



物聯網與邊緣運算

居家老人照護

醫資三 110316108 黃麗文

110316121 黃琪蓁 110316123 吳煜楷



Overview

- 前言
- 物聯網 (IoT) 概述
- 邊緣運算 (Edge Computing) 概述
- 物聯網與邊緣運算在居家老人照護的應用
- 優勢與挑戰
- 方法、目前進度
- 結論





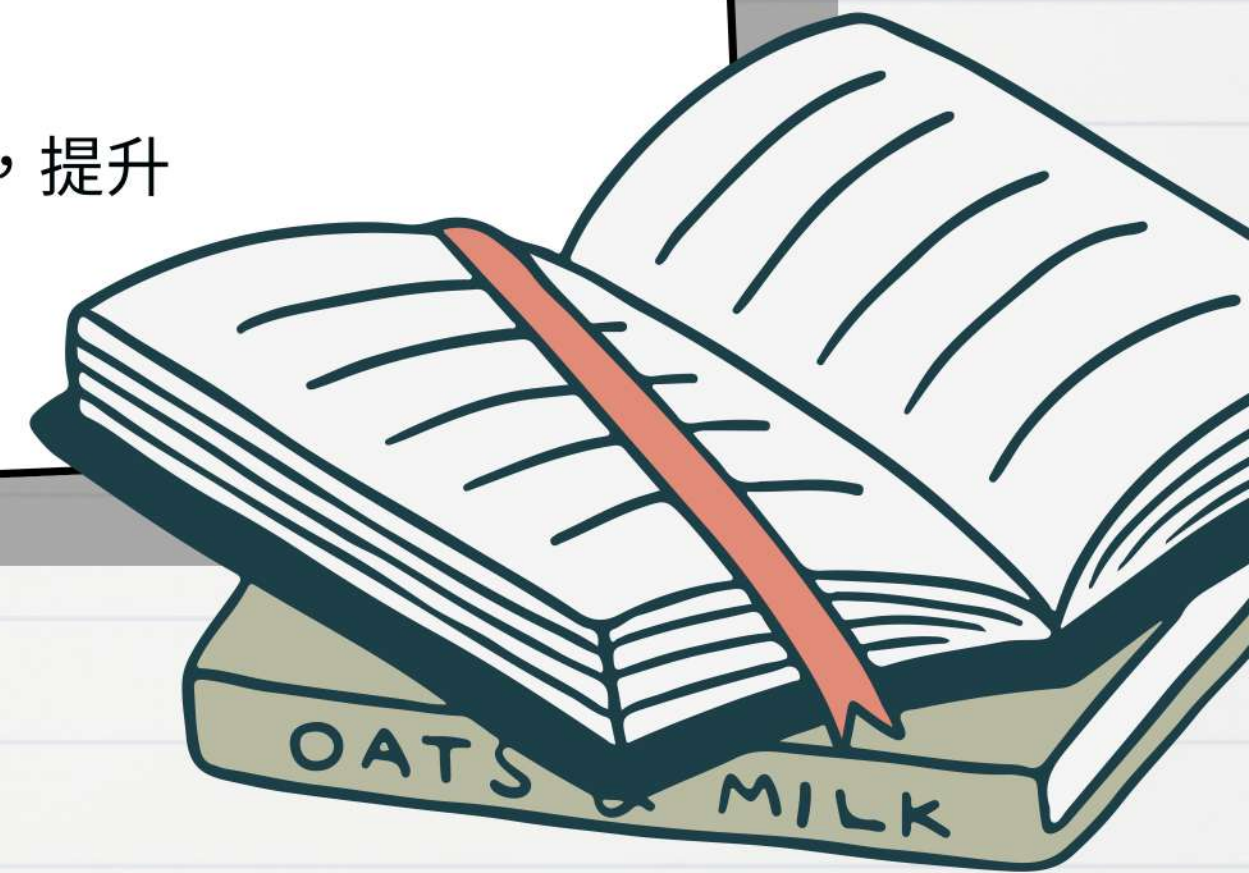
前言

背景

隨著人口老齡化，居家老人照護需求日益增加。

目標

探討物聯網與邊緣運算技術如何改善居家老人照護，提升生活質量與安全性。

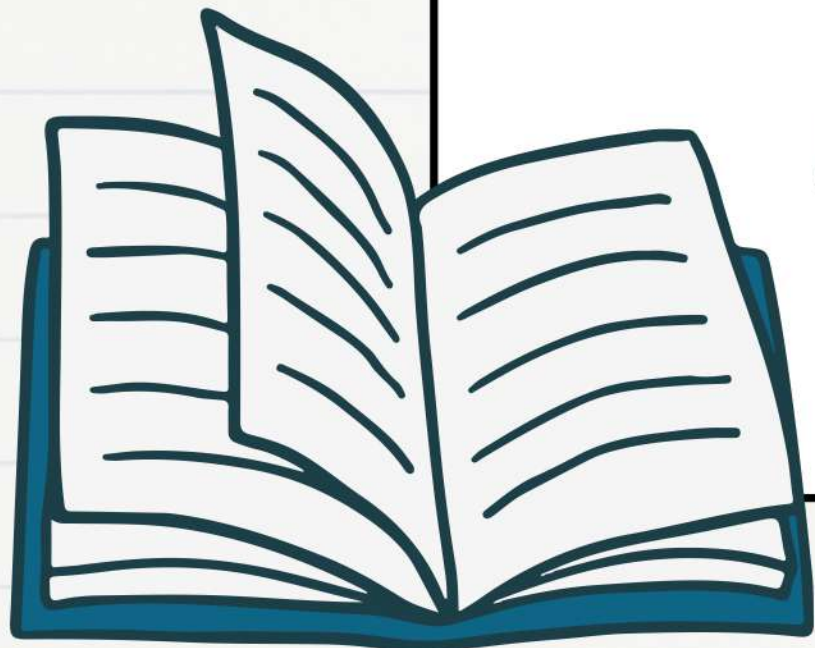


物聯網 (IoT) 概述

定義：物聯網是指通過各種感測設備，將物品與網絡連接，實現智能化識別、定位、追蹤、監控和管理的一種網絡。

1. 感測器
2. 通訊網路
3. 數據處理平台

應用領域：智能家居、智慧城市、醫療健康等。

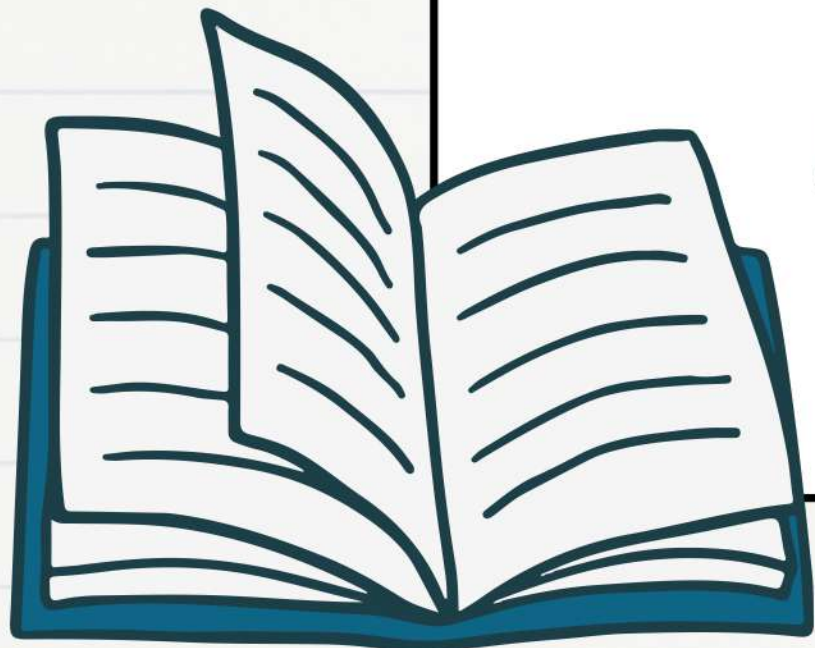


邊緣運算（Edge Computing）概述

定義：邊緣運算是在靠近數據源頭或設備的位置進行數據處理和分析的一種技術，減少數據傳輸延遲，提高處理效率。

1. 邊緣設備（如路由器、閘道器）通訊網路
2. 本地數據處理和分析

