

CV Screener - Currently I am using a single cv as input. Trying to parse the data from the cv using the ocr. Once that is done the data is then parsed to llm to fetch the data which includes personal information, education, experience, skills, certification and project. Once the information gets extracted then it's then saved as a json file. Then as the last step I created a chatbot interface using gradio to query the data fetched from that particular cv.

Python packages and dependencies to be installed

```
!pip install frontend autogen pyautogen[retrievechat] groq frontend fitz PyPDF2 PyMuPDF chromadb pypdf markdownify sentence_transformers==4.2.0
Requirement already satisfied: nvidia-cusolver-cu12==11.6.1.9 in /usr/local/lib/python3.11/dist-packages (from torch>=1.11.0->sentence_transformers==4.2.0)
Requirement already satisfied: nvidia-cusparse-cu12==12.3.1.170 in /usr/local/lib/python3.11/dist-packages (from torch>=1.11.0->sentence_transformers==4.2.0)
Requirement already satisfied: nvidia-nccl-cu12==2.21.5 in /usr/local/lib/python3.11/dist-packages (from torch>=1.11.0->sentence_transformers==4.2.0)
Requirement already satisfied: nvidia-nvtx-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch>=1.11.0->sentence_transformers==4.2.0)
Requirement already satisfied: nvidia-nvjitlink-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch>=1.11.0->sentence_transformers==4.2.0)
Requirement already satisfied: triton==3.1.0 in /usr/local/lib/python3.11/dist-packages (from torch>=1.11.0->sentence_transformers==4.2.0)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.11/dist-packages (from sympy>onnxruntime>1.14.1->chromadb)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.11/dist-packages (from transformers<5.0.0,>=4.41.0->sentence_transformers==4.2.0)
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.11/dist-packages (from transformers<5.0.0,>=4.41.0->sentence_transformers==4.2.0)
Requirement already satisfied: click>=8.0.0 in /usr/local/lib/python3.11/dist-packages (from typer>=0.9.0->chromadb) (8.1.8)
Requirement already satisfied: shellingham>=1.3.0 in /usr/local/lib/python3.11/dist-packages (from typer>=0.9.0->chromadb) (1.5.4)
Requirement already satisfied: httptools>=0.6.3 in /usr/local/lib/python3.11/dist-packages (from uvicorn[standard]>=0.18.3->chromadb)
Requirement already satisfied: uvloop!=0.15.0,!0.15.1,>=0.14.0 in /usr/local/lib/python3.11/dist-packages (from uvicorn[standard]>=0.18.3->chromadb)
Requirement already satisfied: watchfiles>=0.13 in /usr/local/lib/python3.11/dist-packages (from uvicorn[standard]>=0.18.3->chromadb)
Requirement already satisfied: pyParsing!=3.0.0,!=3.0.1,!=3.0.2,!=3.0.3,<4,>=2.4.2 in /usr/local/lib/python3.11/dist-packages (from h5py>=3.8.0->chromadb)
Requirement already satisfied: setuptools>=18.5 in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat]) (7.5.1)
Requirement already satisfied: jedi>=0.16 in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat]) (0.19.2)
Requirement already satisfied: decorator in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat]) (4.4.2)
Requirement already satisfied: pickleshare in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat]) (0.7.5)
Requirement already satisfied: traitlets>=4.2 in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat]) (5.7.0)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat])
Requirement already satisfied: backcall in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat]) (0.2.0)
Requirement already satisfied: matplotlib-inline in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat]) (0.1.1)
Requirement already satisfied: pexpect>4.3 in /usr/local/lib/python3.11/dist-packages (from ipython->pyautogen[retrievechat]) (4.9.0)
