Research Paper Fetching - Project Report

1. Introduction

This task implements a Python-based solution to fetch research papers from PubMed API, identify papers affiliated with pharmaceutical or biotech companies, and output the results in a structured CSV format. The implementation follows the assignment guidelines by ensuring the modularity, efficiency, and robustness.

2. Approach

The Python program is designed as a command-line tool that interacts with the **PubMed API**. It consists of two components in the api_fetcher.py file

1. **Module** – Handles fetching, processing, and formatting research papers.
2. **CLI script** – Allows users to query PubMed via the command line instead of hardcoding search terms.

Key Features:

- **Command-Line Query Input:** The script accepts a search query as a required argument instead of using a fixed term like `"cancer treatment"`.
- **Optional Parameters:**
  - `-f` or `--file` → Saves results to a CSV file instead of printing them.
  - `-d` or `--debug` → Prints additional debug information during execution.
- **Help Option (`-h` or `--help`)**: Displays usage instructions when no query is provided.

Design Considerations:

- **Modular Design:** Separates concerns between fetching data (`research_paper_fetcher.py`) and handling CLI input/output (`get_papers_list.py`).
- **Error Handling:** Manages API failures, invalid queries, missing author information, and network errors gracefully.
- **Scalability:** Supports **PubMed's full query syntax** for advanced searches, making it flexible for different research needs.

3. Methodology

3.1 Data Collection

1. **PubMed API Search:**
    - A query is passed to the PubMed API to retrieve paper IDs.
    - Results are limited by a configurable `max_results` parameter.
2. **Fetching Paper Details:**
    - Paper metadata is retrieved using the PubMed `efetch` API.
    - Extracts key information: **PubmedID, Title, Publication Date, Authors, Affiliations.**

3.2 Data Processing

- **Author Affiliation Analysis:**
    - Extracts affiliations from author metadata.
    - Identifies pharmaceutical or biotech companies based on keywords (`pharma`, `biotech`).
    - Separates **academic** vs. **non-academic** authors.
    - Identifies corresponding author emails (if available).

3.3 Output Formatting

- **CSV Output:**
    - Fields: `PubmedID, Title, Publication Date, Non-academic Author(s), Company Affiliation(s), Corresponding Author Email.`
    - If no filename is provided, results are printed to the console.

3.4 Command-line Interface

- **User Input Handling:**
    - `query` (required): Specifies the search term.
    - `-f, --file`: Saves results to a CSV file.
    - `-d, --debug`: Enables debug mode for additional logging.

4. Results

The program successfully extracts and processes research papers based on given queries. Example statistics:

- **Query:** “cancer treatment”
- **Total Papers Retrieved:** 10
- **Papers with Pharma/Biotech Affiliations:** 2
- **Papers without Affiliations:** 8
- **Results saved to:** `output.csv`

```
C:\Users\meghana\Downloads>python api_fetcher.py "cancer treatment"
Fetching papers for query: cancer treatment
Results will be printed to the console.
Results saved to output.csv
Pharma/Biotech Papers: 2
Other Papers: 8

C:\Users\meghana\Downloads>python api_fetcher.py -h
usage: api_fetcher.py [-h] [-f FILE] [-d] [query]

Fetch research papers from PubMed.

positional arguments:
  query                 Search query for PubMed.

options:
  -h, --help            show this help message and exit
  -f FILE, --file FILE  Specify filename to save results.
  -d, --debug           Print debug information.
```

Output.csv

	A	B	C	D	E	F	G	H	I	J
1	PubmedID	Title	Publicatior	Non-acade	Company /	Correspon	Category			
2	{'@Versior	{'i': 'Canna	2025	Suttithums	Prince of S	N/A	Pharma/Biotech			
3	{'@Versior	Demograp	2025	N/A	N/A	N/A	Other			
4	{'@Versior	Mediators	2025	N/A	N/A	N/A	Other			
5	{'@Versior	Paradoxica	2025	N/A	N/A	N/A	Other			
6	{'@Versior	Integrated	2025	N/A	N/A	N/A	Other			
7	{'@Versior	Cisplatin P	2025	N/A	N/A	N/A	Other			
8	{'@Versior	Prostate d	2025	N/A	N/A	N/A	Other			
9	{'@Versior	Ratifying t	2025	Tey	Pharmacy	N/A	Pharma/Biotech			
10	{'@Versior	Laparosco	2025	N/A	N/A	N/A	Other			
11	{'@Versior	Role of pu	2025	N/A	N/A	N/A	Other			
12										
13										
14										

5. Conclusion

This project demonstrates an efficient method for identifying industry-affiliated research papers. Future improvements could include:

- **Enhanced affiliation matching** using NLP techniques.
- **Parallel processing** for large-scale queries.
- **Integration with a database** for historical data storage.

This approach ensures a structured and automated method for identifying key industry-backed research from PubMed.