

COIMBATORE INSTITUTE OF TECHNOLOGY
(Autonomous institution affiliated to Anna University, Chennai)

COIMBATORE - 641014



19MAMEL07 – GENERATIVE AI LAB

PROJECT DOCUMENTATION

**SKILLMORPH AI: PERSONALIZED CAREER SIMULATOR
WITH AI FEEDBACK AND TALKING AVATAR**

NAME: MITHUNSANKAR S

REG.NO: 71762134028

ABSTRACT

SkillMorph AI is an advanced career simulation platform powered by Generative AI. It provides personalized skill feedback, learning paths, and visual insights based on a user's resume, GitHub, and LinkedIn profiles. The platform integrates a text-to-speech system and a deep learning-based talking avatar (SadTalker) to create an engaging audio-visual report. By combining LLMs, API integrations, and real-time audio-visual generation, the system delivers a powerful personalized upskilling experience for users aiming to bridge the gap between their current and desired job roles.

The project uses the Google Gemini API for resume and skill feedback generation, the GitHub and LinkedIn APIs for live portfolio fetching, and SadTalker for animating avatars based on user feedback. Built with Streamlit, the platform offers a seamless, interactive interface accessible to both technical and non-technical users.

PRE-TRAINED MODELS AND THEIR ARCHITECTURE

1. Google Gemini API (Generative AI LLM)

Architecture Overview

- **Type:** Transformer-based instruction-tuned large language model (LLM)
- **Input:** Resume content and prompt-based queries
- **Output:** Feedback text, gap analysis, learning recommendations
- **API Access:** Gemini Pro via Google AI Studio

Training

- Trained on a mixture of web-scale datasets including text, code, and instructions.
- Fine-tuned with instruction-response pairs to follow structured queries.
- Supports zero-shot and few-shot prompting for skill matching and suggestion generation.

Parameters

- 500B+ parameters (Gemini 1.5 scale, depending on tier)
- Context window: up to 1 million tokens (Gemini Ultra)
- Language modeling + instruction-following head

Implementation in Project

- Resume text is extracted and structured into a JSON prompt
- Gemini responds with:
 - Skill strengths
 - Missing job-aligned skills
 - Personalized weekly plan
 - Motivation-based feedback
- Used in:
 - resume_reviewer.py
 - job_skill_matcher.py
 - weekly_planner.py

Significance

- Enables **human-like personalized feedback** without requiring manual review
- Supports **real-time resume enhancement** suggestions
- Powers the **core intelligence layer** of SkillMorph AI

2. SadTalker (Talking Avatar Generator)

Architecture Overview

- Pipeline-based deep learning model for lip-synced face animation
- Modules:
 - Facial landmark detector
 - 3DMM-based face pose and expression predictor
 - Audio-to-expression coefficient network
 - Face renderer (based on face-vid2vid)
- Input: Avatar image (static) and audio
- Output: Video of avatar talking with synchronized lip and head motion

Training

- Trained on large facial animation datasets like VoxCeleb, HDTF
- Components trained:
 - **Audio2Pose & Audio2Exp:** supervised learning from landmark sequences
 - **Face Renderer:** trained with adversarial loss (GAN)
- 3D Morphable Model (3DMM) for facial structure reference

Parameters

- Model Size: ~1.4GB (all safetensor files)
- Expression Scale: tunable (default 1.0)
- Resolutions Supported: 256x256, 512x512
- Optimized for CPU/GPU execution