Requirement already satisfied: prov>=1.5.2 in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (2.0.1)
Requirement already satisfied: pydot>=1.2.3 in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (3.0.4)
Requirement already satisfied: rdflib>=5.0.0 in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (6.3.2)
Requirement already satisfied: simplejson>=3.8.0 in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (3.20.1)
Requirement already satisfied: traits>=6.2 in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (7.0.2)
Requirement already satisfied: acres in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (0.3.0)
Requirement already satisfied: etelemetry>=0.3.1 in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (0.3.1)
Requirement already satisfied: looseversion!=1.2 in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (1.3.0)
Requirement already satisfied: puremagic in /usr/local/lib/python3.11/dist-packages (from nipyapi->fitz) (1.28)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas->fitz) (2025.1)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas->fitz) (2025.1)
Requirement already satisfied: lxml>=4.3 in /usr/local/lib/python3.11/dist-packages (from pyxnat->fitz) (5.3.1)
Requirement already satisfied: pathlib>=1.0 in /usr/local/lib/python3.11/dist-packages (from pyxnat->fitz) (1.0.1)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.11/dist-packages (from scikit-learn->sentence_transformers) (1.4.0)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.11/dist-packages (from scikit-learn->sentence_transformers)
Requirement already satisfied: ci-info>=0.2 in /usr/local/lib/python3.11/dist-packages (from etelemetry>=0.3.1->nipyapi->fitz) (0.3.0)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.11/dist-packages (from google-auth>=1.0.1->kubernetes)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.11/dist-packages (from google-auth>=1.0.1->kubernetes)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.11/dist-packages (from google-auth>=1.0.1->kubernetes) (28.1.0)
Requirement already satisfied: zipp>=3.20 in /usr/local/lib/python3.11/dist-packages (from importlib-metadata<=8.5.0,>=6.0->openpyxl)
Requirement already satisfied: parso<0.9.0,>=0.8.4 in /usr/local/lib/python3.11/dist-packages (from jedi>=0.16->ipython->pyautogen[retriever])
Requirement already satisfied: jsonpointer>=1.9 in /usr/local/lib/python3.11/dist-packages (from jsonpatch<2.0,>=1.33->langchain-core)
Requirement already satisfied: mdurl~0.1 in /usr/local/lib/python3.11/dist-packages (from markdown-it-py>=2.2.0->rich) (0.1.0)
Requirement already satisfied: ptyprocess>=0.5 in /usr/local/lib/python3.11/dist-packages (from pexpect>4.3->ipython->pyautogen[retriever])
Requirement already satisfied: wcwidth in /usr/local/lib/python3.11/dist-packages (from prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0)
Requirement already satisfied: isodate<0.7.0,>=0.6.0 in /usr/local/lib/python3.11/dist-packages (from rdflib>=5.0.0->nipyapi->fitz) (0.8.0)
Requirement already satisfied: mypy-extensions>=0.3.0 in /usr/local/lib/python3.11/dist-packages (from typing-inspect<1,>=0.4.0->dataclasses)
Requirement already satisfied: humanfriendly>=9.1 in /usr/local/lib/python3.11/dist-packages (from coloredlogs>onnxruntime>1.14.1->chromadb)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-packages (from jinja2>torch>1.11.0->sentence_transformers)
Requirement already satisfied: pyasn1<0.7.0,>=0.4.6 in /usr/local/lib/python3.11/dist-packages (from pyasn1-modules>=0.2.1->google-auth)
Using cached fitz-0.0.1.dev2-py2.py3-none-any.whl (20 kB)
Installing collected packages: fitz
Successfully installed fitz-0.0.1.dev2
```

```
# from google.colab import drive
# drive.mount('/content/drive')
```

