

REPORT

#semanticClimate: THREE-MONTH INTERNSHIP REPORT

Submitted by:

MEBIN JOSEPH

Under the Supervision of

Dr. Gitanjali Yadav
NIPGR, New Delhi

Dr. Peter Murray-Rust
University of Cambridge, UK



**National Institute of Plant Genome Research
New Delhi 110067**

TABLE OF CONTENTS

Sl. No.	Title	Page No.
1	Brief Introduction to #Semanticclimate	1
2	Aims and Objectives	1
3	Internship Mode	2
4	Tools and Software used during Internship	2
5	Selection of a Chapter	3
5.1	Selecting Wordlist from Chapter	4
5.2	Generating Dictionary from wordlist	5
5.3	Table of Contents	7
5.4	Generating Graph for Assessment Report 6 (AR6)	8
5.5	Converting Climate Academy PDF into text format	9
5.6	Hyperbook Workflow Diagram	11
	CONCLUSION	13
	REFERENCES	14

LIST OF FIGURES

Fig No.	Title	Page No.
Fig 3.1	Zoom meeting with Dr. Murray-Rust and fellow Interns	2
Fig 5.1.1	Wordlist from Chapter	4
Fig 5.2.1	Colab	5
Fig 5.2.2	Colab contd.	6
Fig 5.2.3	Generated Dictionary html file	6
Fig 5.3.1	test_graph.py script	7
Fig 5.3.2	Generated TOC graph for WG2_Ch_5	8
Fig 5.4.1	Code for generating AR6 graph	9
Fig 5.4.2	Output graph for AR6	9
Fig 5.5.1	Code for generating TXT file from PDF	10
Fig 5.5.2	The output TXT file	10
Fig 5.6.1	Code for generating Hyperbook Workflow Graph	11
Fig 5.6.2	Hyperbook Workflow Diagram	12

1. Brief Introduction to #semanticClimate

#semanticClimate is an initiative between Scientists led by young Indian science students working to transform the massive and complex UN IPCC AR6 climate report—10,000 pages of critical climate science—into structured, accessible, and semantically rich formats like HTML and XML. By extracting key terms, linking them to Wikidata, building AMI-compatible dictionaries, and creating better navigation tools, they aim to unlock valuable knowledge trapped in PDFs. The team develops open-source tools, hosts collaborative events and hackathons, and invites volunteers, developers, researchers, and funders to join their mission of making climate science more usable for researchers, policymakers, and the public.

2. Aims and Objectives

The IPCC prepares major reports on climate change, including Assessment Reports, Special Reports on specific topics, and Methodology Reports that guide greenhouse gas accounting. These materials are usually published in three volumes, each from one of the IPCC's Working Groups, along with a Synthesis Report:

- **Working Group I** looks at the science behind climate change—what's causing it and how it's changing our world.
- **Working Group II** studies how climate change affects people and nature, and how we can adapt.
- **Working Group III** focuses on ways to reduce greenhouse gas emissions and slow climate change.

Each volume includes chapters, a technical summary, and a simplified summary for policymakers.

The internship was conducted from 28th January 2025 to 1st May 2025, during which I contributed to the #semanticClimate project by working on tools and workflows aimed at semantifying climate science data from the IPCC Sixth Assessment Report (AR6). My responsibilities included exploring and testing open-source tools such as pygetpapers, docanalysis, amilib, and PyMuPDF to extract, process, and structure content from scientific PDFs. I participated in alpha testing and debugging sessions, collaborated with team members using Git and GitHub, and worked on generating semantic dictionaries and visualizations using tools like Graphviz. Additionally, I focused on “Chapter 5 – Food, Fibre, and Other Ecosystem Products” from Working Group II, where I analyzed content and created wordlists to support the development of a comprehensive semantic dictionary.

3. Internship Mode

As part of the internship structure, daily coordination meetings were conducted at 13:30 IST (Indian Standard Time) via the Zoom video conferencing platform. The internship meetings were coordinated by Dr. Peter Murray-Rust, Mentor to the #semanticClimate interns and former staff member of the Department of Chemistry at the University of Cambridge. These sessions were attended by Dr. Renu Kumari, the designated intern manager, together with all participating interns. These sessions served as a platform for task updates, technical discussions, alpha testing, troubleshooting, and project guidance. To ensure continuity and accessibility, recordings of each meeting were uploaded to Slack, the team's primary communication channel, for reference by all team members.

Work progress updates and daily reports were shared regularly on the Slack coordination community and documented on GitHub, ensuring transparency, collaboration, and easy access to project developments.

