Mac Workflow for Tracking Daily Research Progress

2025-06-30

Table of contents

1 Introduction

Tracking daily research progress can quickly become overwhelming without a structured workflow. This guide outlines an efficient system using a well-organized directory structure, automated logging, ChatGPT for dictation and summarization, and Git for version control. By implementing this workflow, you can maintain clear, searchable research logs while reducing manual overhead.

2 Step 1: Organize Your Research Folder Structure

A structured directory keeps your research files easily accessible and prevents clutter. Use the following command to create a research workspace:

```
mkdir -p ~/prj/research_update ~/prj/{X1,X2,X3,X4,X5,X6,X7,X8,X9,X10}
```

Each project folder should contain:

- `notes.md` – Running log of project progress
- `references.bib` – Citation management
- `data/` – Datasets and related files
- `coding/` – Code and analyses
- `figures/` – Graphs and visualizations
- `tables/` – Data summaries
- `archive/` – Storage for non-current files

3 Step 2: Implement a Daily Logging System

Maintaining a centralized daily log helps consolidate research updates. Store daily progress for each repository in:

```sh
~/prj/research_update/daily_log.md
```

To streamline the process, we use a script to automatically generate a structured prompt for ChatGPT.

4 Step 3: Automate Dictation & Summarization via ChatGPT

Use macos built-in dictation tool to provide audio summary, not necessary to follow normal dictation rules e.g. “new line” ChatGPT refines dictated notes into concise summaries.

4.1 Step 3.1: Initialize a chatGPT dictation prompt by

running this `bash` script to copy a prelude to the chatGPT prompt to your clipboard: Call it `dp` (dictation prompt).

```
#!/bin/bash

# Get current date and time
current_time=$(date +"%Y-%m-%d %H:%M:%S")

# Get the current directory name
current_dir=$(basename "$PWD")

# Define the prompt with explicit instructions
prompt="I'm an academic biostatistician. I'm working on a data analysis project.
I'm about to dictate daily research progress notes.
When I'm done, provide a concise summary that includes:

1. The date  and time of dictation ($current_time).  The line with date and time
should be the second line of the summary. The first line should be blank. The
date and time line shound be enclosed in a box of ascii characters to set it apart.
2. The name of the current research project directory ($current_dir).
3. Each line of the summary including the blank line and the date and time line
and enclosing box lines should begin with \"$current_dir:\" so that it can be
extracted using ripgrep.

The notes start here: "

# Copy the prompt to clipboard (MacOS pbcopy)
echo -n "$prompt" | pbcopy

# Notify the user
echo "Prompt copied to clipboard. Paste it into ChatGPT when ready."
```

4.2 Step 3.2: Dictating Notes

1. Open ChatGPT (done automatically by "dp" script) and follow these steps:
2. copy text from clipboard into the prompt box.
3. submit prompt to prep chatGPT for summarization.
4. Click chatGPT microphone and Dictate your research notes.
5. When finished dictating submit prompt to ChatGPT for summarization.
6. Copy and generated summary onto the clipboard.

Use the following script to append the summary to your daily log: and push the changes to daily_log.md to the remote repository on GitHub.

```
#!/bin/bash

# Get the current directory name
current_dir=$(basename "$PWD")
```

```
# Get the current date and time
current_time=$(date +"%Y-%m-%d %H:%M:%S")

# Get the clipboard content (MacOS pbpaste)
clipboard_content=$(pbpaste)

# Echo the output
#
echo "$clipboard_content" >> ~/prj/research_update/daily_log.md
echo "" >> ~/prj/research_update/daily_log.md

# Confirm success
echo "Update for $current_dir appended to daily_log.md in ~/prj/research_update"
cd ~/prj/research_update
  git add .
    git commit -a -m "Daily log update $(date +'%Y-%m-%d')"
    git push
```

5 Step 5: Search & Retrieve Past Notes

Use `grep` to find past research logs:

The following script searches the daily log for the current project directory and displays the results: Call it "sp" (search project).

```
#!/bin/bash
current_dir=$(basename "$PWD")
rg $current_dir ~/prj/research_update/daily_log.md | cut -c6-
```

6 Summary of the Workflow

1. Organize project folders.
2. Set up a centralized daily log.
3. Use ChatGPT for dictation and summarization.
4. Automate log updates with a script.
5. Track changes with Git.
6. Easily search and retrieve past notes.

By integrating this workflow, you'll add structured daily documentation with minimal effort.

7 Appendix: Experimental Approach

In theory it should be possible to use ChatGPT to generate the daily log directly

7.1 Prerequisites

In development

7.2 Step-by-Step Implementation

In development

7.3 Key Takeaways

In development

7.4 Further Reading

In development