Project : SkillGapAI – Analyzing Resume and Job Post for Skill Gap
Milestone 2: Skill Extraction Using NLP

Introduction

SkillGapAI is an AI-driven platform designed to identify gaps between a candidate's skills and job market requirements. Milestone 2 focuses on automatic skill extraction from resumes and job descriptions using NLP techniques.

Objectives

Extract relevant skills from resumes using NLP. • Identify required skills from job descriptions. • Preprocess and normalize data for accurate comparison. • Build the foundation for an automated skillgap analysis module.

Methodology

The project used the following NLP steps: 1. Data Preprocessing - Tokenization - Stopword removal - Lemmatization - Cleaning special characters & formatting 2. Skill Extraction -Keywordbased matching - Named Entity Recognition (NER) -Domain specific skill dictionaries 3. Skill Comparison - Categorizing technical, soft, and domain skills - Matching extracted skills with job requirements - Identifying missing or weak skill areas

Code:

```python
import streamlit as st
import spacy
import re
import matplotlib.pyplot as plt
from io import BytesIO
st.set_page_config(page_title="SkillGapAI - Milestone 2", layout="wide")
st.markdown(
    """
    <h2 style='color:white; background-color:#117A65; padding:15px; border-radius:10px'>
    🧠 SkillGapAI - Milestone 2: Skill Extraction using NLP
    </h2>
    <p><b>Objective:</b> Extract and classify technical & soft skills separately
    from both Resume and Job Description using spaCy-based NLP pipelines.
    Display structured tags, wanted job skills, and skill distribution charts.</p>
    """,
    unsafe_allow_html=True
)
@st.cache_resource
def load_model():
    try:
        return spacy.load("en_core_web_sm")
    except:
        from spacy.cli import download
        download("en_core_web_sm")
        return spacy.load("en_core_web_sm")
nlp = load_model()
technical_skills = [
    "python", "java", "c++", "sql", "html", "css", "javascript", "react", "node.js",
    "tensorflow", "pytorch", "machine learning", "data analysis", "data visualization",
    "aws", "azure", "gcp", "power bi", "tableau", "django", "flask", "scikit-learn", "nlp"
]
```



```python
]
soft_skills = [
    "communication", "leadership", "teamwork", "problem solving", "time management",
    "adaptability", "critical thinking", "creativity", "collaboration", "decision making"
]
def clean_text(text):
    text = re.sub(r'\s+', ' ', text)
    text = re.sub(r'[^\w\s]', '', text)
    return text.lower().strip()
def extract_skills(text):
    text = clean_text(text)
    found_tech = [skill.title() for skill in technical_skills if skill in text]
    found_soft = [skill.title() for skill in soft_skills if skill in text]
    return list(set(found_tech)), list(set(found_soft))
col_resume, col_jd = st.columns(2)

with col_resume:
    st.markdown("### 👩‍💼 Resume Text")
    resume_text = st.text_area("Paste Resume Content Here:", "", height=250)

with col_jd:
    st.markdown("### 🏢 Job Description Text")
    jd_text = st.text_area("Paste Job Description Content Here:", "", height=250)

if resume_text or jd_text:
    st.markdown("---")
    st.markdown("## 🔍 Skill Extraction Results")
    if resume_text:
        tech_resume, soft_resume = extract_skills(resume_text)
        total_resume = len(tech_resume) + len(soft_resume)
```