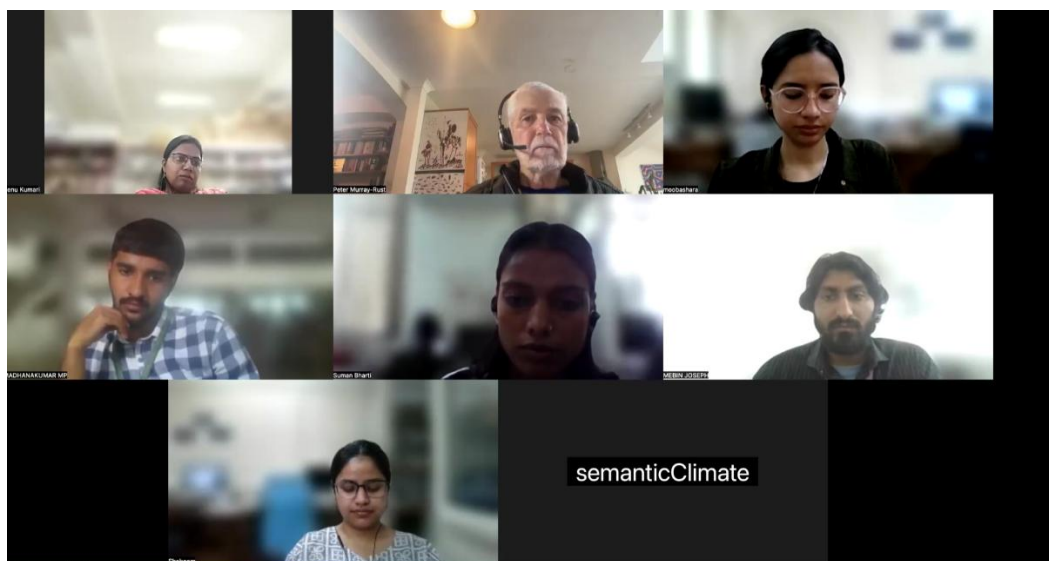


Fig 3.1 Zoom meeting with Dr. Murray-Rust and fellow Interns

4. Tools and Software used during Internship

These tools helped in gathering relevant scientific literature and converting it into usable formats:

- **Python:** A high-level programming language used extensively for scripting and building data-processing tools.
- **amilib:** A Python library that assists in converting PDF documents (like IPCC chapters) into structured HTML for semantic processing.

- **pygetpapers:** A text mining tool used to search and download open-access scientific articles from various repositories.
- **docanalysis:** An open-source command-line tool that enables downloading papers from Europe PMC based on specific queries.
- **PyMuPDF:** to extract and process text and images from PDFs, enabling efficient conversion of PDFs report chapters into structured, semantic-ready .txt file content.
- **Git:** A distributed version control system used to track changes in source code during development.
- **GitHub:** A platform used for hosting and collaborating on code. All project repositories are publicly accessible here.
- **VS Code and PyCharm:** Integrated Development Environments (IDEs) used for coding, debugging, and testing scripts.
- **Google Colab:** A free cloud-based platform used for running Python code with access to GPU/TPU support, ideal for prototyping and experimentation.
- **Slack:** A real-time communication platform used for team coordination and updates.

5. Selection of a Chapter

I selected and worked on **Chapter 5 from Working Group II — titled “Food, Fiber, and Other Ecosystem Products”** — for creating a wordlist to support the development of the semantic dictionary. This chapter highlights the serious threats climate change poses to our food, forests, and oceans — and stresses the need for urgent, inclusive, and science-based solutions to protect ecosystems and livelihoods.

Key points:

Climate change is causing droughts, heatwaves, floods, and pests to increase, which harms crop yields, livestock, and food production.

- Biodiversity loss, wildfires, and reduced forest health are affecting ecosystems and their ability to store carbon.
- Fisheries are also at risk due to ocean warming, acidification, and loss of oxygen, which threaten marine life and coastal communities.

- If climate change continues unchecked, food security will worsen, especially in vulnerable regions.
- Soil degradation, water shortages, and disrupted ecosystems could lead to hunger, migration, and economic loss for people who depend on farming, fishing, and forests.
- The chapter suggests using climate-smart agriculture, agroecology, and sustainable land and water practices to build resilience.
- Technology like early warning systems and better policies can help farmers and fishers adapt. However, progress is slowed by inequality, lack of funding, and weak regulations.

5.1 Selecting Wordlist from Chapter

The PDF of Chapter 5 – Food, Fibre and Other Ecosystem Products was downloaded from the official IPCC website at: <https://www.ipcc.ch/report/ar6/wg2/chapter/chapter-5/>. To support the development of the semantic dictionary, key terms related to climate science were identified through a thorough manual reading of the chapter. These terms were compiled into a .txt file for further processing and semantic enrichment.

