聯合新聞網新聞標題爬蟲與文字雲生成

一.前置作業

1. 預先安裝:

```
pip install selenium pandas wordcloud jieba
```

檔案準備

1. 下載停用詞檔案
2. 將 `stopwords.txt` 存放在專案資料夾
3. 字型檔案(本文件提供Noto Sans Traditional Chinese 連結:{https://fonts.google.com/noto/specimen/Noto+Sans+TC})

二.程式說明

1. 模組導入

```
# 套件
import time
import random
import requests
import pandas as pd
from PIL import Image
import numpy as np
from wordcloud import WordCloud
import matplotlib.pyplot as plt
import jieba
import cv2
```

2.Header 設置

```
HEADERS = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko
}
```

3.定義爬蟲方式

```python
def get_news_list(page_num=20):
    """爬取新聞標題"""
    base_url = "https://udn.com/api/more"
    news_titles = []

    for page in range(1, page_num + 1):
        query = f"page={page}&channelId=1&cate_id=0&type=breaknews"
        news_list_url = f"{base_url}?{query}"

        try:
            r = requests.get(news_list_url, headers=HEADERS)
            r.raise_for_status()
            news_data = r.json()

            news_titles.extend([
                {
                    "標題": news.get('title', '標題未知'),
                    "連結": f"https://udn.com{news.get('url', '#')}",
                    "時間": news.get('time', '時間未知'),
                }
                for news in news_data['lists']
            ])

            print(f"✅ 已抓取第 {page} 頁，累計 {len(news_titles)} 篇新聞")
            time.sleep(random.uniform(1, 2))

        except Exception as e:
            print(f"❌ 第 {page} 頁抓取失敗：{str(e)}")
            continue

    return news_titles
```

4.定義文字雲生成

```python
def generate_wordcloud(csv_path, font_path, stopwords_path):
    """生成文字雲"""
    try:
        # 讀取數據
        df = pd.read_csv(csv_path)
        text = " ".join(df['標題'].dropna())

        # 載入停用詞
        with open(stopwords_path, 'r', encoding='utf-8') as f:
            stopwords = set(f.read().splitlines())

        # 中文分詞
        segmented_text = " ".join([
            word for word in jieba.cut(text)
            if word not in stopwords and len(word) > 1
        ])

        # 生成文字雲
        wordcloud = WordCloud(
            font_path=font_path,
            width=1600,
            height=1200,
            background_color='white',
            max_words=300,
            collocations=False
        ).generate(segmented_text)

        # 顯示與保存
        plt.figure(figsize=(20, 15))
        plt.imshow(wordcloud, interpolation='bilinear')
        plt.axis("off")
        plt.show()
        wordcloud.to_file("news_wordcloud.png")
        print("✅ 文字雲已生成並保存為 news_wordcloud.png")

    except Exception as e:
        print(f"❌ 文字雲生成失敗: {str(e)}")
```

5.執行

```python
if __name__ == "__main__":
    # 爬取新聞（範例抓取3頁）
    news_data = get_news_list(page_num=3)

    # 保存CSV
    csv_filename = "udn_news.csv"
    pd.DataFrame(news_data).to_csv(csv_filename, index=False, encoding='utf-8-sig')
    print(f"✅ 數據已保存至 {csv_filename}")

    # 生成文字雲（需替換以下路徑）
    generate_wordcloud(
        csv_path=csv_filename,
        font = r"./NotoSansTC-Bold.ttf",          # 也可替換為您的字型路徑
        stopwords_path=r"./stopwords.txt"          # 也可替換為您的停用詞文件路徑
    )
```

預期輸出

- `udn_news.csv`：新聞數據檔案
- `news_wordcloud.png`：文字雲圖片