```
pip install docx2txt
```

```
pip install python-docx
```

```
import docx
import os
import re
from sklearn.feature_extraction.text import CountVectorizer
from numpy.lib.function_base import piecewise
from sklearn.metrics.pairwise import cosine_similarity

def extract_resume_information(resume_file_path, log):

    # Load the .docx file
    resume = docx.Document(resume_file_path)

    # Extract text from the resume
    resume_text = ""
    for para in resume.paragraphs:
        resume_text += para.text

    # Extract relevant information from the resume
    resume_info = {}

    # Extract the first paragraph text
    first_paragraph_text = resume.paragraphs[0].text.strip()

    # Extract the name using a regular expression pattern
    name_pattern = r"\b[A-Z][a-z']+ [A-Z][a-z']+\b"
    name = re.findall(name_pattern, first_paragraph_text)

    # Extracted name will be the first element in the list, if found
    if name:
        extracted_name = name[0]
    else:
        extracted_name = None

    # Print the extracted name
    resume_info['name'] = extracted_name

    # Extract email
    for word in resume_text.split():
```

```
    if '@' in word:
        resume_info['email'] = word.strip()
        break
#Phone Number
pattern=re.compile(r'\(?\d{3}\)?[-.\s]?\d{3}[-.\s]?-\d{4}')
phone=pattern.findall(resume_text)
for i in phone:
    resume_info['phone_number'] = i

# Extract skills
skills_keywords = ['skill', 'skills', 'Skills','proficient', 'environment']
skills_section = None
for para in resume.paragraphs:
    if any(keyword in para.text.lower() for keyword in skills_keywords):
        skills_section = para
        break
if skills_section:
    skills_text = skills_section.text
    skills_list = [skill.strip() for skill in skills_text.split('-') if skill.s
    resume_info['skills'] = skills_list

resume = docx2txt.process(resume_file_path)
job_desc = docx2txt.process("/content/sample_data/Job Desc.docx")
text = [resume, job_desc]
cv = CountVectorizer()
count_matrix = cv.fit_transform(text)
#print(cosine_similarity(count_matrix))
matches = cosine_similarity(count_matrix)[0][1]
matches = matches *100
matches = round(matches,2)
resume_info['score'] = matches
log = [resume_info.get('name'),
          resume_info.get('email'),
          resume_info.get('phone_number'),
          resume_info.get('skills'),
          resume_info.get('score')]
logger.append(log)
```

```
import docx2txt
logger = []
folder_path = '/content/sample_data/resume'
all_files = os.listdir(folder_path)

# Loop over each resume file
for resume_file in all_files:
  if resume_file.endswith('.docx'):
    file_path = os.path.join(folder_path, resume_file)
    resume_info = extract_resume_information(file_path, logger)
```

```
import pandas as pd
df_log = pd.DataFrame(logger, columns = ['Name', 'Email', 'Phone Number', 'Skills',
df_log = df_log[(df_log['Email']!='USRC@tiktok.com.')]
df_log
```

	Name	Email	Phone Number	Skills	Score
0	Rachana Choudhary	None	201-539-0815	[TECHNICAL SKILLS]	71.81
1	Srini Bhattiprolu	None	646-598-1583	[Experience in Migrating Code to different Env...	69.90
2	Naveen Yerraguntla	469-605-7255navgaytar@gmail.com	469-605-7255	[12+ years of experience Database Design, Data...	70.44
3	None	None	None	[Experience in configuration of BODS environme...	65.39
4	Padma Shneha	padmashneha7@gmail.com	954-854-7696	[Skills: Python, MySQL, Presto, Hive, Dataswar...	52.63
5	Padma Shneha	None	(954) 854-7696	[Demonstrated strong analytical and statistica...	63.28
6	None	None	513-641-7357	[Proficient in creating solution driven views ...	67.32
7	None	None	None	[Expertise in architecture design of Extractio...	64.47
8	Padma Shneha	padmashneha7@gmail.com	(954) 854-7696	[Skills: Python, MySQL, Presto, Hive, Dataswar...	52.63
9	Srinivas	None	212-300-	[Over 21 years of experience,	52.92

Colab paid products - Cancel contracts here