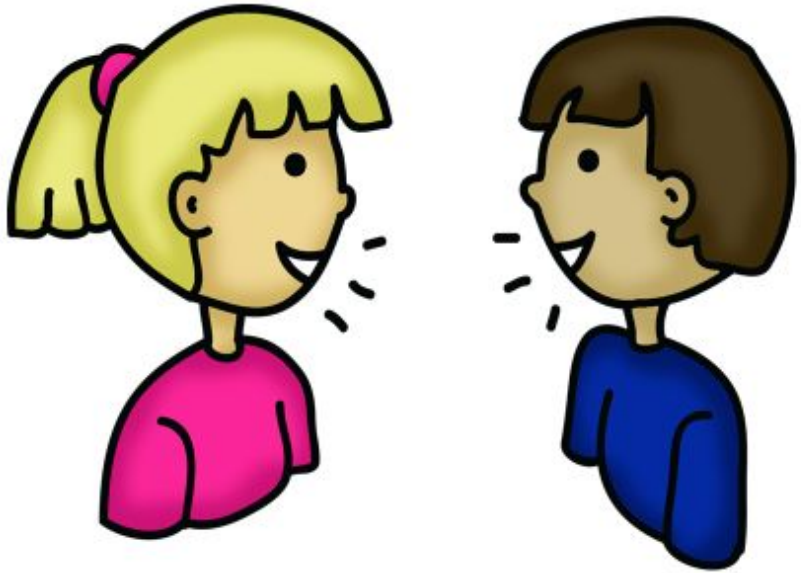
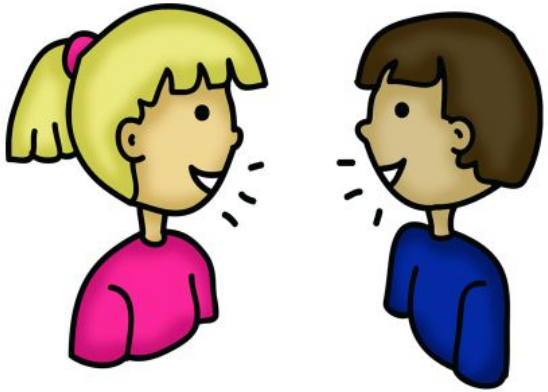
Google Colab & Gemini for batch speech analysis

