

# 北科大 111360205 電子四乙 謝進權

## LLM 以網路爬蟲為 prompt 基準\_測試報告

### 重點說明

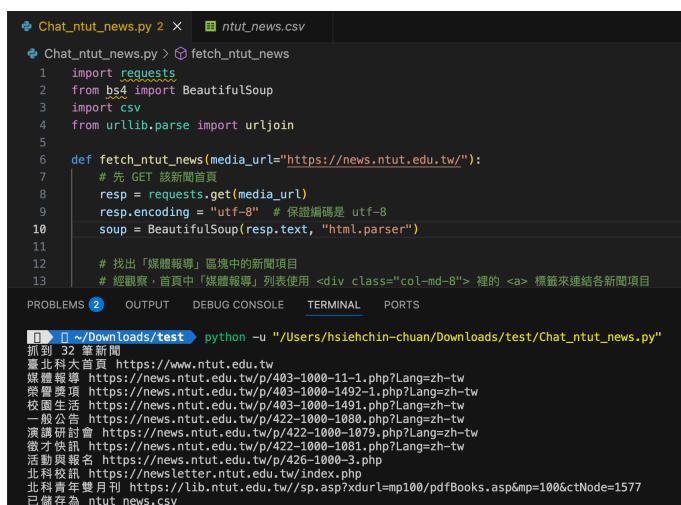
本次作業旨在建立個人基準測試（Benchmarks），主題選定為「Python 網路爬蟲實作與優化」。我測試了兩個主流的大型語言模型（LLM）：**ChatGPT** 和 **Claude**，以評估它們在處理以下專業問題時的表現：

1. **核心實作要求**：撰寫 Python 程式，利用 `requests` 和 `BeautifulSoup` 爬取北科大新聞網頁，並將結果存為 CSV 檔案。
2. **延伸問題 1**：若有反爬蟲機制，該如何處理？
3. **延伸問題 2**：如何使用多執行緒或非同步來加速爬取多個頁面？

我透過實際運程式碼和分析延伸問題的回答深度，比較兩個模型在程式碼可靠性與實戰知識深度方面的優劣。

### 重點截圖

- ChatGPT 提供的程式碼結果：
  - 程式碼成功運行，並且抓到網頁內容。



```
Chat_ntut_news.py 2 x ntut_news.csv
Chat_ntut_news.py > fetch_ntut_news
1 import requests
2 from bs4 import BeautifulSoup
3 import csv
4 from urllib.parse import urljoin
5
6 def fetch_ntut_news(media_url="https://news.ntut.edu.tw/"):
7     # 先 GET 該新聞首頁
8     resp = requests.get(media_url)
9     resp.encoding = "utf-8" # 保證編碼是 utf-8
10    soup = BeautifulSoup(resp.text, "html.parser")
11
12    # 找出「媒體報導」區塊中的新聞項目
13    # 經觀察，首頁中「媒體報導」列表使用 <div class="col-md-8"> 裡的 <a> 標籤來連結各新聞項目
14
15 PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS
python -u "/Users/hsiehchin-chuan/Downloads/test/Chat_ntut_news.py"
抓到 32 篇新聞
臺北科大首頁 https://www.ntut.edu.tw
媒體報導 https://news.ntut.edu.tw/p/403-1000-11-1.php?Lang=zh-tw
榮譽獎項 https://news.ntut.edu.tw/p/403-1000-1492-1.php?Lang=zh-tw
校園生活 https://news.ntut.edu.tw/p/403-1000-1491.php?Lang=zh-tw
一般公告 https://news.ntut.edu.tw/p/422-1000-1080.php?Lang=zh-tw
演講研討會 https://news.ntut.edu.tw/p/422-1000-1079.php?Lang=zh-tw
徵才快訊 https://news.ntut.edu.tw/p/422-1000-1081.php?Lang=zh-tw
活動與報名 https://news.ntut.edu.tw/p/426-1000-3.php
北科校訊 https://newsletter.ntut.edu.tw/index.php
北科青年雙月刊 https://lib.ntut.edu.tw/sp.asp?xdurl=mp100/pdfBooks.asp&mp=100&ctNode=1577
已儲存為 ntut_news.csv
```

- 檢視 .csv 檔案，發覺抓取到錯誤的網頁標籤內容。

```
ntut_news.csv
1 標題,連結
2 臺北科大首頁,https://www.ntut.edu.tw
3 媒體報導,https://news.ntut.edu.tw/p/403-1000-11-1.php?Lang=zh-tw
4 榮譽獎項,https://news.ntut.edu.tw/p/403-1000-1492-1.php?Lang=zh-tw
5 校園生活,https://news.ntut.edu.tw/p/403-1000-1491.php?Lang=zh-tw
6 一般公告,https://news.ntut.edu.tw/p/422-1000-1080.php?Lang=zh-tw
7 演講研討會,https://news.ntut.edu.tw/p/422-1000-1079.php?Lang=zh-tw
8 徵才快訊,https://news.ntut.edu.tw/p/422-1000-1081.php?Lang=zh-tw
9 活動與報名,https://news.ntut.edu.tw/p/426-1000-3.php
10 北科校訊,https://newsletter.ntut.edu.tw/index.php
11 北科青年雙月刊,https://lib.ntut.edu.tw//sp.asp?xdurl=mp100/pdfBooks.asp&mp=100&ctNode=1577
12 Taipei Tech Post,https://www-en.ntut.edu.tw/p/404-1006-37298.php?Lang=en
13 在學學生,https://www.ntut.edu.tw/p/404-1007-85223.php
```