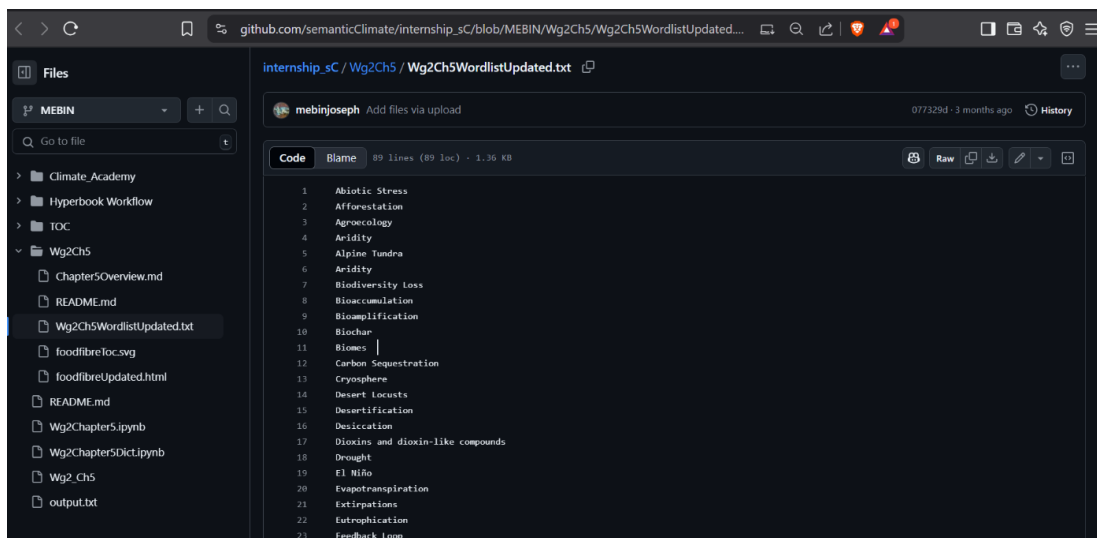


Fig 5.1.1 Wordlist from Chapter

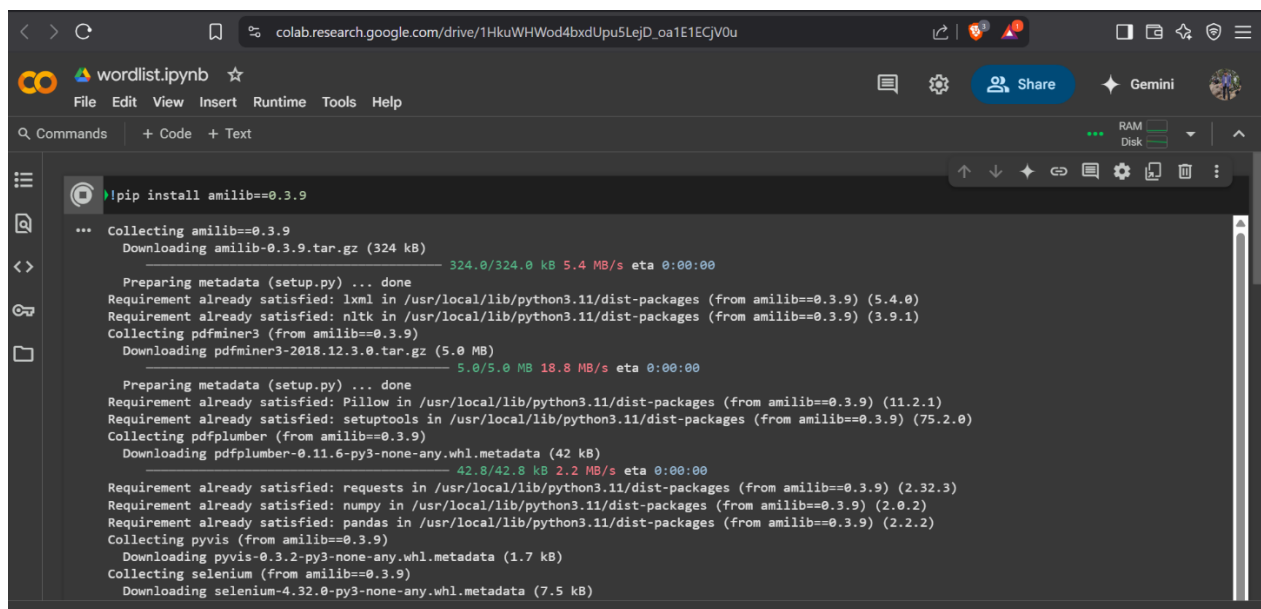
The wordlist file has been uploaded to the GitHub repository and is available at:

https://github.com/semanticClimate/internship_sC/blob/MEBIN/Wg2Ch5/Wg2Ch5WordlistUpdated.txt

5.2 Generating Dictionary from wordlist

The wordlist compiled in the '.txt' file was used to generate a semantic dictionary by fetching definitions and contextual information from Wikipedia. This process was carried out using Google Colab, a cloud-based Jupyter Notebook environment.

The Colab notebook was accessed via a web browser, and the required library 'amilib' (version 0.3.9) was installed using the command: **!pip install amilib==0.3.9**.

A screenshot of a Google Colab notebook interface. The browser address bar shows the URL 'colab.research.google.com/drive/1HkuWHWod4bxdUpu5LejD_oa1E1EGJV0u'. The notebook is titled 'wordlist.ipynb'. The command bar shows '!pip install amilib==0.3.9'. The output of the command is displayed in a dark-themed terminal window, showing the installation progress for 'amilib==0.3.9', including downloading the tar.gz file, preparing metadata, and installing dependencies like lxml, nltk, pdfminer3, pdfplumber, requests, numpy, pandas, pyvis, and selenium. The terminal output includes progress bars and file sizes for each step.

```
!pip install amilib==0.3.9

... Collecting amilib==0.3.9
  Downloading amilib-0.3.9.tar.gz (324 kB)
    324.0/324.0 kB 5.4 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: lxml in /usr/local/lib/python3.11/dist-packages (from amilib==0.3.9) (5.4.0)
Requirement already satisfied: nltk in /usr/local/lib/python3.11/dist-packages (from amilib==0.3.9) (3.9.1)
Collecting pdfminer3 (from amilib==0.3.9)
  Downloading pdfminer3-2018.12.3.0.tar.gz (5.0 MB)
    5.0/5.0 MB 18.8 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: Pillow in /usr/local/lib/python3.11/dist-packages (from amilib==0.3.9) (11.2.1)
Requirement already satisfied: setuptools in /usr/local/lib/python3.11/dist-packages (from amilib==0.3.9) (75.2.0)
Collecting pdfplumber (from amilib==0.3.9)
  Downloading pdfplumber-0.11.6-py3-none-any.whl.metadata (42 kB)
    42.8/42.8 kB 2.2 MB/s eta 0:00:00
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from amilib==0.3.9) (2.32.3)
Requirement already satisfied: numpy in /usr/local/lib/python3.11/dist-packages (from amilib==0.3.9) (2.0.2)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (from amilib==0.3.9) (2.2.2)
Collecting pyvis (from amilib==0.3.9)
  Downloading pyvis-0.3.2-py3-none-any.whl.metadata (1.7 kB)
Collecting selenium (from amilib==0.3.9)
  Downloading selenium-4.32.0-py3-none-any.whl.metadata (7.5 kB)
```

Fig 5.2.1 Colab

The wordlist file ('Wg2Ch5WordlistUpdated.txt') was uploaded to the Colab environment, and the following command was executed to generate the dictionary: **!amilib DICT -words /content/Wg2Ch5WordlistUpdated.txt -description 5ikipedia -dict foodfibreUpdated.html -figures -operation create**.

The screenshot shows a Google Colab notebook named 'wordlist.ipynb'. The left sidebar displays a file explorer with a folder 'sample_data' containing a file 'Wg2Ch5WordlistUpdated.txt'. The main area shows the terminal output of a command: `!amilib DICT --words /content/Wg2Ch5WordlistUpdated.txt --description wikipedia --dict foodfibreUpdated.html`. The output includes package installation logs for 'tinycss' and 'amilib', and a series of debug messages from the 'amilib' library.