daniels@kochi-tech.ac.jp

Speaking assessment



Batch assessment

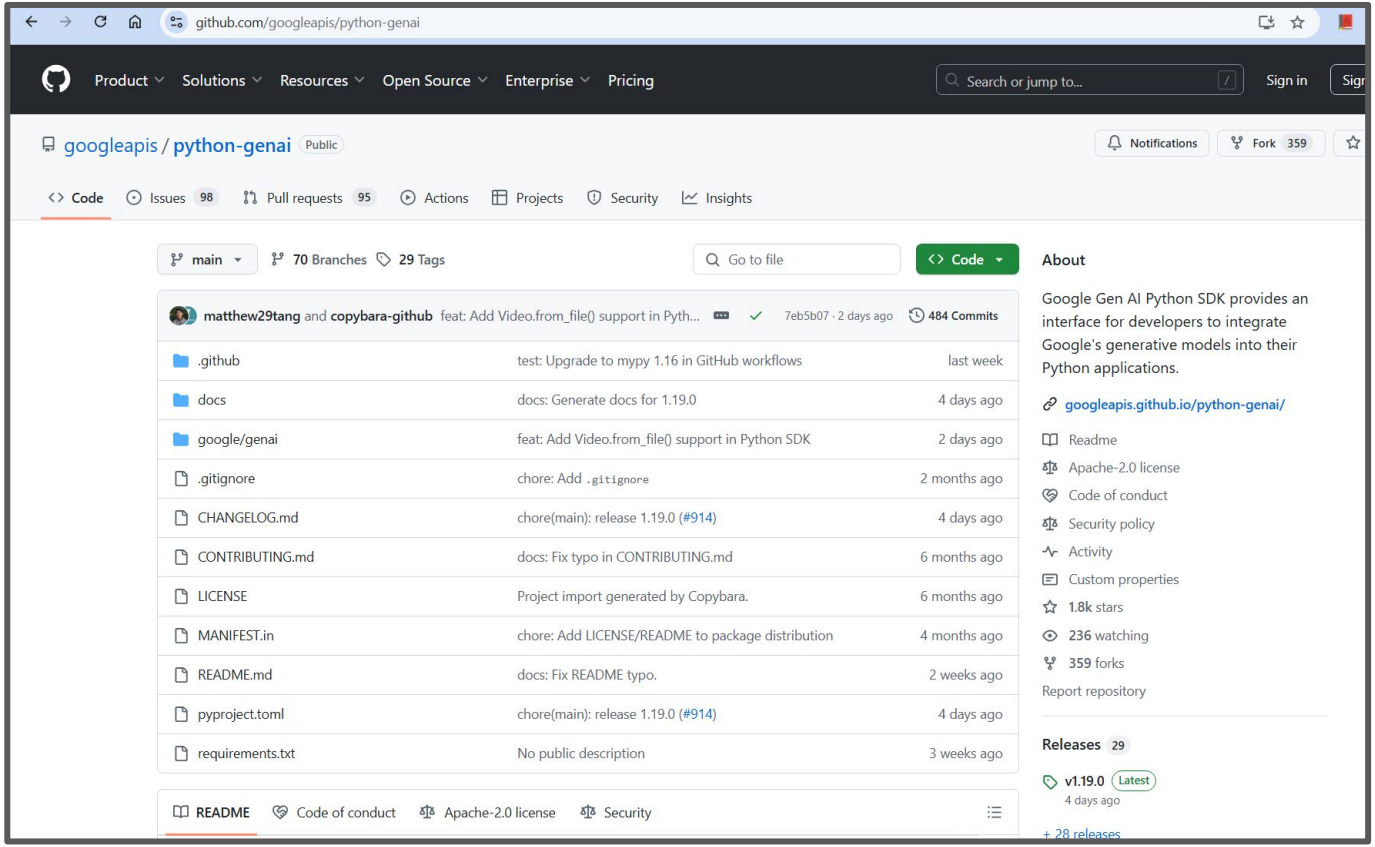


- audio to text
- prosody & fluency
- grammar & vocabulary
- CEFR level
- words spoken & sentence length
- lexical density

Colaboratory

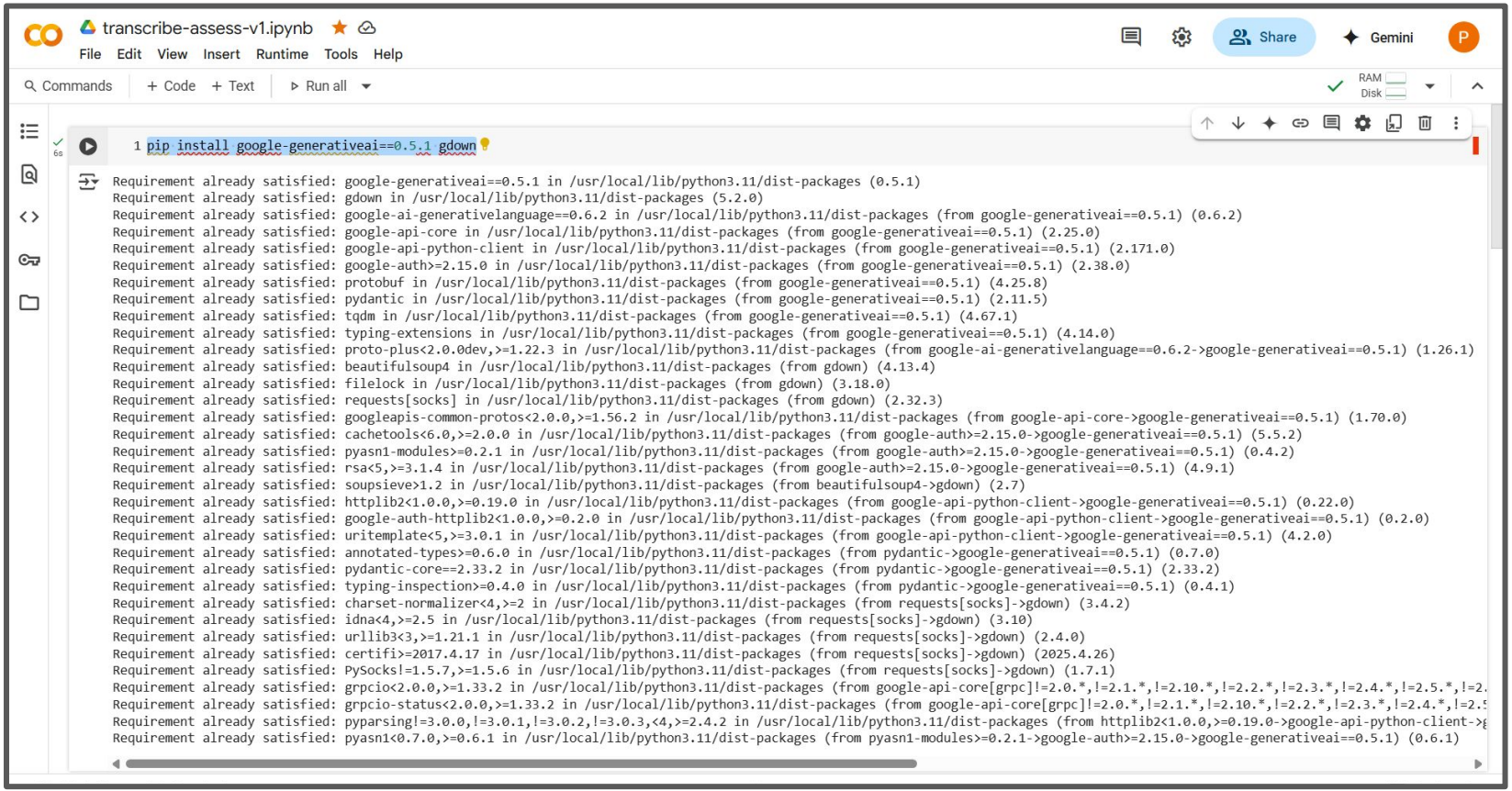
- Free
- Access to computing resources (GPU / TPU)
- Access via web browser
- Write & execute Python code
- Machine learning & data science
- No setup or configuration