Setting up ocr in my machine and fetching the information from the pdf

```
!apt-get update
!apt-get install -y tesseract-ocr
```

```
→ Hit:1 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Hit:4 https://cloud.r-project.org/bin/linux/ubuntu jammy-cran40/ InRelease
Hit:5 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86_64 InRelease
Hit:6 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:7 https://r2u.stat.illinois.edu/ubuntu jammy InRelease
Hit:8 https://ppa.launchpadcontent.net/deadsnakes/ppa/ubuntu jammy InRelease
Hit:9 https://ppa.launchpadcontent.net/graphics-drivers/ppa/ubuntu jammy InRelease
Hit:10 https://ppa.launchpadcontent.net/ubuntugis/ppa/ubuntu jammy InRelease
Fetched 257 kB in 1s (199 kB/s)
Reading package lists... Done
W: Skipping acquire of configured file 'main/source/Sources' as repository 'https://r2u.stat.illinois.edu/ubuntu jammy InRelease' does not have a Release file
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
tesseract-ocr is already the newest version (4.1.1-2.1build1).
0 upgraded, 0 newly installed, 0 to remove and 33 not upgraded.
```

```
!pip install pytesseract
```

```
→ Requirement already satisfied: pytesseract in /usr/local/lib/python3.11/dist-packages (0.3.13)
Requirement already satisfied: packaging>=21.3 in /usr/local/lib/python3.11/dist-packages (from pytesseract) (24.2)
Requirement already satisfied: Pillow>=8.0.0 in /usr/local/lib/python3.11/dist-packages (from pytesseract) (11.1.0)
```

```
import pytesseract
pytesseract.pytesseract.tesseract_cmd = r'/usr/bin/tesseract' #setting exe file in windows to avoid error in processing
```

```
# !pip install PyMuPDF
# !pip install pytesseract==0.3.8
# !pip install pillow==8.4.0
```

```
!pip install PymuPDF==1.23.3
```

```
→ Collecting PymuPDF==1.23.3
  Downloading PyMuPDF-1.23.3-cp311-none-manylinux2014_x86_64.whl.metadata (3.4 kB)
Collecting PyMuPDFB==1.23.3 (from PymuPDF==1.23.3)
  Downloading PyMuPDFB-1.23.3-py3-none-manylinux2014_x86_64.manylinux_2_17_x86_64.whl.metadata (1.3 kB)
  Downloading PyMuPDF-1.23.3-cp311-none-manylinux2014_x86_64.whl (4.3 MB)
    ━━━━━━━━━━━━━━━━ 4.3/4.3 MB 32.0 MB/s eta 0:00:00
  Downloading PyMuPDFB-1.23.3-py3-none-manylinux2014_x86_64.manylinux_2_17_x86_64.whl (30.6 MB)
    ━━━━━━━━━━━━━━━━ 30.6/30.6 MB 59.4 MB/s eta 0:00:00
Installing collected packages: PyMuPDFB, PymuPDF
  Attempting uninstall: PymuPDF
    Found existing installation: PyMuPDF 1.25.3
    Uninstalling PyMuPDF-1.25.3:
      Successfully uninstalled PyMuPDF-1.25.3
  Successfully installed PyMuPDFB-1.23.3 PymuPDF-1.23.3
```

```
!pip uninstall fitz
# !pip uninstall PyMuPDF
# !pip install --upgrade PyMuPDF
# !pip install --upgrade --force-reinstall pymupdf
```

```
→ Found existing installation: fitz 0.0.1.dev2
Uninstalling fitz-0.0.1.dev2:
Would remove:
  /usr/local/bin/fitz
  /usr/local/bin/log2design.py
  /usr/local/lib/python3.11/dist-packages/.DS_Store
  /usr/local/lib/python3.11/dist-packages/fitz-0.0.1.dev2.dist-info/*
  /usr/local/lib/python3.11/dist-packages/fitz/*
  /usr/local/lib/python3.11/dist-packages/scripts/*
Would not remove (might be manually added):
  /usr/local/lib/python3.11/dist-packages/fitz/table.py
  /usr/local/lib/python3.11/dist-packages/fitz/utils.py
  /usr/local/lib/python3.11/dist-packages/scripts/readme-gen/readme_gen.py
  /usr/local/lib/python3.11/dist-packages/scripts/run_emscripten_tests.py
  /usr/local/lib/python3.11/dist-packages/scripts/test_imports.py
  /usr/local/lib/python3.11/dist-packages/scripts/test_leak.py
```

```
Proceed (Y/n)? Y
Successfully uninstalled fitz-0.0.1.dev2
```

