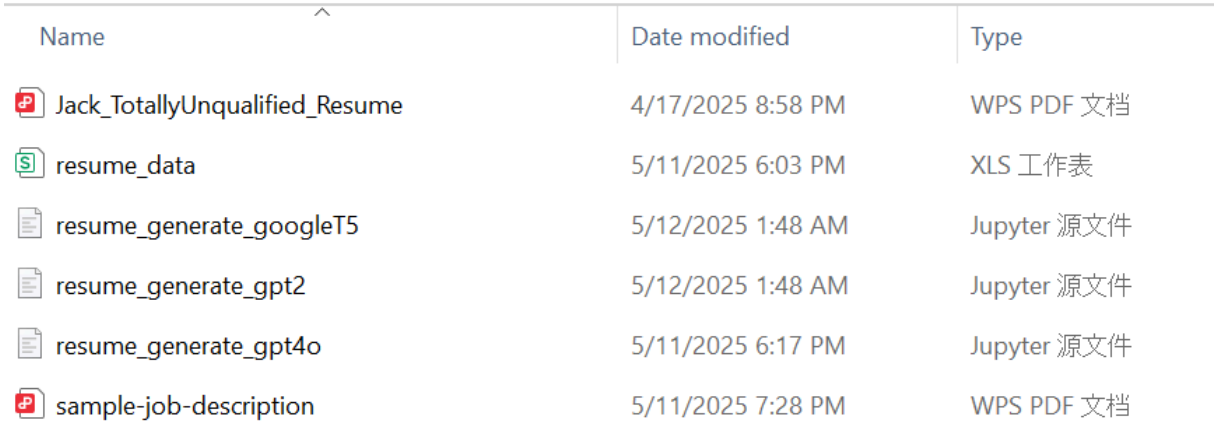
Project Tutorial

Important Notes: Before you run the coding for this project, you should see some crucial initial setup files, which are essential for the coding programs to run: **three Jupyter notebook coding files** in the project file; one **resume data csv** file for training the model; one sample **resume** PDF, and one **job description** PDF (which you could replace with your resume and job description documents)



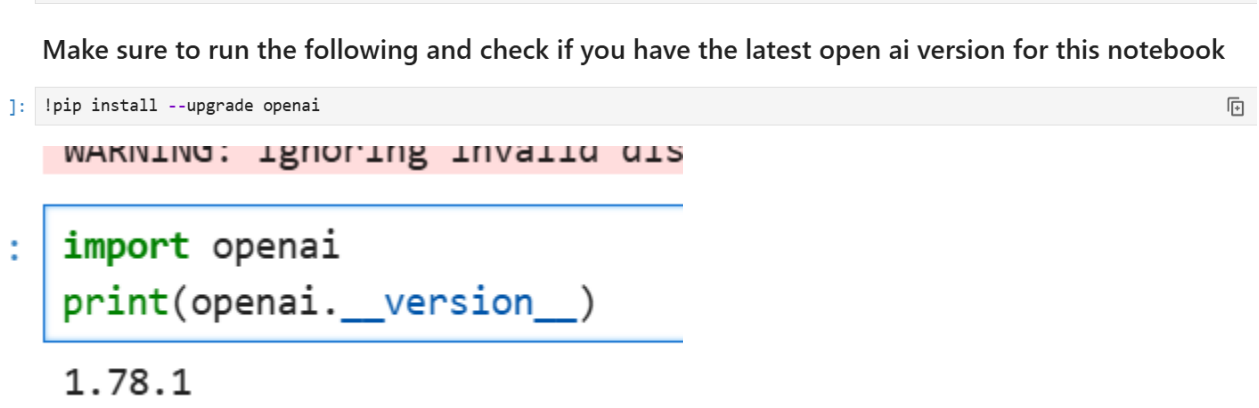
Name	Date modified	Type
Jack_TotallyUnqualified_Resume	4/17/2025 8:58 PM	WPS PDF 文档
resume_data	5/11/2025 6:03 PM	XLS 工作表
resume_generate_googleT5	5/12/2025 1:48 AM	Jupyter 源文件
resume_generate_gpt2	5/12/2025 1:48 AM	Jupyter 源文件
resume_generate_gpt4o	5/11/2025 6:17 PM	Jupyter 源文件
sample-job-description	5/11/2025 7:28 PM	WPS PDF 文档

Also, you need to prepare or have an openai api key and Hugging Face token ID before running these; and check your Python notebook version, mine is 3.13.2:

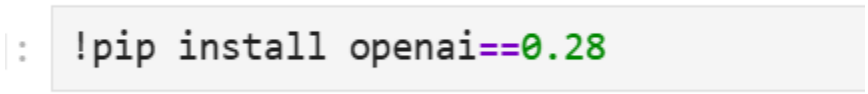
```
[ ]: import platform
print(platform. python_version())

3.13.2
```

Besides, you have to have two versions of OpenAI in your local file:

1. For the GPT4o model, you need the latest OpenAI version



   a. Then, reinstall the 0.28 version if you want to do for testing result and comparison with real resume data purpose.
2. For the GPT 2.0 model, you need to uninstall above package and using the following the OpenAI versopm



Required Steps for the notebook with GPT-4o model interaction and the notebook with training the GPT-2 model:

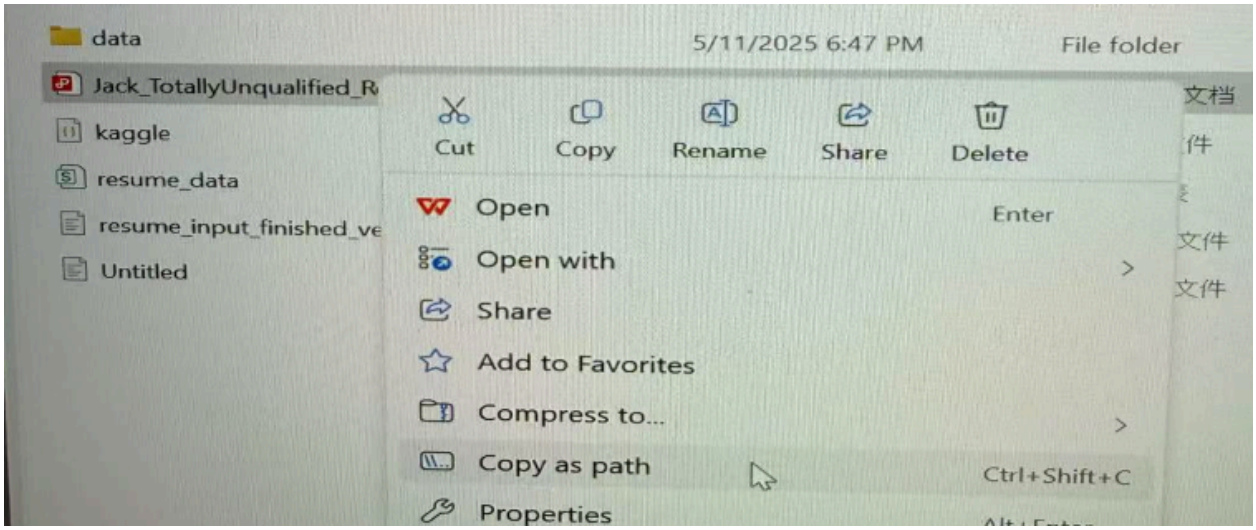
1. Environment Setup and Imports

So for this project, we need you to go to https://www.kaggle.com/datasets/saugataroyarghya/resume-dataset to download the dataset in this same file directory for ai training purpose

The link is here: https://www.kaggle.com/datasets/saugataroyarghya/resume-dataset

Download it and place it in your project if needed. I should have one downloaded in the project file for you guys to use, just in case the author updates any useful data.

For this project, run the cells accordingly.
Then, users could copy their resume file paths to the interactive interface, shown in the following screenshots.



Step1: Ask User to upload their Resume using required file path

```
*]: if __name__ == '__main__':
        input_dir = 'data/raw'
        output_dir = 'data/processed'
        dp = DataPreprocessor(input_dir, output_dir)
        dp.process()
```

```
No resume files found in 'data/raw'.
Please enter either:
  • An absolute or relative file path (e.g. C:\Users\Rick\Desktop\resume.pdf or ~/resumes/resume.pdf)
  • A filename to fuzzy-search your workspace (e.g. 'resume.pdf')
Enter directory path, file path, or filename:
```

Then, after running required cells, go to this website to find your own OpenAI key:
https://platform.openai.com/api-keys
Copy and paste it to the required place, shown in the following screenshot

Step2: Ask User to enter their own Api Key

```
:  # Cell 1: Prompt user for API key
   print("Note: This pipeline currently uses OpenAI. Please enter your OpenAI API key.")
   api_key = input("API key: ")
```

```
Note: This pipeline currently uses OpenAI. Please enter your OpenAI API key.
API key:
```

Then, copy and paste your required job description file into the environment:

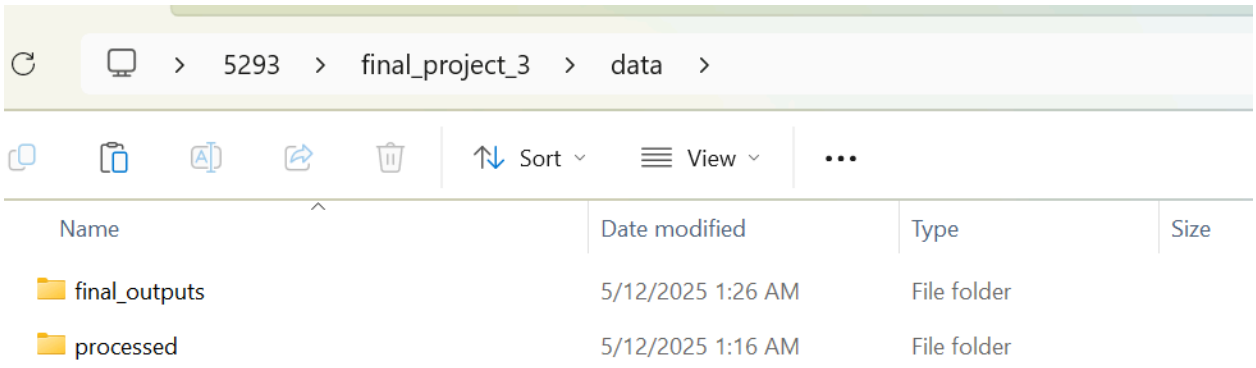
Step3: Ask User to put their own job description with required format needed

```
print("\nEnter job description (.txt/.pdf/.docx path or URL):")
src = input()
job_description = load_job_description(src)
```

```
Enter job description (.txt/.pdf/.docx path or URL):
```

That's all the user is required to proceed with this project. The following is just to run all the remaining code, and the output will be saved in the working directory:
The project file → data → final_outputs: (except the project file is the one you download from this project, other files are created by running the code in the notebook)

You can find your generated resume and cover letter in the final_output file.



Required Steps for the notebook with training a T5 model:

These steps are similar, but there are two differences here:

1. **Instead of providing the OpenAI API key, we require the user to provide their Hugging Face token ID.**

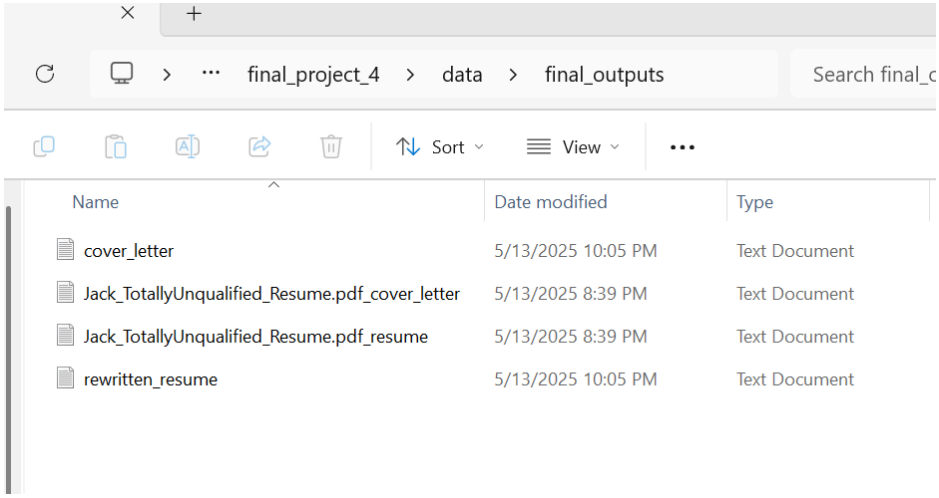
Step3: Ask User to enter their own hugging face token key

```
[*]: import os
import pandas as pd
from datasets import Dataset
import torch
from transformers import (
    AutoTokenizer,
    AutoModelForSeq2SeqLM,
    DataCollatorForSeq2Seq,
    Seq2SeqTrainingArguments,
    Seq2SeqTrainer
)


# 1) Prompt user for Hugging Face access token
hf_token = input("Enter your Hugging Face access token: ").strip()
```

```
The installed version of bitsandbytes was compiled without GPU support. 8-bit optimizers, 8-bit multiplic
e.
Enter your Hugging Face access token:
```

2. **You need to go to the Google-T5_base model website to grant access for the model: https://huggingface.co/google-t5/t5-base**

After these setups and running code accordingly, you should see the output in the project file → data → final_outputs **in text format of your resume and cover letter, like the following:**



Then, you are good to go!