Google Gen AI Python SDK

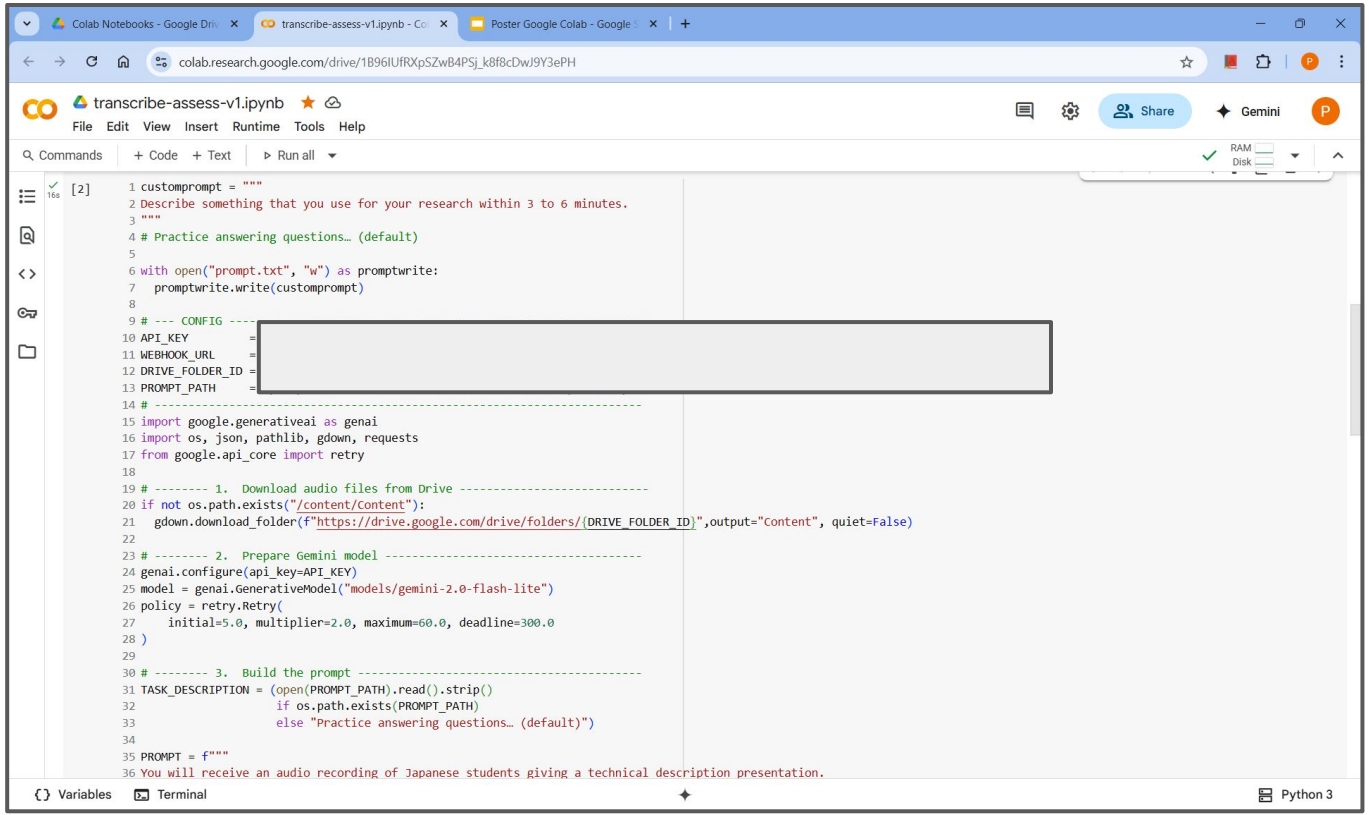


integrate Google's generative models into Python applications

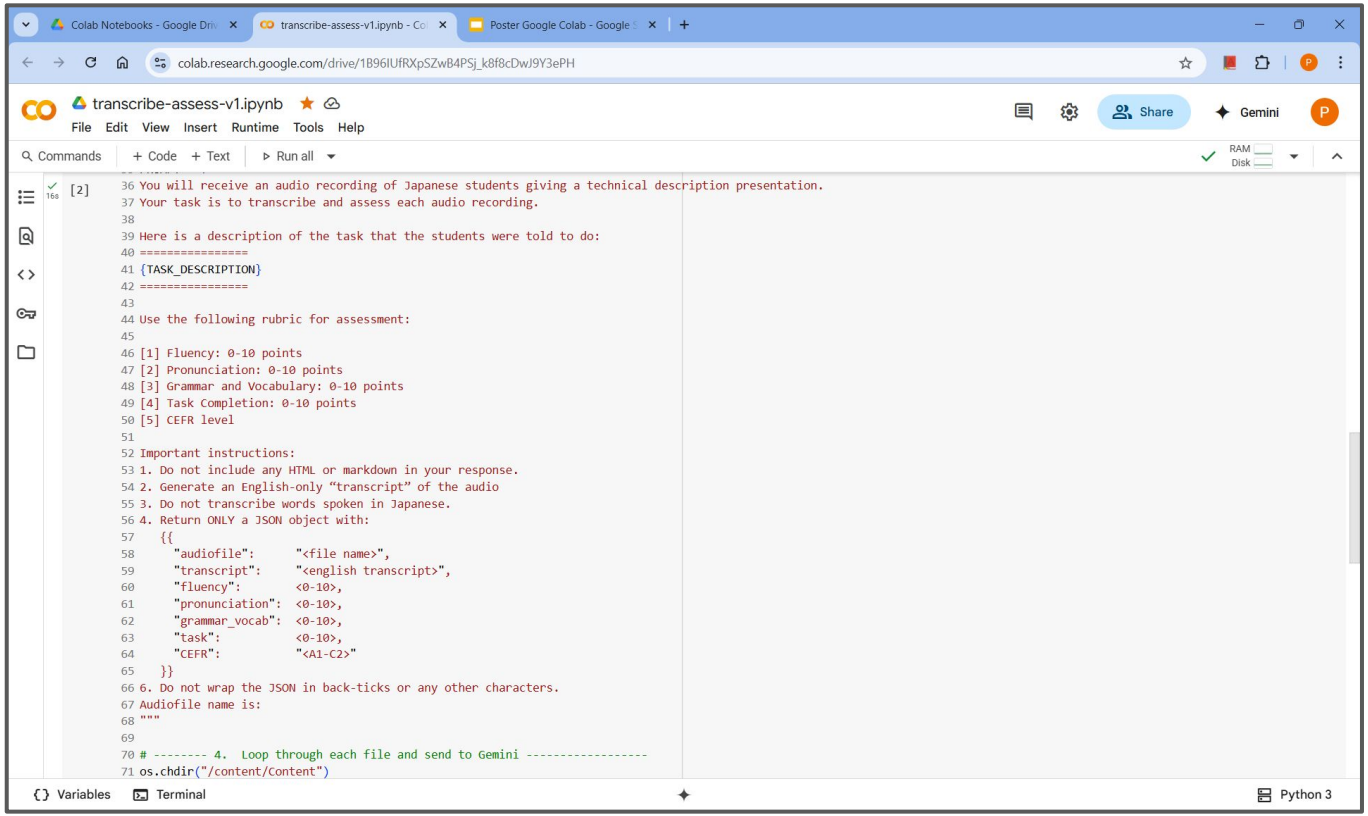
Install Google generative AI



Copy Python code

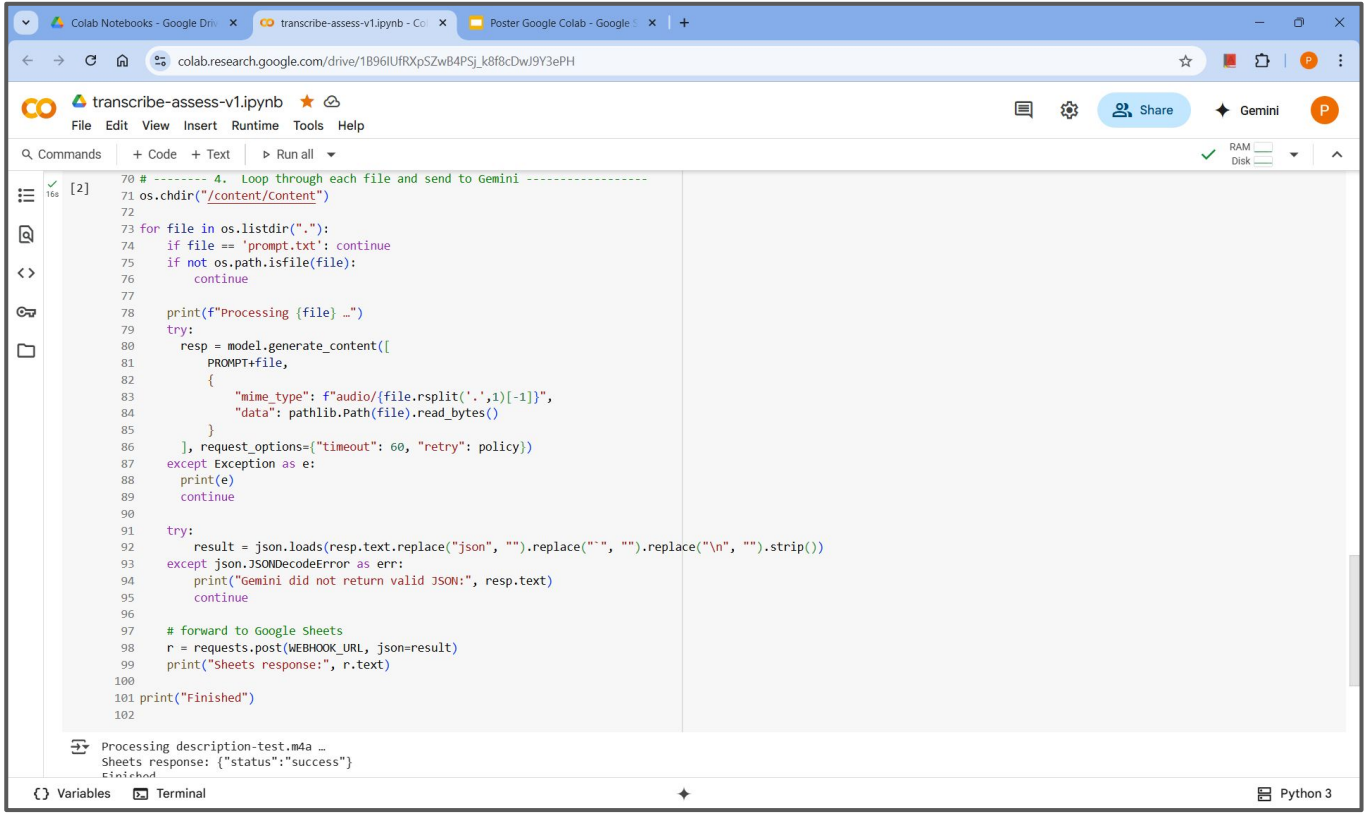


Python code

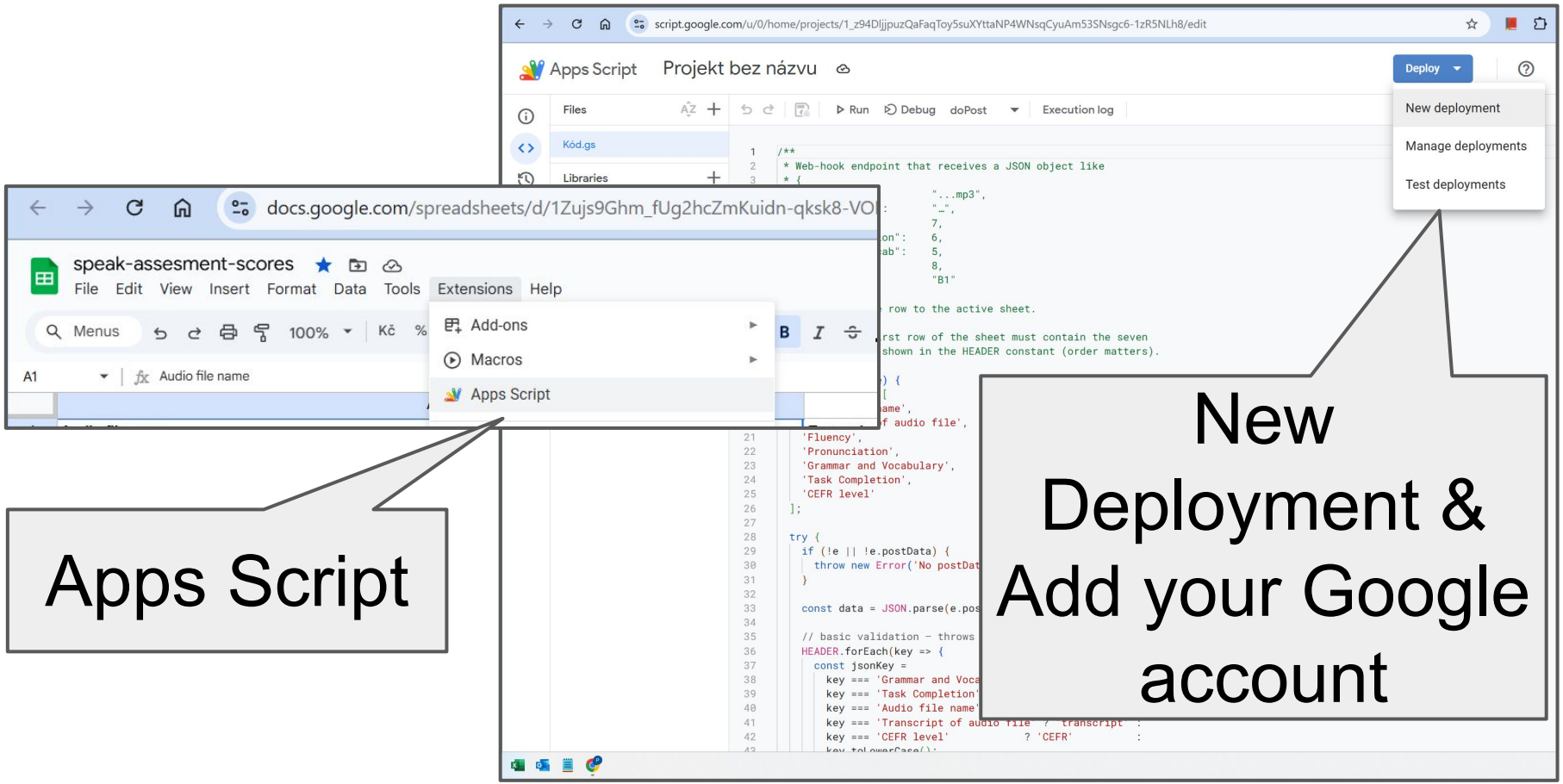


```
You will receive an audio recording of Japanese students giving a technical description presentation.
Your task is to transcribe and assess each audio recording.

Here is a description of the task that the students were told to do:
================
{TASK_DESCRIPTION}
================

Use the following rubric for assessment:

[1] Fluency: 0-10 points
[2] Pronunciation: 0-10 points
[3] Grammar and Vocabulary: 0-10 points
[4] Task Completion: 0-10 points
[5] CEFR level

Important instructions:
1. Do not include any HTML or markdown in your response.
2. Generate an English-only "transcript" of the audio
3. Do not transcribe words spoken in Japanese.
4. Return ONLY a JSON object with:
   {{
     "audiofile":      "<file name>",
     "transcript":     "<english transcript>",
     "fluency":        <0-10>,
     "pronunciation":  <0-10>,
     "grammar_vocab":  <0-10>,
     "task":           <0-10>,
     "CEFR":           "<A1-C2>"
   }}
6. Do not wrap the JSON in back-ticks or any other characters.
Audiofile name is:
"""

# -------- 4.  Loop through each file and send to Gemini ------------------
os.chdir("/content/Content")
```

