Smart resume generator

Sample code:

```
Import re
From reportlab.lib.pagesizes import letter
From reportlab.pdfgen import canvas


Def get_user_input():
    Print(“Enter your details for the resume.”)
    Name = input(“Full Name: “)
    Contact = input(“Contact Information (Email/Phone): “)
    Skills = input(“Enter your skills (comma separated): “).split(“,”)
    Experiences = input(“Enter your experiences (comma separated, job title, company name, and duration): “).split(“,”)
    Return name, contact, skills, experiences


Def analyze_job_description(job_desc):
    # This is a simple keyword-based matching. You can enhance this using AI models later.
    Skills_keywords = [“Python”, “Java”, “Communication”, “Leadership”, “Teamwork”, “Project Management”, “SQL”]
    Required_skills = []
    For skill in skills_keywords:
        If re.search(r’\b’ + skill + r’\b’, job_desc, re.IGNORECASE):
            Required_skills.append(skill)
    Return required_skills


Def match_skills(user_skills, job_skills):
Matched_skills=set(user_skills).intersection(set(job_skills))
    Return matched_skills
```

```
Def generate_text_resume(name, contact, matched_skills, experiences):
  Resume = f”””
  Resume for {name}
  Contact: {contact}

  Skills:
  {‘, ‘.join(matched_skills)}

  Experience:
  “””
  For experience in experiences:
    Resume += f”- {experience}\n”

  Return resume

Def generate_pdf_resume(name, contact, matched_skills, experiences, filename=”resume.pdf”):
  C = canvas.Canvas(filename, pagesize=letter)
  Width, height = letter

  c.setFont(“Helvetica”, 12)
  c.drawString(30, height – 40, f”Resume for {name}”)
  c.drawString(30, height – 60, f”Contact: {contact}”)

  c.drawString(30, height – 80, “Skills:”)
  for I, skill in enumerate(matched_skills):
    c.drawString(30, height – 100 – (I * 20), f”- {skill}”)
  c.drawString(30, height – 100 – (len(matched_skills) * 20) – 40, “Experience:”)
  for I, experience in enumerate(experiences):
    c.drawString(30, height – 120 – (len(matched_skills) * 20) – (I * 20), f”- {experience}”)

  c.save()

def main():
  job_desc = input(“Enter the job description for the position you’re applying for: “)
  name, contact, user_skills, user_experiences = get_user_input()
  job_skills = analyze_job_description(job_desc)
  matched_skills = match_skills(user_skills, job_skills)

  # Generate Text Resume
  Text_resume = generate_text_resume(name, contact, matched_skills,
user_experiences)
  Print(“\nYour Customized Resume (Text format):\n”)
  Print(text_resume)

  # Optionally generate PDF resume
  Generate_pdf_resume(name, contact, matched_skills, user_experiences,
“custom_resume.pdf”)
  Print(“\nYour Customized Resume has been saved as ‘custom_resume.pdf’.”)

If __name__ == “__main__”:
  Main()
```