlizzi. (2008). "Robots in organizations: the role of robot group membership on human-robot collaboration".