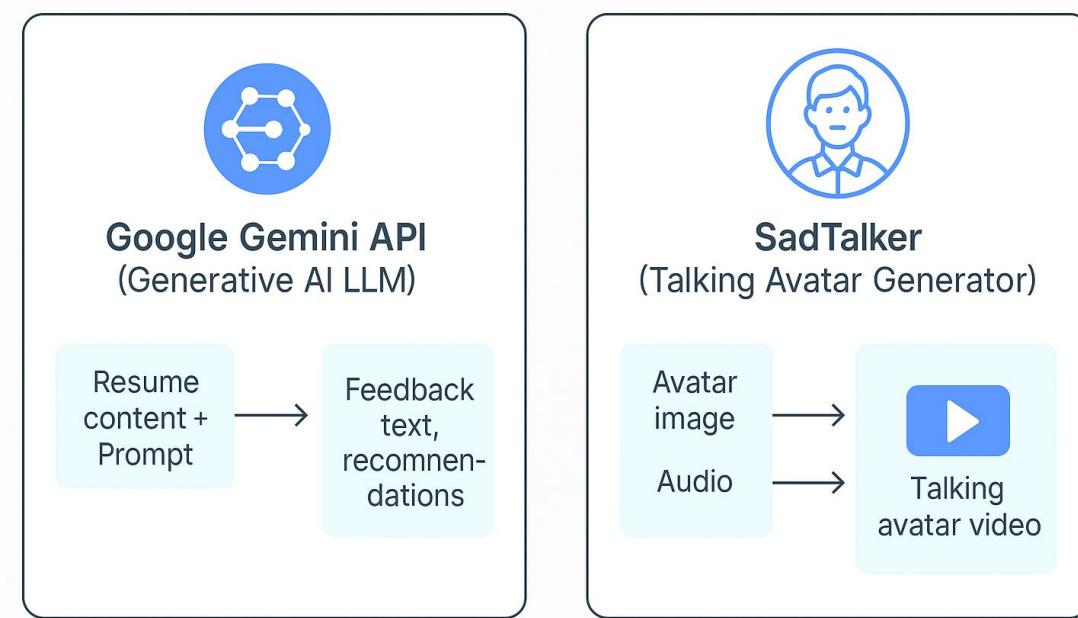
Implementation in Project

- User uploads a static image
- Audio is generated from resume feedback using gTTS or Gemini TTS
- Avatar is animated via:
 - `run_sadtalker()` function
 - `SadTalker's inference.py` script
- Final .mp4 video is rendered and playable/downloadable in Streamlit

Significance

- Adds **visual and emotional feedback** to an otherwise text-only report
- Boosts user engagement through **realistic talking avatars**
- Makes SkillMorph a **next-gen career guidance experience**

Pretrained Models and their Architecture



SAMPLE CODE (Resume + Avatar Integration)

```
from pydub import AudioSegment

def run_sadtalker(image_path, audio_path,
output_path="sadtalker_output.mp4", max_duration=10):

    from pathlib import Path

    import subprocess, shutil, os

    source_image_dir = "SadTalker/examples/source_image"
    audio_dir = "SadTalker/examples/driven_audio"
    result_dir = "SadTalker/results"

    # Create directories
    os.makedirs(source_image_dir, exist_ok=True)
    os.makedirs(audio_dir, exist_ok=True)
    os.makedirs(result_dir, exist_ok=True)

    # Clear old files
    for f in Path(source_image_dir).glob("*"): f.unlink()
    for f in Path(audio_dir).glob("*"): f.unlink()

    # Copy image
    shutil.copy(image_path, os.path.join(source_image_dir, "avatar.png"))

    # Trim audio to 10 seconds max
    audio = AudioSegment.from_file(audio_path)
    trimmed_audio = audio[:max_duration * 1000]
    trimmed_audio_path = os.path.join(audio_dir, "audio.mp3")
    trimmed_audio.export(trimmed_audio_path, format="mp3")
```

```
# Run SadTalker inference

command = [
    "python", "inference.py",
    "--driven_audio", "examples/driven_audio/audio.mp3",
    "--source_image", "examples/source_image/avatar.png",
    "--result_dir", "results",
    "--enhancer", "gfpgan",
    "--preprocess", "extcrop",
    "--size", "512"
]

subprocess.run(command, cwd="SadTalker")

# Get result

video_files = [f for f in os.listdir(result_dir) if f.endswith(".mp4")]

if not video_files:

    raise Exception("✖ No video was generated by SadTalker.")

latest_video = max(
    [os.path.join(result_dir, f) for f in video_files],
    key=os.path.getmtime
)

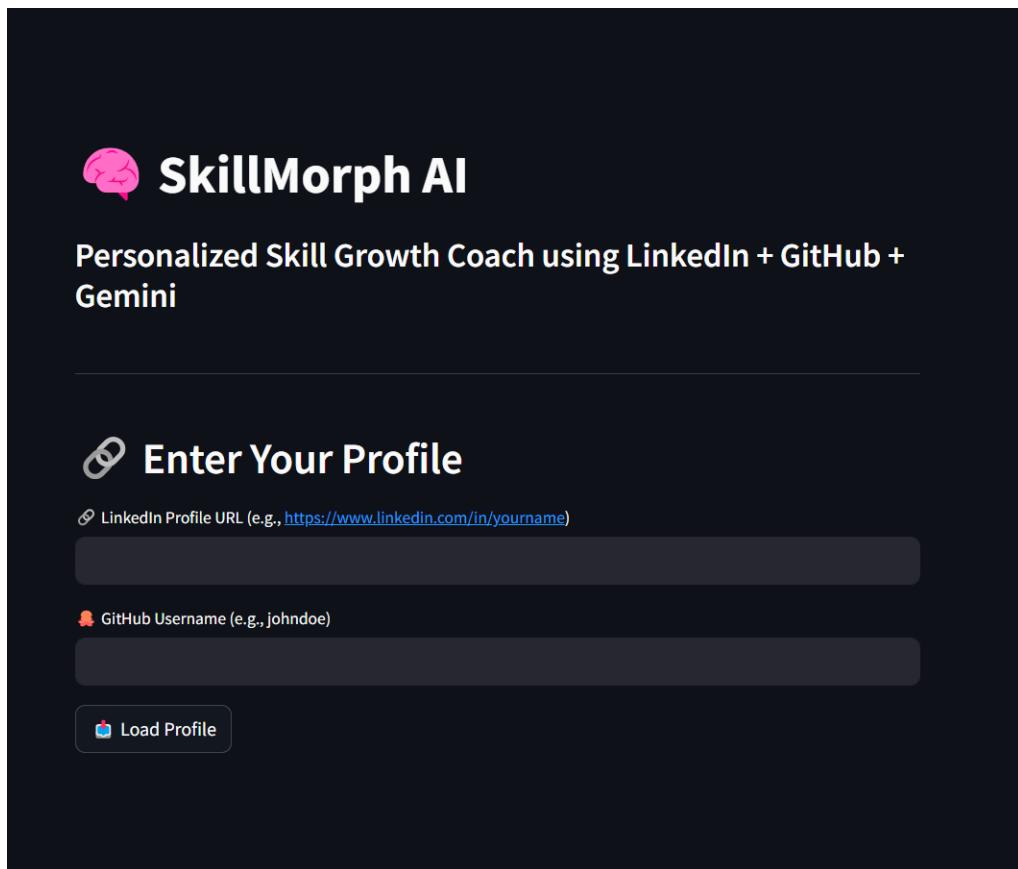
shutil.copy(latest_video, output_path)

return output_path
```