1. DOCUMENT PROCESSING

```
import fitz # PyMuPDF
import pytesseract
from PIL import Image
import io

# Function to extract text using OCR from a PDF
def ocr_pdf(pdf_path):
    # Open the PDF
    document = fitz.open(pdf_path)
    extracted_text = ""

    # Iterate through each page of the PDF
    for page_num in range(document.page_count):
        page = document.load_page(page_num)

        # Get the images from the page
        image_list = page.get_images(full=True)

        # Process each image on the page
        for img_index, img in enumerate(image_list):
            xref = img[0]
            base_image = document.extract_image(xref)
            image_bytes = base_image["image"] # Image in bytes

            # Convert bytes to an image using PIL
            image = Image.open(io.BytesIO(image_bytes))

            # Perform OCR on the image
            text = pytesseract.image_to_string(image)

            # Append extracted text to the result
            extracted_text += text

    return extracted_text

# Example usage
pdf_path = 'File.pdf'
text = ocr_pdf(pdf_path)

# Print the extracted text
# print(text)
```

Output of document processing

```
text
→ 'GIULIA GONZALEZ\n\nCONTACT\nngonzalez@email.com\n(123) 456-7890\nDetroit, MI\nLinkedIn\nGitHub\nEDUCATION\nMS.\nComputer Science\nUniversity of Chicago\n2014 - 2016\nChicago, IL\nB.S.\nComputer Science\nUniversity of Pittsburgh\n2010 - 2014\nPittsburgh, PA\nSKILLS\nHTML/CSS\nSQL (PostgreSQL, Oracle)\nJavaScript (Angular)\nPython (Django)\nREST APIs (GraphQL)\nAWS (Redshift, S3)\nGit\nTHON DEVELOPER\nWORK EXPERIENCE\nPython Developer\nDoorDash\nSeptember 2017 - current / Detroit, MI\nWorked on building new Angular components for the customer-facing\nweb app, which improved the time on page for the average user by 2\nminutes\nCollaborated with an agile team of 6, and helped prioritize and scope\nfeature requests to ensure that the biggest impact features were\nworked on first\nBuilt extensive test coverage for all new features, which reduced the\nnumber of customer complaints by 23%\nAcquired and ingested data to build and maintain data p...'
```

2. LLM INTEGRATION

```
import os
from groq import Groq
config_list = [{"model": "llama3-70b-8192", "api_key": "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", "base_url": "https://api.groq.com"}]
```

```
import chromadb
import autogen
from autogen.agentchat.contrib.retrieve_assistant_agent import RetrieveAssistantAgent
from autogen.agentchat.contrib.retrieve_user_proxy_agent import RetrieveUserProxyAgent

→ /usr/local/lib/python3.11/dist-packages/pydantic/_internal/_generate_schema.py:502: UserWarning: <built-in function any> is not a Python
warn(
```

```
assistant = autogen.AssistantAgent(
    name = "Assistant",
    llm_config = {"config_list": config_list},
    # AUTOGEN_USE_DOCKER = 0 #
    system_message = "You are a smart agent who reads and understand the given document and extract all the information present in it. Also
)
user_proxy = autogen.UserProxyAgent(
    name = "user_proxy",
    human_input_mode = "NEVER",
    max_consecutive_auto_reply = 1, #1,#
    is_termination_msg=lambda msg: msg.get("content","").rstrip().endswith("TERMINATE"),#is not None and "TERMINATE" in msg["content"],
    code_execution_config = {"work_dir":"web", "use_docker": False},
    llm_config = {"config_list": config_list},
)
```

task = """Extract the personal information, education, experience, skills, projects and certifications from the following resume and create
#"""Can you please tell give me 6 points containing the personal information, education history, work experience, skills, projects and certi

```
result = user_proxy.initiate_chat(
    assistant,
    message=task
)
```

→ user_proxy (to Assistant):

Extract the personal information, education, experience, skills, projects and certifications from the following resume and create it

[autogen.oai.client: 02-24 06:20:03] {583} WARNING - Model llama3-70b-8192 is not found. The cost will be 0. In your config_list, add
WARNING:autogen.oai.client:Model llama3-70b-8192 is not found. The cost will be 0. In your config_list, add field {"price" : [prompt_
Assistant (to user_proxy):

```
```
{
 "personal_info": {
 "name": "Giulia Gonzalez",
 "email": "ggonzalez@email.com",
 "phone": "(123) 456-7890",
 "location": "Detroit, MI",
 "linkedin": "Linkedin",
 "github": "Github"
 },
 "education": {
 "master": {
 "degree": "MS",
 "field": "Computer Science",
 "university": "University of Chicago",
 "duration": "2014 - 2016",
 "location": "Chicago, IL"
 },
 "bachelor": {
 "degree": "B.S.",
 "field": "Computer Science",
 "university": "University of Pittsburgh",
 "duration": "2010 - 2014",
 "location": "Pittsburgh, PA"
 }
 },
 "experience": [
 {
 "title": "Python Developer",
 "company": "DoorDash",
 "duration": "September 2017 - current",
 "location": "Detroit, MI",
 "projects": [
 "Worked on building new Angular components for the customer-facing web app",
 "Collaborated with an agile team of 6",
 "Built extensive test coverage for all new features",
 "Acquired and ingested data to build and maintain data pipelines",
]
 }
]
}
```