Fig 5.2.2 Colab contd.

This command created an HTML-formatted dictionary named 'foodfibreUpdated.html', which included descriptions and associated figures for each term. The resulting dictionary is both human-readable and suitable for further semantic processing.

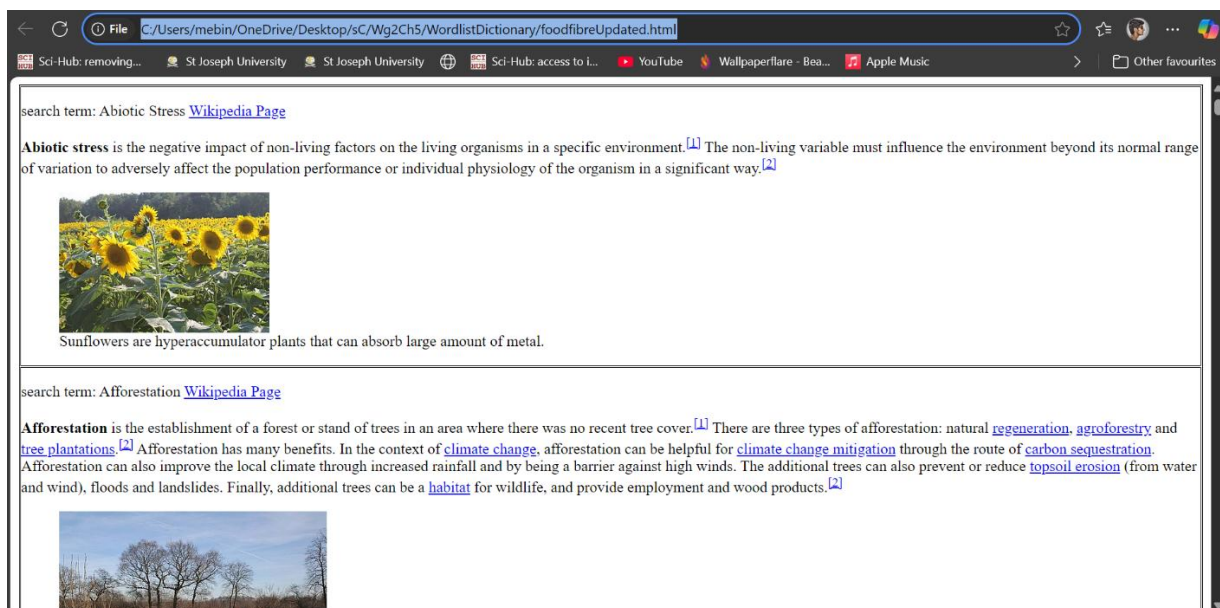


Fig 5.2.3 Generated Dictionary html file

Link to dictionary html file:

https://github.com/semanticClimate/internship_sC/blob/MEBIN/Wg2Ch5/foodfibreUpdated.html

The complete process has been documented and uploaded to GitHub. It is available in the repository at:

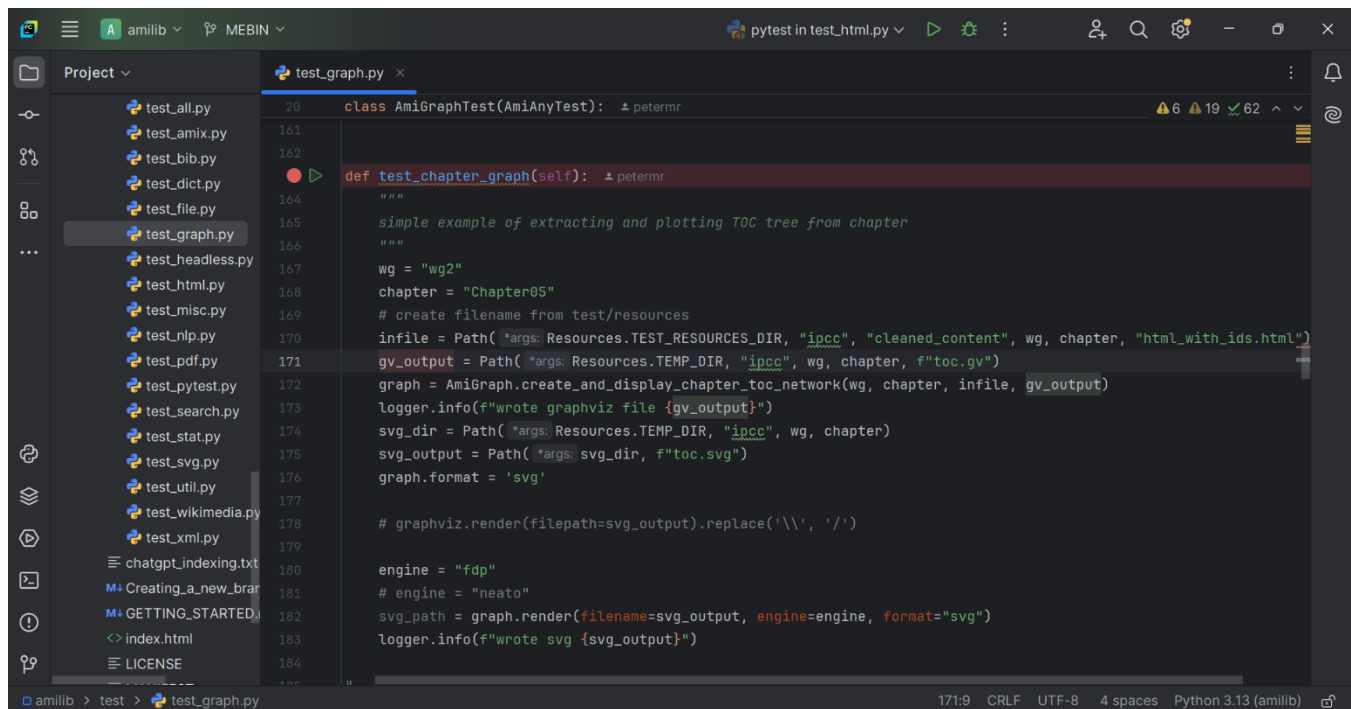
https://github.com/semanticClimate/internship_sC/blob/MEBIN/Wg2Chapter5Dict.ipynb