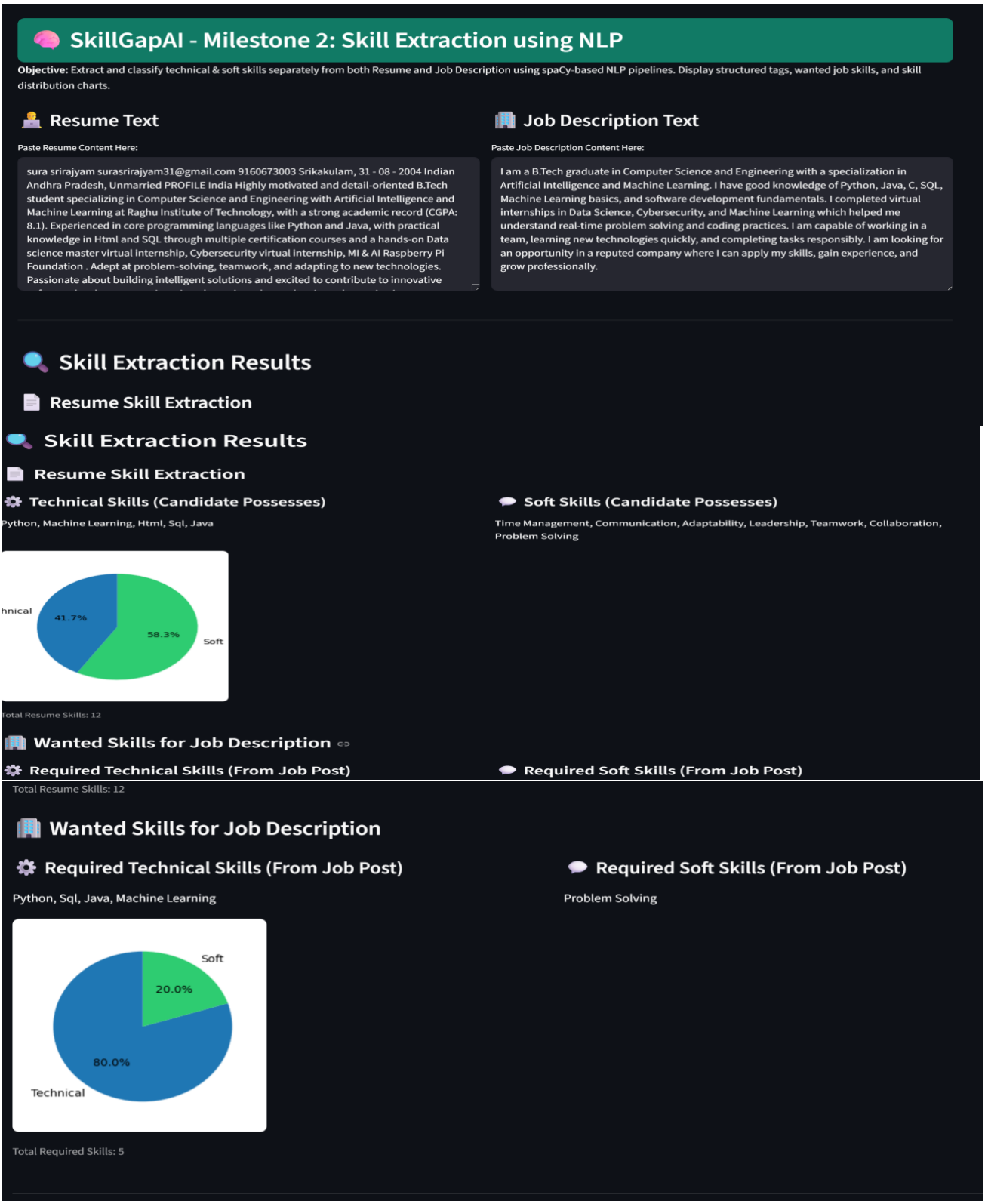
ne2.py > load_model

```
    total_resume = len(tech_resume) + len(soft_resume)
    st.markdown("### 📄 Resume Skill Extraction")
    col1, col2 = st.columns(2)
    with col1:
        st.markdown("#### ⚙️ Technical Skills (Candidate Possesses)")
        st.write(", ".join(tech_resume) if tech_resume else "None found.")
    with col2:
        st.markdown("#### 💬 Soft Skills (Candidate Possesses)")
        st.write(", ".join(soft_resume) if soft_resume else "None found.")
    fig, ax = plt.subplots(figsize=(3, 3))
    labels = ["Technical", "Soft"]
    sizes = [len(tech_resume), len(soft_resume)]
    colors = ["#1F77B4", "#2ECC71"]
    ax.pie(sizes, labels=labels, autopct="%1.1f%%", colors=colors, startangle=90)
    ax.axis("equal")
    buf = BytesIO()
    fig.savefig(buf, format="png")
    st.image(buf)
    st.caption(f"Total Resume Skills: {total_resume}")
if jd_text:
    tech_jd, soft_jd = extract_skills(jd_text)
    total_jd = len(tech_jd) + len(soft_jd)
    st.markdown("### 🏢 Wanted Skills for Job Description")
    col3, col4 = st.columns(2)
    with col3:
        st.markdown("#### ⚙️ Required Technical Skills (From Job Post)")
        st.write(", ".join(tech_jd) if tech_jd else "No technical skills mentioned.")
    with col4:
        st.markdown("#### 💬 Required Soft Skills (From Job Post)")
        st.write(", ".join(soft_jd) if soft_jd else "No soft skills mentioned.")
```