應用領域：工業自動化、智能交通、醫療健康等。



物聯網與邊緣運算在居家老人照護的應用

健康監測

- 佩戴式設備監測心率、血壓、血氧等
- 智能床墊監測睡眠質量

安全防護

- 智能門窗感應器
- 跌倒監測與警報系統

日常生活支持

- 智能家居控制
- 自動藥物提醒與分配系統





優勢



實時數據處理與反應



減少網絡延遲和帶寬需求



提高系統可靠性和安全性





挑戰



數據隱私與安全問題



設備兼容性與標準化



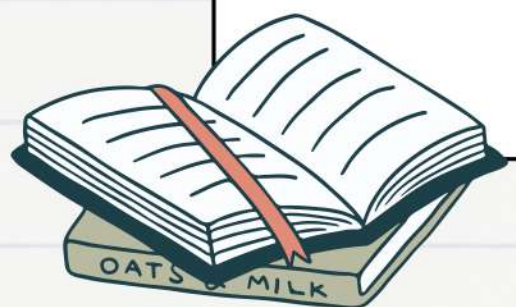
成本與技術門檻



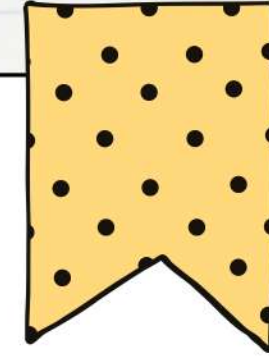


方法

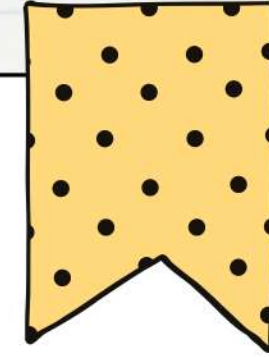
設計並實施一款專門為初期阿茲海默症老人或未患病的老人設計的聊天機器人，以提升其生活質量並幫助記憶功能的維持。該項目將使用OpenAI的GPT模型進行語言處理和生成，並在Google Colab上進行數據分析和程式開發。



所需工具與資源：



- OpenAI GPT模型
- Google Colab
- 阿茲海默症相關的健保資料（可匿名化的醫療記錄、病史等）
- Python及相關數據分析庫
- 自然語言處理相關庫
- Ollama機器學習平台