5.3 Table of Contents

The 'amilib Python library v0.3.9' was cloned onto my local system and set up in both PyCharm and VS Code. To ensure that my contributions did not affect the main branch, I created a dedicated branch named 'MEBIN'. I modified the test_graph.py script to include the details of my selected chapter and working group. Upon running the script, it successfully generated a table of contents (TOC) graph for the chapter in the svg format.

The generated output file has been uploaded to the GitHub repository and is available at:

https://github.com/semanticClimate/internship_sC/blob/MEBIN/Wg2Ch5/foodfibreToc.svg



```
20 class AmiGraphTest(AmiAnyTest):
21     def test_chapter_graph(self):
22         """
23         simple example of extracting and plotting TOC tree from chapter
24         """
25         wg = "wg2"
26         chapter = "Chapter05"
27         # create filename from test/resources
28         infile = Path(*args: Resources.TEST_RESOURCES_DIR, "ipcc", "cleaned_content", wg, chapter, "html_with_ids.html")
29         gv_output = Path(*args: Resources.TEMP_DIR, "ipcc", wg, chapter, f"toc.gv")
30         graph = AmiGraph.create_and_display_chapter_toc_network(wg, chapter, infile, gv_output)
31         logger.info(f"wrote graphviz file {gv_output}")
32         svg_dir = Path(*args: Resources.TEMP_DIR, "ipcc", wg, chapter)
33         svg_output = Path(*args: svg_dir, f"toc.svg")
34         graph.format = 'svg'
35
36         # graphviz.render(filepath=svg_output).replace('\\', '/')
37
38         engine = "fdp"
39         # engine = "neato"
40         svg_path = graph.render(filename=svg_output, engine=engine, format="svg")
41         logger.info(f"wrote svg {svg_output}")
```