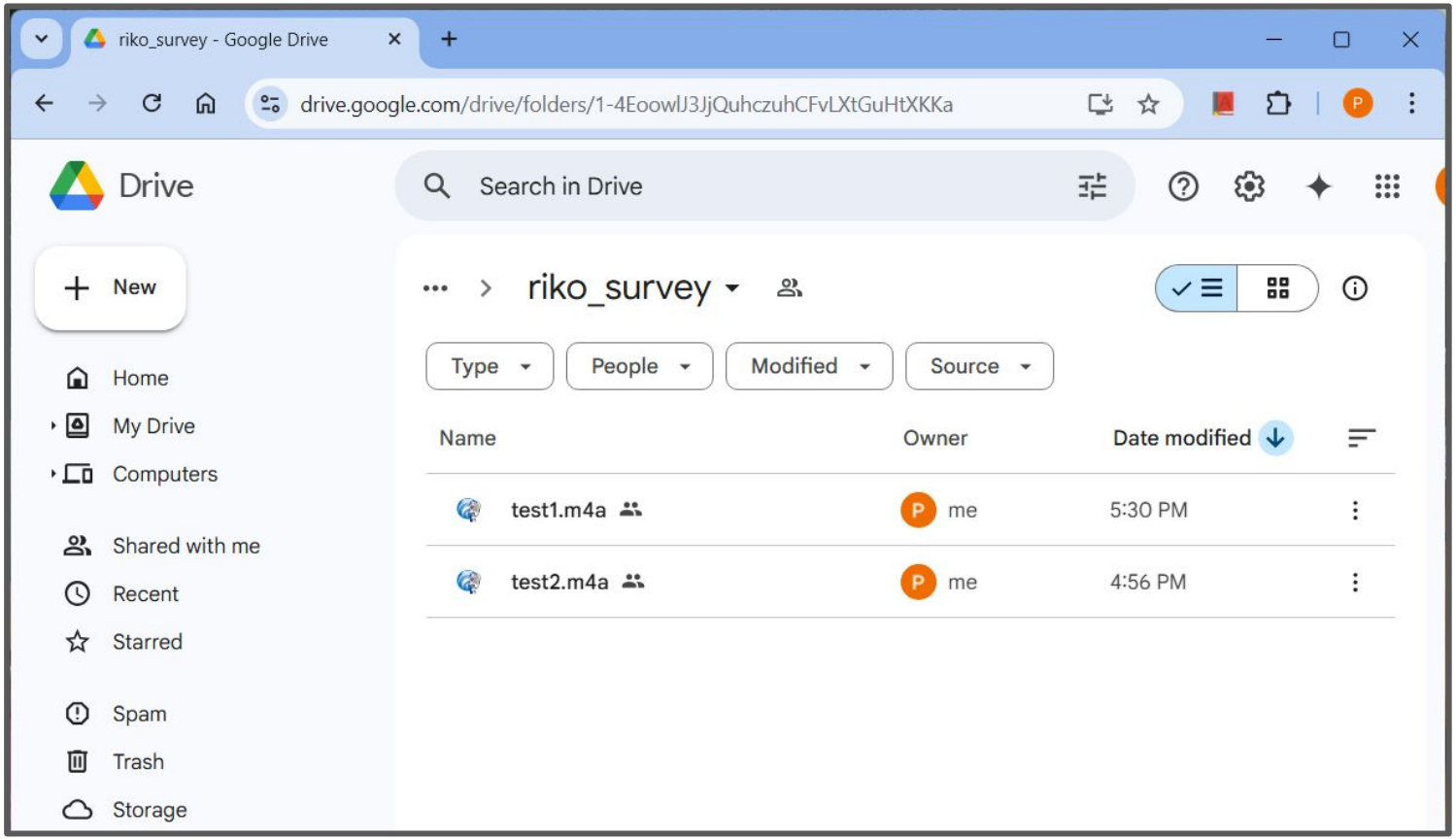
Python code


transcribe-assess-v1.ipynb
File Edit View Insert Runtime Tools Help
Share
Gemini
Commands + Code + Text Run all
RAM
Disk
[2]
# -------- 4. Loop through each file and send to Gemini ----------
os.chdir("/content/Content")
for file in os.listdir("."):
if file == 'prompt.txt': continue
if not os.path.isfile(file):
continue
print(f"Processing {file} …")
try:
resp = model.generate_content([
PROMPT+file,
{
"mime_type": f"audio/{file.rsplit('.',1)[-1]}",
"data": pathlib.Path(file).read_bytes()
}
], request_options={"timeout": 60, "retry": policy})
except Exception as e:
print(e)
continue
try:
result = json.loads(resp.text.replace("json", "").replace("`", "").replace("\n", "").strip())
except json.JSONDecodeError as err:
print("Gemini did not return valid JSON:", resp.text)
continue
# forward to Google Sheets
r = requests.post(WEBHOOK_URL, json=result)
print("Sheets response:", r.text)
print("Finished")
Processing description-test.m4a …
Sheets response: {"status":"success"}
Variables
Terminal
Python 3