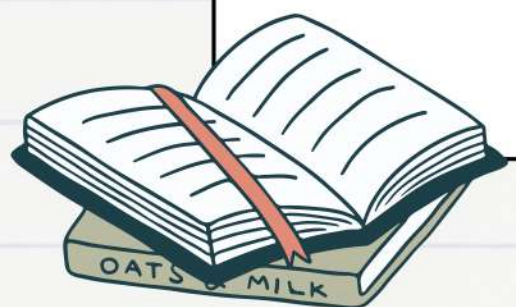
關於健保資料：

1. 收集阿茲海默症患者的匿名化健保數據，包括診斷記錄、治療方案、病史記錄等。
2. 處理缺失值和異常值 + 提取與阿茲海默症相關的關鍵信息，如診斷結果、治療過程等。



聊天機器人設計與模型訓練

1. 對話設計與樣本收集：設計適合初期阿茲海默症老人或未患病老人的對話樣本，包含日常問候、記憶激發問題、健康提醒等。
2. 模型訓練：使用Ollama進行聊天機器人模型的訓練和調整。在Google Colab上編寫訓練腳本，並根據反饋調整模型參數。



目前進度

01

✓
10
秒

[1] !pip install "openai<1"



Collecting openai<1

Downloading openai-0.28.1-py3-none-any.whl (76 kB)

77.0/77.0 kB 1.5 MB/s eta 0:00:00

Requirement already satisfied: requests>=2.20 in /usr/local/lib/python3.10/dist-packages (from openai<1) (2.31.0)

Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from openai<1) (4.66.4)