Fig 5.3.1 test_graph.py script

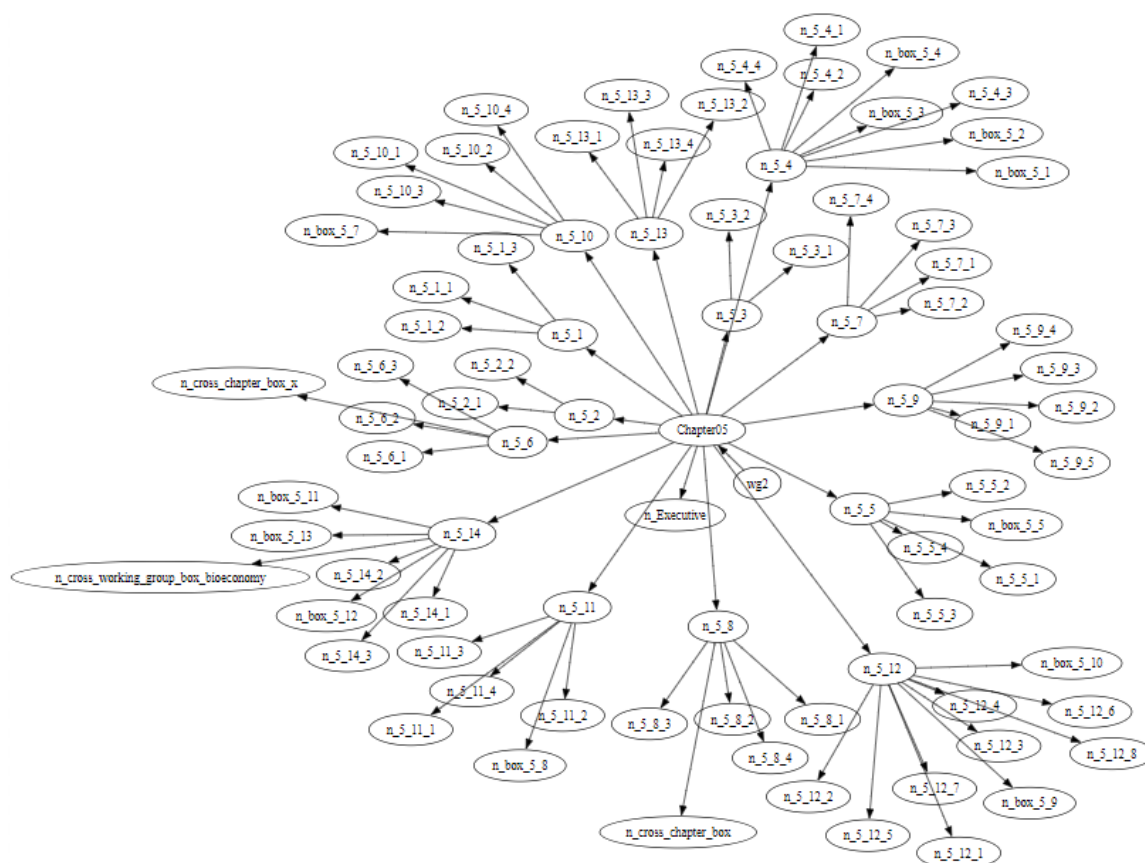


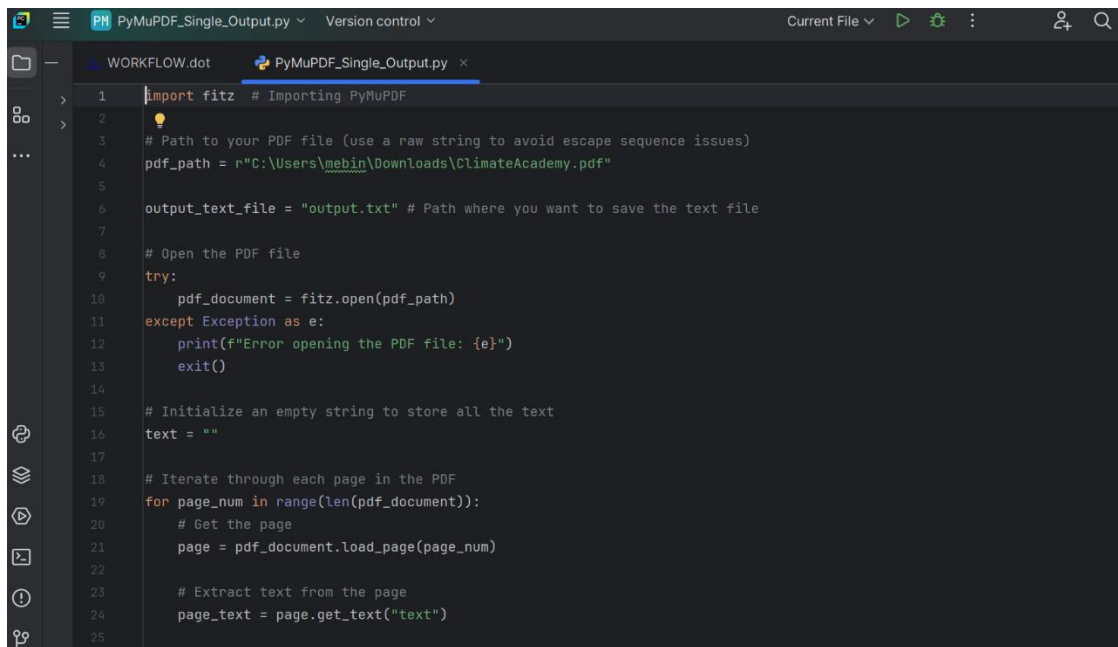
Fig 5.3.2 Generated TOC graph for WG2_Ch_5

5.4 Generating Graph for Assessment Report 6 (AR6)

Using the Graphviz tool integrated with Python in the PyCharm IDE, I created a graph representing the structure of the IPCC AR6, including chapters and cross-chapter linkages across all three Working Groups. The code was written and executed to visualize the interconnections between the chapters. Two versions of the graph were generated — one in horizontal layout and the other in vertical layout, both exported in SVG format.

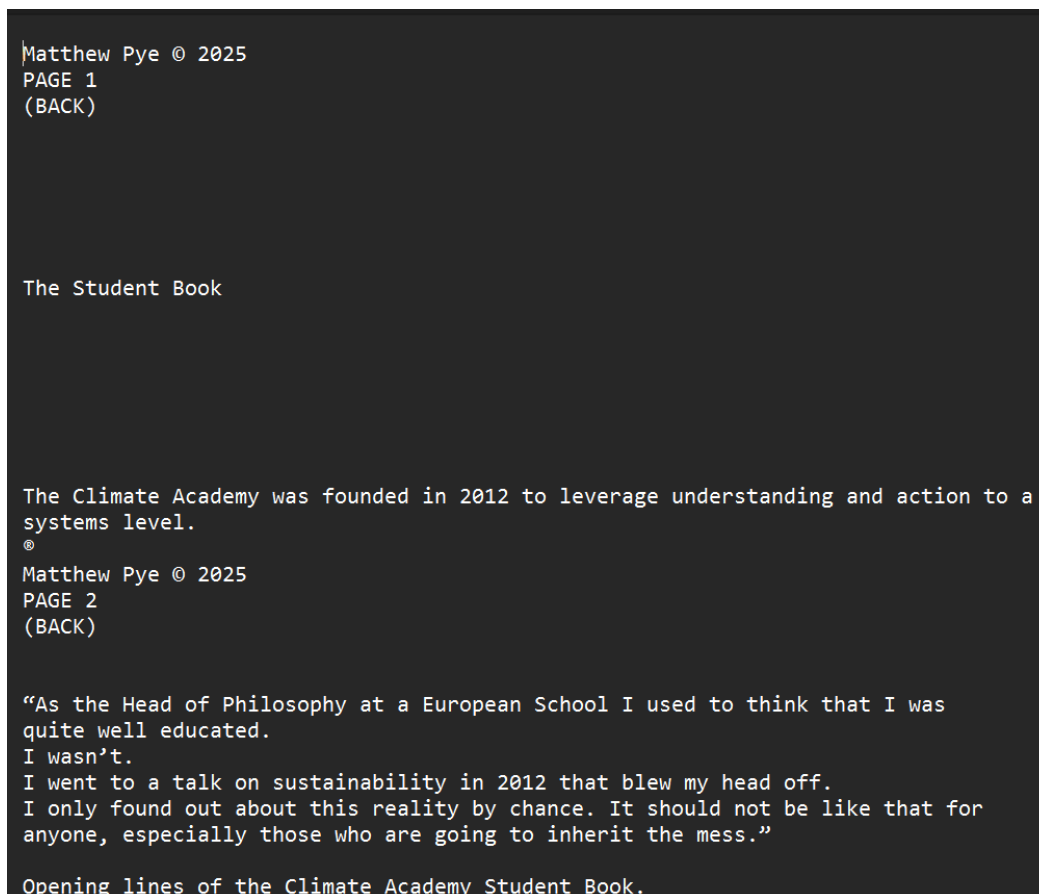
The Python scripts and the resulting graph files have been uploaded to the GitHub repository and are available at:

https://github.com/semanticClimate/internship_sC/tree/MEBIN/TOC

A screenshot of a code editor showing a Python script named 'PyMuPDF_Single_Output.py'. The script uses the PyMuPDF library to open a PDF file, iterate through its pages, and extract text. The code is as follows:

```
1 import fitz # Importing PyMuPDF
2
3 # Path to your PDF file (use a raw string to avoid escape sequence issues)
4 pdf_path = r"C:\Users\mebin\Downloads\ClimateAcademy.pdf"
5
6 output_text_file = "output.txt" # Path where you want to save the text file
7
8 # Open the PDF file
9 try:
10     pdf_document = fitz.open(pdf_path)
11 except Exception as e:
12     print(f"Error opening the PDF file: {e}")
13     exit()
14
15 # Initialize an empty string to store all the text
16 text = ""
17
18 # Iterate through each page in the PDF
19 for page_num in range(len(pdf_document)):
20     # Get the page
21     page = pdf_document.load_page(page_num)
22
23     # Extract text from the page
24     page_text = page.get_text("text")
25
```

Fig 5.5.1 Code for generating TXT file from PDF

A screenshot of a text file containing the following content:

```
Matthew Pye © 2025
PAGE 1
(BACK)

The Student Book

The Climate Academy was founded in 2012 to leverage understanding and action to a
systems level.
©
Matthew Pye © 2025
PAGE 2
(BACK)

"As the Head of Philosophy at a European School I used to think that I was
quite well educated.
I wasn't.
I went to a talk on sustainability in 2012 that blew my head off.
I only found out about this reality by chance. It should not be like that for
anyone, especially those who are going to inherit the mess."

Opening lines of the Climate Academy Student Book.
```

Fig 5.5.2 The output TXT file

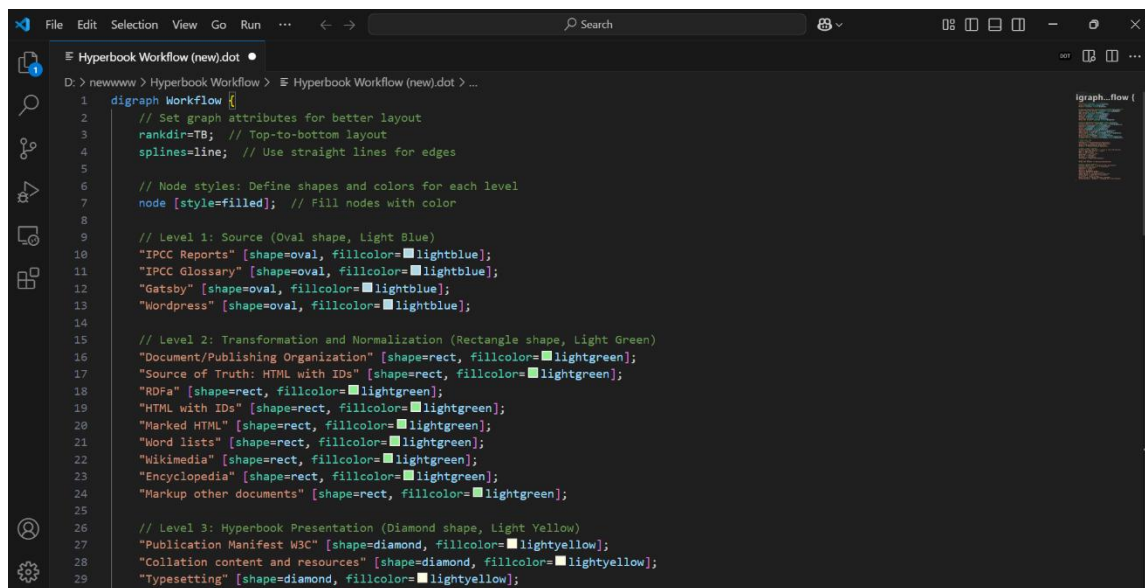
5.6 Hyperbook Workflow Diagram

The Hyperbook Workflow Diagram was created using the Graphviz tool to visually represent the sequential steps involved in transforming the IPCC AR6 report into a semantically structured format. The diagram outlines the complete workflow, starting from the extraction of a word list, followed by semantic enrichment to build a glossary, and concluding with content publishing.

This visualization was developed in VS Code, where the hierarchical structure of the process was defined using Graphviz's DOT language. The resulting diagram illustrates how each component connects and contributes to the overall goal of semantifying climate knowledge.

Both the source `.dot` files and the rendered `.svg` files have been saved and uploaded to the GitHub repository for reference and further development and are available at:

https://github.com/semanticClimate/internship_sC/tree/MEBIN/Hyperbook%20Workflow

A screenshot of the Visual Studio Code editor showing a file named 'Hyperbook Workflow (new).dot'. The code is written in Graphviz DOT language and defines a hierarchical workflow graph. The graph is organized into three levels: Level 1 (Source) with light blue oval nodes, Level 2 (Transformation and Normalization) with light green rectangle nodes, and Level 3 (Hyperbook Presentation) with light yellow diamond nodes. The nodes are connected by edges, representing the flow of the workflow from source reports to final presentation. The code includes comments for each level and specific node definitions with their shapes and colors.

```
1 digraph Workflow {
2     // Set graph attributes for better layout
3     rankdir=TB; // Top-to-bottom layout
4     splines=line; // Use straight lines for edges
5
6     // Node styles: Define shapes and colors for each level
7     node [style=filled]; // Fill nodes with color
8
9     // Level 1: Source (Oval shape, Light Blue)
10    "IPCC Reports" [shape=oval, fillcolor=lightblue];
11    "IPCC Glossary" [shape=oval, fillcolor=lightblue];
12    "Gatsby" [shape=oval, fillcolor=lightblue];
13    "Wordpress" [shape=oval, fillcolor=lightblue];
14
15    // Level 2: Transformation and Normalization (Rectangle shape, Light Green)
16    "Document/Publishing Organization" [shape=rect, fillcolor=lightgreen];
17    "Source of Truth: HTML with IDs" [shape=rect, fillcolor=lightgreen];
18    "RDFa" [shape=rect, fillcolor=lightgreen];
19    "HTML with IDs" [shape=rect, fillcolor=lightgreen];
20    "Marked HTML" [shape=rect, fillcolor=lightgreen];
21    "Word lists" [shape=rect, fillcolor=lightgreen];
22    "Wikimedia" [shape=rect, fillcolor=lightgreen];
23    "Encyclopedia" [shape=rect, fillcolor=lightgreen];
24    "Markup other documents" [shape=rect, fillcolor=lightgreen];
25
26    // Level 3: Hyperbook Presentation (Diamond shape, Light Yellow)
27    "Publication Manifest W3C" [shape=diamond, fillcolor=lightyellow];
28    "Collation content and resources" [shape=diamond, fillcolor=lightyellow];
29    "Typesetting" [shape=diamond, fillcolor=lightyellow];
30 }
```