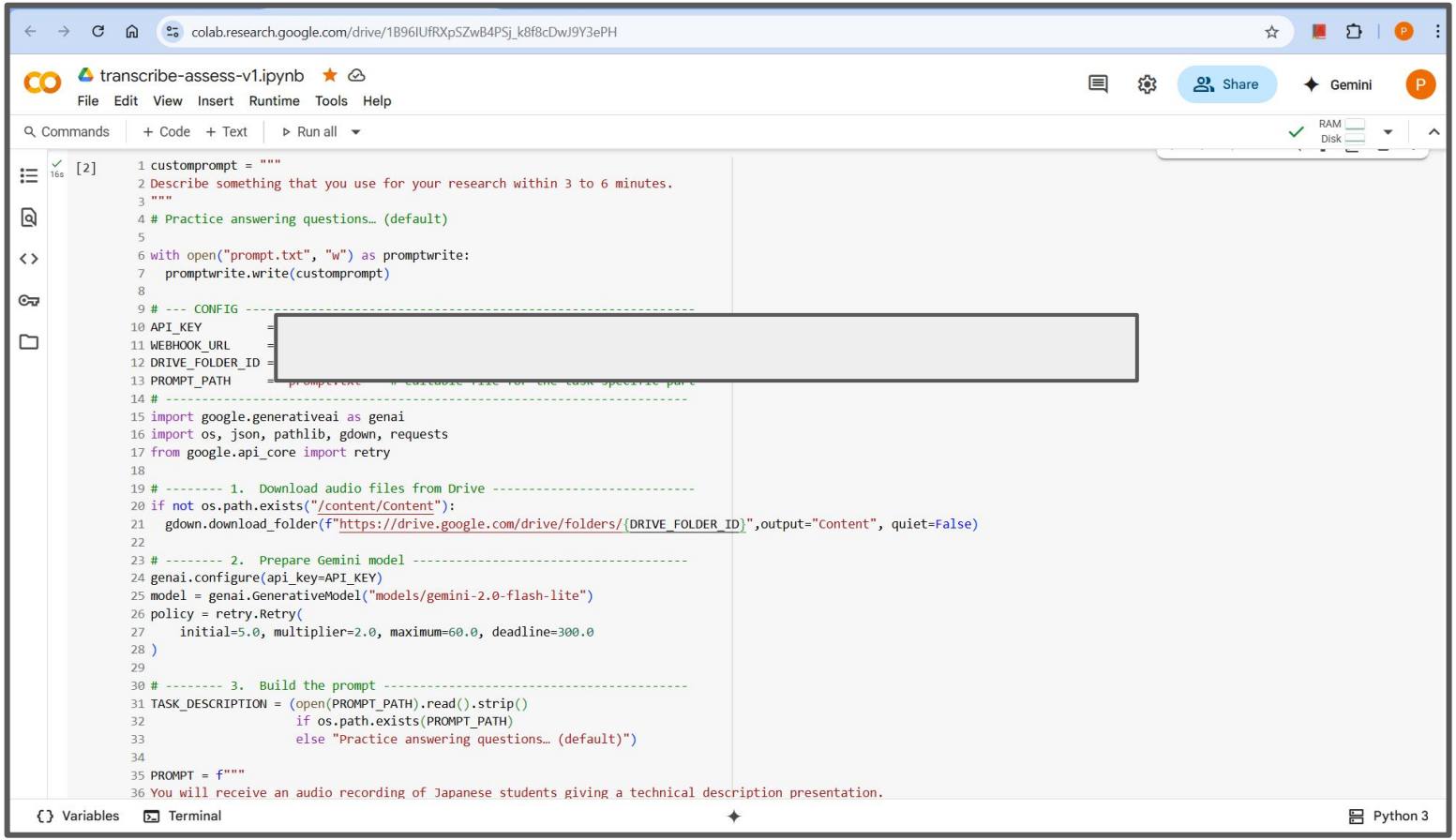
Google Sheets Apps Script



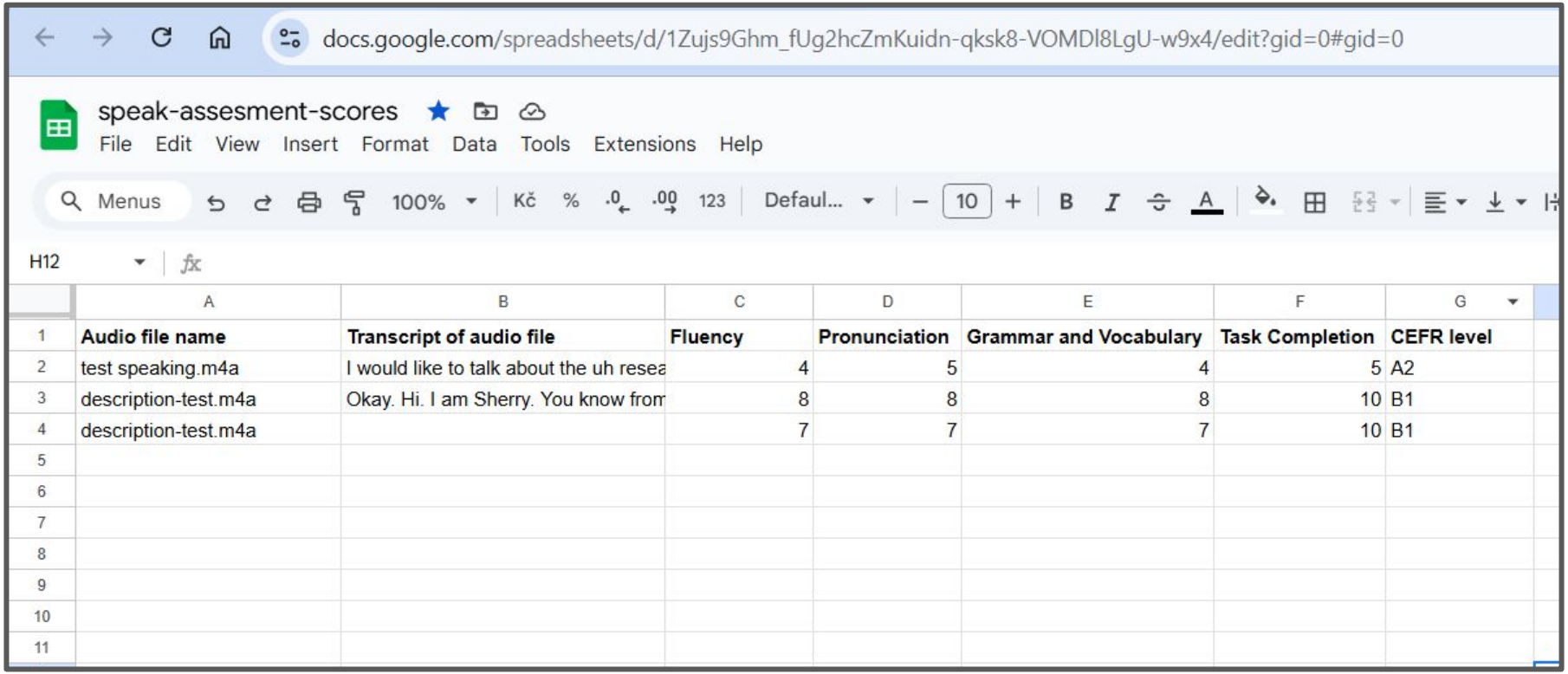
Add audio files to Drive



Run script


colab.research.google.com/drive/1B96IUfRXpSZwB4PSj_k8f8cDwJ9Y3ePH
transcribe-assess-v1.ipynb
File Edit View Insert Runtime Tools Help
Share
Gemini
Commands
+ Code
+ Text
Run all
RAM
Disk
[2]
16s
1 custompromt = """
2 Describe something that you use for your research within 3 to 6 minutes.
3 """
4 # Practice answering questions… (default)
5
6 with open("prompt.txt", "w") as promptwrite:
7 promptwrite.write(custompromt)
8
9 # --- CONFIG ---
10 API_KEY =
11 WEBHOOK_URL =
12 DRIVE_FOLDER_ID =
13 PROMPT_PATH =
14 # ---
15 import google.generativeai as genai
16 import os, json, pathlib, gdown, requests
17 from google.api_core import retry
18
19 # -------- 1. Download audio files from Drive ----------
20 if not os.path.exists("/content/Content"):
21 gdown.download_folder(f"https://drive.google.com/drive/folders/{DRIVE_FOLDER_ID}",output="Content", quiet=False)
22
23 # -------- 2. Prepare Gemini model ----------
24 genai.configure(api_key=API_KEY)
25 model = genai.GenerativeModel("models/gemini-2.0-flash-lite")
26 policy = retry.Retry(