Requirement already satisfied: aiohttp in /usr/local/lib/python3.10/dist-packages (from openai<1) (3.9.5)

Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests>=2.20->openai<1) (3.3.2)

Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests>=2.20->openai<1) (3.7)

Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests>=2.20->openai<1) (2.0.7)

Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests>=2.20->openai<1) (2024.6.2)

Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp->openai<1) (1.3.1)

Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->openai<1) (23.2.0)

Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp->openai<1) (1.4.1)

Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp->openai<1) (6.0.5)

Requirement already satisfied: yarl<2.0,>=1.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->openai<1) (1.9.4)

Requirement already satisfied: async-timeout<5.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->openai<1) (4.0.3)

Installing collected packages: openai

Successfully installed openai-0.28.1

✓
0
秒

[2] import os

import openai

設置你的 API 密鑰

openai.api_key = "sk-..."



目前進度

02

```
[3] def chat_with_gpt(prompt, model="gpt-3.5-turbo", max_tokens=150):  
    response = openai.ChatCompletion.create(  
        model=model,  
        messages=[  
            {"role": "user", "content": prompt}  
        ],  
        max_tokens=max_tokens  
    )  
    return response['choices'][0]['message']['content']
```

```
▶ user_input = input("你好！請輸入你的問題或對話：")  
while user_input.lower() != 'exit':  
    response = chat_with_gpt(user_input)  
    print("機器人:", response)  
    user_input = input("你：")  
print("再見！")
```

... 你好！請輸入你的問題或對話：



目前進度

03



```
user_input = input("你好！請輸入你的問題或對話：")
while user_input.lower() != 'exit':
    response = chat_with_gpt(user_input)
    print("機器人:", response)
    user_input = input("你：")
print("再見！")
```

...

你好！請輸入你的問題或對話：你好
機器人：你好！有什么可以帮助你吗？
你：請給我5個台灣美食(項目就好)
機器人：1. 小籠包
2. 臭豆腐
3. 台灣牛肉麵
4. 礁溪泡菜鍋
5. 蚵仔煎
你：



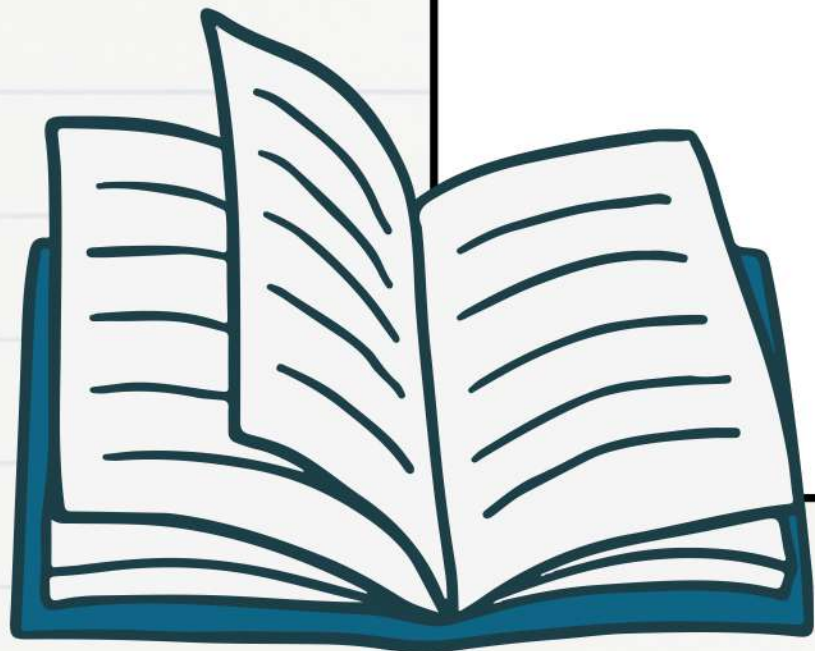
結論

物聯網與邊緣運算在居家老人照護中展現出巨大潛力，

能夠顯著提升老人生活質量與安全性。

隨著技術的不斷進步和應用的普及，未來居家老人照護

將更加智能化和人性化。



Thank You