miletone2.py > load_model

```
        with col3:
            st.markdown("#### ⚙️ Required Technical Skills (From Job Post)")
            st.write(", ".join(tech_jd) if tech_jd else "No technical skills mentioned.")
        with col4:
            st.markdown("#### 💬 Required Soft Skills (From Job Post)")
            st.write(", ".join(soft_jd) if soft_jd else "No soft skills mentioned.")
        fig2, ax2 = plt.subplots(figsize=(3, 3))
        labels2 = ["Technical", "Soft"]
        sizes2 = [len(tech_jd), len(soft_jd)]
        colors2 = ["#1F77B4", "#2ECC71"]
        ax2.pie(sizes2, labels=labels2, autopct="%1.1f%%", colors=colors2, startangle=90)
        ax2.axis("equal")
        buf2 = BytesIO()
        fig2.savefig(buf2, format="png")
        st.image(buf2)
        st.caption(f"Total Required Skills: {total_jd}")
else:
    st.info("Please paste resume and/or job description text to extract skills.")
st.markdown("---")
st.markdown(
    "<p style='text-align:center; color:gray;'>Milestone 2 • Skill Extraction using NLP • SkillGapAI Project •
    unsafe_allow_html=True
)
```

Output


SkillGapAI - Milestone 2: Skill Extraction using NLP
Objective: Extract and classify technical & soft skills separately from both Resume and Job Description using spaCy-based NLP pipelines. Display structured tags, wanted job skills, and skill distribution charts.
Resume Text
Paste Resume Content Here:
sura srirajyam surasrirajyam31@gmail.com 9160673003 Srikakulam, 31 - 08 - 2004 Indian Andhra Pradesh, Unmarried PROFILE India Highly motivated and detail-oriented B.Tech student specializing in Computer Science and Engineering with Artificial Intelligence and Machine Learning at Raghu Institute of Technology, with a strong academic record (CGPA: 8.1). Experienced in core programming languages like Python and Java, with practical knowledge in Html and SQL through multiple certification courses and a hands-on Data science master virtual internship, Cybersecurity virtual internship, MI & AI Raspberry Pi Foundation . Adept at problem-solving, teamwork, and adapting to new technologies. Passionate about building intelligent solutions and excited to contribute to innovative
Job Description Text
Paste Job Description Content Here:
I am a B.Tech graduate in Computer Science and Engineering with a specialization in Artificial Intelligence and Machine Learning. I have good knowledge of Python, Java, C, SQL, Machine Learning basics, and software development fundamentals. I completed virtual internships in Data Science, Cybersecurity, and Machine Learning which helped me understand real-time problem solving and coding practices. I am capable of working in a team, learning new technologies quickly, and completing tasks responsibly. I am looking for an opportunity in a reputed company where I can apply my skills, gain experience, and grow professionally.
Skill Extraction Results
Resume Skill Extraction
Skill Extraction Results
Resume Skill Extraction
Technical Skills (Candidate Possesses)
Python, Machine Learning, Html, Sql, Java
Soft Skills (Candidate Possesses)
Time Management, Communication, Adaptability, Leadership, Teamwork, Collaboration, Problem Solving
hnical
41.7%
58.3%
Soft
Total Resume Skills: 12
Wanted Skills for Job Description
Required Technical Skills (From Job Post)
Required Soft Skills (From Job Post)
Total Resume Skills: 12
Wanted Skills for Job Description
Required Technical Skills (From Job Post)
Python, Sql, Java, Machine Learning
Required Soft Skills (From Job Post)
Problem Solving
Soft
20.0%
80.0%
Technical
Total Required Skills: 5

Results

• Successfully extracted technical and soft skills from multiple resumes. • Identified jobspecific skills from various job descriptions. • Detected clear skill gaps between candidate capabilities and job expectations. • Provided structured skill lists useful for the next milestone (recommendation system).

Conclusion

Milestone 2 successfully established a strong technical base for SkillGapAI. The NLP-driven skill extraction system accurately identifies skills from unstructured text and highlights gaps when compared to job requirements. The output of this milestone enables: • Datadriven career recommendations • Personalized learning pathways • Automated resume enhancement tools This milestone plays a crucial role in building an intelligent and scalable skillgap analysis platform.

Future Enhancements

- Integrating deep-learning-based contextual skill detection models.
- Adding support for multilingual resumes.
- Improving accuracy with industry-standard skill ontologies.
- Building a realtime recommendation engine based on missing skills.