```
 "Communicated with internal teams and stakeholders"
]
},
{
 "title": "Python Developer Intern",
 "company": "Knewton",
 "duration": "April 2016 - April 2017",
 "location": "Chicago, IL",
 "projects": [
 "Worked alongside another developer to implement RESTful APIs in Django",
 "Using Selenium, built out a unit testing infrastructure for a client web application",
]
}
```

## OUTPUT JSON CONTAINING PI, EDUCATION, EXPERIENCE, CERTIFICATION &amp; PROJECTS

```
import json
output = result.chat_history[1]['content']

with open(output_file, 'w') as f:
json.dump(output, f, indent=6)

print(output)

{
 "personal_info": {
 "name": "Giulia Gonzalez",
 "email": "ggonzalez@email.com",
 "phone": "(123) 456-7890",
 "location": "Detroit, MI",
 "linkedin": "Linkedin",
 "github": "Github"
 },
 "education": {
 "master": {
 "degree": "MS",
 "field": "Computer Science",
 "university": "University of Chicago",
 "duration": "2014 - 2016",
 "location": "Chicago, IL"
 },
 "bachelor": {
 "degree": "B.S.",
 "field": "Computer Science",
 "university": "University of Pittsburgh",
 "duration": "2010 - 2014",
 "location": "Pittsburgh, PA"
 }
 },
 "experience": [
 {
 "title": "Python Developer",
 "company": "DoorDash",
 "duration": "September 2017 - current",
 "location": "Detroit, MI",
 "projects": [
 "Worked on building new Angular components for the customer-facing web app",
 "Collaborated with an agile team of 6",
 "Built extensive test coverage for all new features",
 "Acquired and ingested data to build and maintain data pipelines",
 "Communicated with internal teams and stakeholders"
]
 },
 {
 "title": "Python Developer Intern",
 "company": "Knewton",
 "duration": "April 2016 - April 2017",
 "location": "Chicago, IL",
 "projects": [
 "Worked alongside another developer to implement RESTful APIs in Django",
 "Using Selenium, built out a unit testing infrastructure for a client web application",
 "Provided project updates to leadership team of 3",
 "Diagnosed issues causing slow speeds in applications",
 "Participated in writing scalable code with a team of 4 interns and 1 developer"
]
 },
 {
 "skills": [
 "HTML/CSS",
 "SQL (PostgreSQL, Oracle)",
 "JavaScript (Angular)",
]
 }
],
}
```

QUERYING THE CV AND FETCHING THE INFO - Currently using a single cv as input, querying and fetching the info from the same.

```
assistant = autogen.AssistantAgent(
 name = "Assistant",
 llm_config = {"config_list": config_list},
 # AUTOGEN_USE_DOCKER = 0 #,
 system_message = "You are a smart agent who reads and understand the given document and provide accurate response to the given question"
)
user_proxy = autogen.UserProxyAgent(
 name = "user_proxy",
 human_input_mode = "NEVER",
 max_consecutive_auto_reply = 1, #1,#
 is_termination_msg=lambda msg: msg.get("content","").rstrip().endswith("TERMINATE"),#is not None and "TERMINATE" in msg["content"],
 code_execution_config = {"work_dir":"web", "use_docker": False},
 llm_config = {"config_list": config_list},
)
```