Fig 5.6.1 Code for generating Hyperbook Workflow Graph

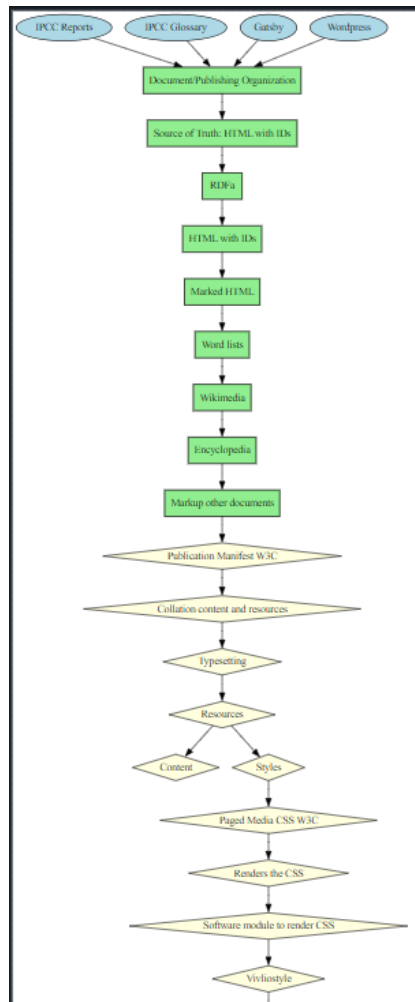


Fig 5.6.2 Hyperbook Workflow Diagram

CONCLUSION

My internship with #semanticClimate was a highly enriching experience that allowed me to contribute meaningfully to the global effort of making climate science more accessible and semantically structured. Over the course of the program, from 28th January 2025 to 1st May 2025, I worked extensively with open-source tools and technologies to process and transform complex scientific data from the IPCC Sixth Assessment Report (AR6) into structured, machine-readable formats.

I gained hands-on experience in using tools such as Python, amilib, PyMuPDF, docanalysis, and pygetpapers for text mining, PDF processing, dictionary creation, and semantic enrichment. My focus on Chapter 5 – Food, Fibre, and Other Ecosystem Products from Working Group II involved manually identifying key domain-specific terms, creating a wordlist, and generating a Wikipedia-based semantic dictionary using Google Colab. This dictionary, now publicly available in HTML format on GitHub, can serve as a valuable resource for researchers, educators, and policymakers.

In addition to data processing, I developed visualizations using Graphviz, including table of contents graphs, a cross-chapter linkage graph for AR6, and a Hyperbook Workflow Diagram, all of which were shared on GitHub in SVG format. These visual tools help illustrate the structure and transformation pipeline of the IPCC report, supporting better understanding and navigation.

I also created Python scripts to convert PDFs into plain text, one of which was applied to the Climate Academy PDF by Matthew Pye, demonstrating a scalable method for future use.

Throughout the internship, I used Git and GitHub for version control and collaborative development, ensuring clean and well-documented code contributions. Daily meetings via Zoom with Dr. Peter Murray-Rust and Dr. Renu Kumari, along with active participation on Slack, enhanced my ability to communicate effectively within a distributed team and stay aligned with project goals.

This internship not only sharpened my technical skills in programming, testing, documentation, and semantic web concepts but also deepened my understanding of climate change impacts, the importance of open science, and the urgent need for global knowledge sharing. It has inspired me to continue contributing to projects that leverage technology for environmental awareness and sustainable development.

REFERENCES

1. Murray-Rust, P., & semanticClimate contributors. (n.d.). *Amilib* – A Python library for processing and analyzing open-access scientific documents. GitHub. <https://github.com/petermr/amilib>.
2. Python Software Foundation. (2025). Python – A high-level programming language for general-purpose programming. Python Core Development Team. Retrieved from <https://www.python.org/>.
3. Ellson, J., Gansner, E. R., Koutsofios, E., & North, S. (2001). Graphviz - Graph visualization software. AT&T Labs Research. Retrieved from <https://graphviz.gitlab.io/>.
4. McKie, J. X., & Artifex Software, Inc. (n.d.). *PyMuPDF: Python bindings for MuPDF – A lightweight PDF, XPS, and eBook viewer, renderer, and toolkit*. GitHub. <https://github.com/pymupdf/pymupdf>.
5. Garg, A., Murray-Rust, P., & The OpenVirus Community. (n.d.). *pygetpapers – A command-line tool for retrieving scientific papers and metadata from open-access APIs*. GitHub. <https://github.com/petermr/pygetpapers>.
6. Murray-Rust, P., Garg, A., Hegde, S. N., & Mietchen, D. (n.d.). *docanalysis – A command-line tool for text mining and semantic analysis of scientific documents*. GitHub. <https://github.com/petermr/docanalysis>.