27 initial=5.0, multiplier=2.0, maximum=60.0, deadline=300.0
28)
29
30 # -------- 3. Build the prompt ----------
31 TASK_DESCRIPTION = (open(PROMPT_PATH).read().strip()
32 if os.path.exists(PROMPT_PATH)
33 else "Practice answering questions… (default)")
34
35 PROMPT = f"""
36 You will receive an audio recording of Japanese students giving a technical description presentation.
Variables
Terminal
Python 3

Results posted to Google Sheets



docs.google.com/spreadsheets/d/1Zujs9Ghm_fUg2hcZmKuidn-qksk8-VOMDl8LgU-w9x4/edit?gid=0#gid=0

speak-assesment-scores

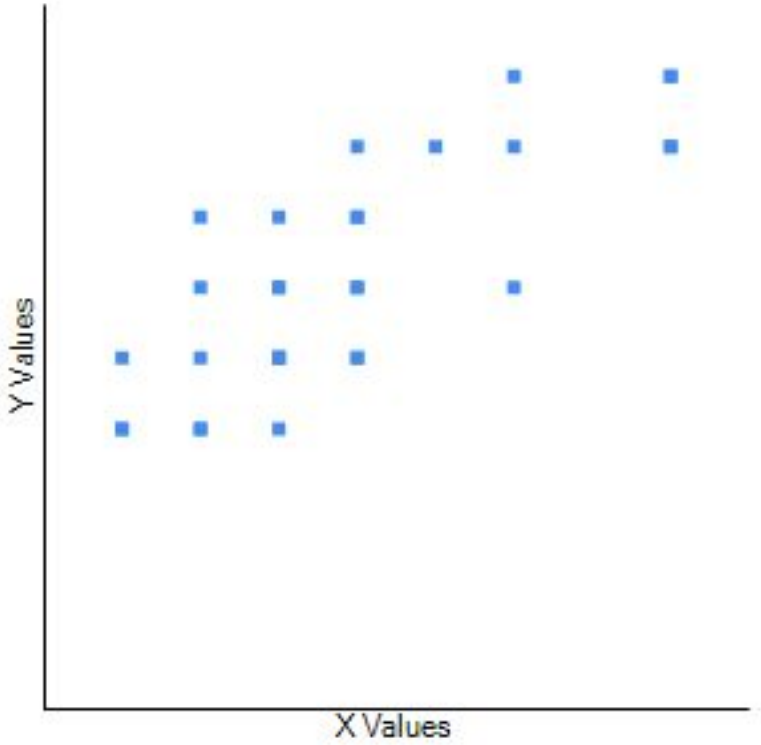
File Edit View Insert Format Data Tools Extensions Help

Menus 100% Kč % .0 .00 123 Defaul... 10 B I A

H12

	A	B	C	D	E	F	G
1	**Audio file name**	**Transcript of audio file**	**Fluency**	**Pronunciation**	**Grammar and Vocabulary**	**Task Completion**	**CEFR level**
2	test speaking.m4a	I would like to talk about the uh resea	4	5	4	5	A2
3	description-test.m4a	Okay. Hi. I am Sherry. You know from	8	8	8	10	B1
4	description-test.m4a		7	7	7	10	B1
5							
6							
7							
8							
9							
10							
11							

Human & AI scores



R = 0.7953
N = 31

Positive correlation