```
!pip install --upgrade gradio
```

```
Collecting gradio
 Downloading gradio-5.17.1-py3-none-any.whl.metadata (16 kB)
Collecting aiofiles<24.0,>=22.0 (from gradio)
 Downloading aiofiles-23.2.1-py3-none-any.whl.metadata (9.7 kB)
Requirement already satisfied: aiofiles<24.0,>=3.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.7.1)
Requirement already satisfied: fastapi<1.0,>=0.115.2 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.115.8)
Collecting ffmpeg (from gradio)
 Downloading ffmpeg-0.5.0-py3-none-any.whl.metadata (3.0 kB)
Collecting gradio-client==1.7.1 (from gradio)
 Downloading gradio_client-1.7.1-py3-none-any.whl.metadata (7.1 kB)
Requirement already satisfied: httpx>=0.24.1 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.28.1)
Requirement already satisfied: huggingface-hub>=0.28.1 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.28.1)
Requirement already satisfied: jinja2<4.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.1.5)
Collecting markupsafe~=2.0 (from gradio)
 Downloading MarkupSafe-2.1.5-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (3.0 kB)
Requirement already satisfied: numpy<3.0,>=1.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (1.26.4)
Requirement already satisfied: orjson~=3.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (3.10.15)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from gradio) (24.2)
Requirement already satisfied: pandas<3.0,>=1.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.2.2)
Requirement already satisfied: pillow<12.0,>=8.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (11.1.0)
Requirement already satisfied: pydantic>=2.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (2.10.6)
Collecting pydub (from gradio)
 Downloading pydub-0.25.1-py2.py3-none-any.whl.metadata (1.4 kB)
Collecting python-multipart>=0.18 (from gradio)
 Downloading python_multipart-0.0.20-py3-none-any.whl.metadata (1.8 kB)
Requirement already satisfied: pyyaml<7.0,>=5.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (6.0.2)
Collecting ruff>=0.9.3 (from gradio)
 Downloading ruff-0.9.7-py3-none-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (25 kB)
Collecting safethtpx<0.2.0,>=0.1.6 (from gradio)
 Downloading safethtpx-0.1.6-py3-none-any.whl.metadata (4.2 kB)
Collecting semantic-version~=2.0 (from gradio)
 Downloading semantic_version-2.10.0-py2.py3-none-any.whl.metadata (9.7 kB)
Requirement already satisfied: starlette<1.0,>=0.40.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.45.3)
Collecting tomlkit<0.14.0,>=0.12.0 (from gradio)
 Downloading tomlkit-0.13.2-py3-none-any.whl.metadata (2.7 kB)
Requirement already satisfied: typet<1.0,>=0.12 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.15.1)
Requirement already satisfied: typing-extensions~=4.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (4.12.2)
Requirement already satisfied: uvicorn>=0.14.0 in /usr/local/lib/python3.11/dist-packages (from gradio) (0.34.0)
Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages (from gradio-client==1.7.1->gradio) (2024.10.0)
Requirement already satisfied: websockets<15.0,>=10.0 in /usr/local/lib/python3.11/dist-packages (from gradio-client==1.7.1->gradio)
Requirement already satisfied: idna>=2.8 in /usr/local/lib/python3.11/dist-packages (from anyio<5.0,>=3.0->gradio) (3.10)
Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.11/dist-packages (from anyio<5.0,>=3.0->gradio) (1.3.1)
Requirement already satisfied: certifi in /usr/local/lib/python3.11/dist-packages (from https>=0.24.1->gradio) (2025.1.31)
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.11/dist-packages (from https>=0.24.1->gradio) (1.0.7)
Requirement already satisfied: h11<0.15,>=0.13 in /usr/local/lib/python3.11/dist-packages (from httpcore==1.*->https>=0.24.1->gradio)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (3.17.0)
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (2.32.3)
Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.28.1->gradio) (4.67.1)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0->gradio) (2.8)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0->gradio) (2025.1)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas<3.0,>=1.0->gradio) (2025.1)
Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.11/dist-packages (from pydantic>=2.0->gradio) (0.7.0)
Requirement already satisfied: pydantic-core==2.27.2 in /usr/local/lib/python3.11/dist-packages (from pydantic>=2.0->gradio) (2.27.2)
Requirement already satisfied: click>=8.0.0 in /usr/local/lib/python3.11/dist-packages (from typet<1.0,>=0.12->gradio) (8.1.8)
Requirement already satisfied: shellingham>=1.3.0 in /usr/local/lib/python3.11/dist-packages (from typet<1.0,>=0.12->gradio) (1.5.4)
Requirement already satisfied: rich>=10.11.0 in /usr/local/lib/python3.11/dist-packages (from typet<1.0,>=0.12->gradio) (13.9.4)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas<3.0,>=1.0->gradio)
```

T B I <> ⌂ “ ≡ − ☺ ☻

## SAMPLE QUESTIONS

- \* What's the educational background of the cv holder
- \* What's the name of the cv holder
- \* Does the candidate have expertise in python? If so can you please list the relevant experience?

## SAMPLE QUESTIONS

- What's the educational background of the cv holder
- What's the name of the cv holder
- Does the candidate have expertise in python? If so can you please list the relevant experience?