- Claude 提供的程式碼與測試結果：

- 網址直接錯誤，無法抓取網頁資料。

```
Claude_ntut_news.py 2 x
Claude_ntut_news.py > scrape_ntut_news
6 def scrape_ntut_news():
7     """
8     爬取北科大網頁新聞，抓取標題和連結並存為 CSV 檔案
9     """
10    # 北科大新聞網址
11    url = "https://www.ntut.edu.tw/p/403-1000-1366-1.php"
12    #url = "https://news.ntut.edu.tw/"
13
14    # 設定 headers 模擬瀏覽器
15    headers = {
16        'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML
17    }
18
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS
~/Downloads/test python -u "/Users/hsiehchin-chuan/Downloads/test/Claude_ntut_news.py"
連接失敗，狀態碼：404
```

- 修正網址內容，還是無法抓到新聞標題。

```
Claude_ntut_news.py > scrape_ntut_news > url
6 def scrape_ntut_news():
7     """
8     爬取北科大網頁新聞，抓取標題和連結並存為 CSV 檔案
9     """
10    # 北科大新聞網址
11    #url = "https://www.ntut.edu.tw/p/403-1000-1366-1.php"
12    url = "https://news.ntut.edu.tw/"
13
14    # 設定 headers 模擬瀏覽器
15    headers = {
16        'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML
17    }
18
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS
~/Downloads/test python -u "/Users/hsiehchin-chuan/Downloads/test/Claude_ntut_news.py"
成功連接到北科大新聞網頁！
成功爬取 0 則新聞！
結果已儲存至 ntut_news.csv
前 5 則新聞預覽：
```

- 過程截圖：

- ChatGPT

ChatGPT

請用 Python 的 requests 和 BeautifulSoup 函式庫，撰寫一個程式來爬取某個靜態網頁（例如某大學的最新新聞頁面）。程式需要抓取標題和連結，並將結果存為一個 CSV 檔案。

好的！我幫你寫一個範例程式，使用 requests 和 BeautifulSoup 來爬取某大學的最新新聞頁面（這裡我用假設的靜態網頁 URL，你可以替換成你要的網站），抓取「新聞標題」和「連結」，最後存成 news.csv。

以下是完整程式：

```
python
import requests
from bs4 import BeautifulSoup
import csv
```

↓

↑ 目錄 (18) (請點選你要的新聞頁面)

○ Claude



模型回答的看法與比較

1. 核心程式碼的正確性（可靠性）

模型	程式碼運行結果	程式碼評估	實務影響
ChatGPT	運行成功，但內容錯誤。抓取到的是網頁分頁區塊的標題和連結，而非單一新聞項目。	程式碼能夠執行，但核心的HTML 選擇器不夠精準，未能準確定位到目標新聞列表。	輸出結果錯誤，仍需人工介入修正，實用性低。
Claude	運行失敗，爬取數量為 0。	程式架構完整，但 HTML 選擇器有誤，無法定位到任何內容。	實用性最低，因為程式無法產生任何有效輸出。

修正結論： 兩者皆未通過「程式碼正確性」的考驗。ChatGPT 犯了「抓取錯誤內容」的錯誤，而 Claude 犯了「抓取不到內容」的錯誤。兩者在處理動態且複雜的網頁結構\*\*時，其 HTML 選擇器定位能力都有待加強。

2. 實戰知識與解方深度（延伸問題）

延伸問題	ChatGPT 表現	Claude 表現	比較點
反爬蟲機制	提供了 5 點常見解法（如 User-Agent、延遲、代理）與實用檢查流程。	提供了 更詳盡的 6 點主要對策，包含「隨機 User-Agent 輪換」、「Session 管理」和「智慧重試機制」。	Claude 勝。 知識深度更高，涵蓋了更全面的實戰技巧。
效率優化（並發）	提供了 ThreadPooExecutor 和 asyncio + aiohttp 的可執行範例程式碼與優缺點比較。	提供了兩種方法的詳細優缺點分析，並以表格化方式呈現預估耗時、資源消耗和複雜度的對比。	平手。 兩者知識皆正確，Claude 的呈現結構更佳，ChatGPT 的程式碼更為直接。
介面/開發體驗	標準文字輸出。	提供了程式碼編輯介面（Canva），可即時修改，提升了除錯和嘗試的便利性。	Claude 勝。 介面設計更貼合程式開發者的需求。

結論

我最終比較喜歡 Claude。

選擇 Claude 的原因（深度與體驗優先）

雖然 Claude 在核心實作題上犯了最嚴重的「抓取不到內容」的錯誤，但這次基準測試的目的是可延伸詢問與分辨好壞，而 Claude 在這兩方面的表現更為出色：

- 1. 知識深度： 在延伸問題（反爬蟲、效率優化）中，Claude 展現了更豐富、更工程化的專業知識，例如隨機 User-Agent 和 Session 管理，其知識廣度和專業性超越了 ChatGPT。
- 2. 開發體驗： Claude 提供了程式碼編輯介面，雖然程式碼有誤，但這個介面讓我可以即時進行嘗試和修正。這對於學習和快速原型開發來說，是一個非常友善且有用的功能。
- 3. 錯誤性質： 雖然兩者程式碼都錯誤，但 Claude 的輸出系統性、專業性強。我認為它的回答更適合作為技術報告或知識庫的參考，而 ChatGPT 雖然運行了，但抓取了錯誤的內容，其程式碼的「欺騙性」反而更高。

總結： Claude 憑藉其更深的專業知識、更優異的報告結構以及對開發友善的介面，使其在「知識廣度」和「實戰經驗傳授」方面表現更為突出，因此我更傾向於將其視為一個高階的技術諮詢工具。