```
import gradio as gr
```

```
def query(question):
 result = user_proxy.initiate_chat(
 assistant,
 message=question
)
 return result.chat_history[1]['content']
```

```
demo = gr.Interface(fn=query, inputs="textbox", outputs="textbox")
```

```
if __name__ == "__main__":
 demo.launch()
```

→ Running Gradio in a Colab notebook requires sharing enabled. Automatically setting `share=True` (you can turn this off by setting `share=False`)

Colab notebook detected. To show errors in colab notebook, set debug=True in launch()  
 \* Running on public URL: <https://d9894ef1fb86c3c60b.gradio.live>

This share link expires in 72 hours. For free permanent hosting and GPU upgrades, run `gradio deploy` from the terminal in the working cell.

question

What's the educational background of the cv holder

Clear

Submit

output

The educational background of the CV holder is:

\* Master's degree in Computer Science from the University of Chicago, completed between 2014 and 2016, in Chicago, IL.  
 \* Bachelor's degree in Computer Science from the University of Pittsburgh, completed between 2010 and 2014, in Pittsburgh, PA.

Flag

Use via API 🔍 · Built with Gradio 🎨 · Settings 🌐

```
import gradio as gr
```

```
def query(question):
 result = user_proxy.initiate_chat(
 assistant,
 message=question
)
```

```

return result.chat_history[1]['content']

demo = gr.Interface(fn=query, inputs="textbox", outputs="textbox")

if __name__ == "__main__":
 demo.launch()

→ Running Gradio in a Colab notebook requires sharing enabled. Automatically setting `share=True` (you can turn this off by setting `share=False`)
Colab notebook detected. To show errors in colab notebook, set debug=True in launch()
* Running on public URL: https://b16dfce55112f83cdb.gradio.live

This share link expires in 72 hours. For free permanent hosting and GPU upgrades, run `gradio deploy` from the terminal in the working cell

```

question

**Clear****Submit**

output

**Flag**

Use via API 🔍 · Built with Gradio 🎨 · Settings⚙️

```
import gradio as gr
```

```

def query(question):
 result = user_proxy.initiate_chat(
 assistant,
 message=question
)
 return result.chat_history[1]['content']

```

```

demo = gr.Interface(fn=query, inputs="textbox", outputs="textbox")

if __name__ == "__main__":
 demo.launch()

```

Running Gradio in a Colab notebook requires sharing enabled. Automatically setting `share=True` (you can turn this off by setting `share=False`)

Colab notebook detected. To show errors in colab notebook, set debug=True in launch()

\* Running on public URL: <https://f45a839ff7cc91e10c.gradio.live>

This share link expires in 72 hours. For free permanent hosting and GPU upgrades, run `gradio deploy` from the terminal in the working directory.

experience?

**Clear**

**Submit**

output

Yes, the candidate has expertise in Python. According to the provided document, the candidate has experience as a Python Developer at DoorDash and as a Python Developer Intern at Knewton. Here are the relevant experiences:

- \* As a Python Developer at DoorDash (September 2017 - current):
  - + No specific Python-related projects are mentioned, but the candidate is working as a Python Developer, which implies proficiency in the language.
- \* As a Python Developer Intern at Knewton (April 2016 - April 2017):
  - + Worked alongside another developer to implement RESTful APIs in Django.
  - + Participated in writing scalable code with a team of 4 interns and 1 developer.