OUTPUTS

- PDF Resume with AI Suggestions
- Weekly Upskilling Planner
- Talking Avatar Video with Feedback
- Master Career Report

UI SNAPSHOT (Streamlit)



 LinkedIn Profile URL (e.g., <https://www.linkedin.com/in/yourname>)

<https://www.linkedin.com/in/muhilan-g-s-b35283276>

 GitHub Username (e.g., johndoe)

Muhilan23

 Load Profile

Profile Overview



 Name: Muhilan G S B35283276

 LinkedIn: <https://www.linkedin.com/in/muhilan-g-s-b35283276>

 GitHub Username: [Muhilan23](#)

 Repositories: 1

 Languages: Jupyter Notebook

 Step 2: Generate Your AI Skill Report



analysis are good starting points.

5. **Motivational note:** Building a strong online presence and a portfolio of projects takes time and consistent effort. Focus on making incremental progress each week. Every line of code you write, every project you complete, and every contribution you make strengthens your skills and increases your visibility to potential employers. Stay persistent, and you will see results.

 Download Report as PDF

 Step 3: Skill Gap Analyzer



 Step 4: Weekly Upskilling Planner



 Step 5: Upload Resume for AI Review

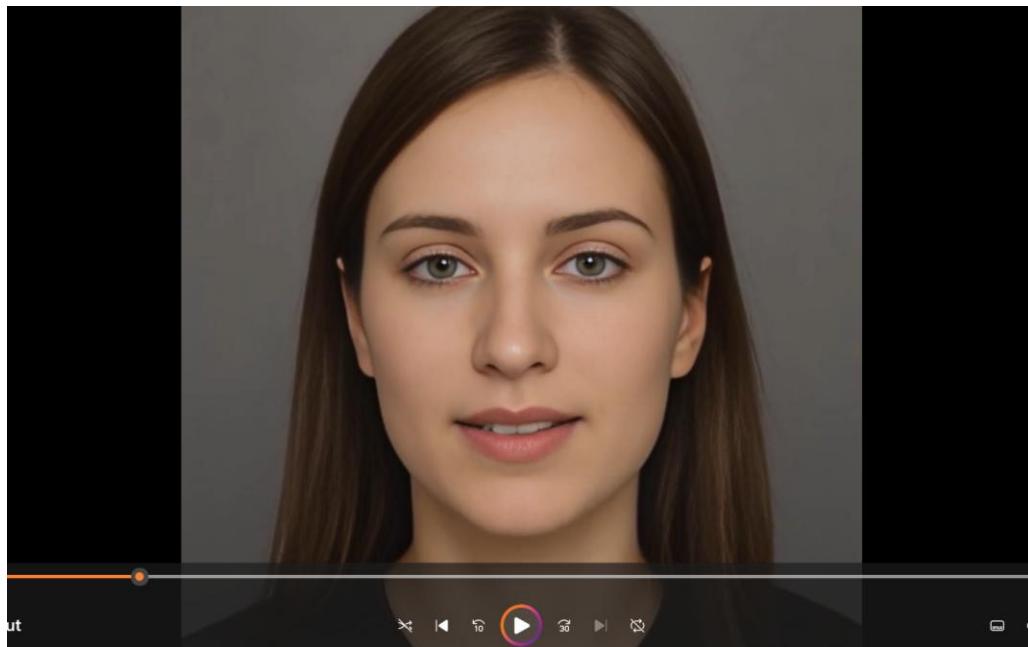


 Step 6: Generate Master Career Report



 AI Talking Avatar (SadTalker v0.0.2)





CONCLUSION

SkillMorph AI demonstrates the effective fusion of Generative AI and personalized career development. By combining real-time data from professional profiles, resume analysis through Gemini API, and interactive voice/video feedback via SadTalker, the platform empowers users to understand their skill gaps and take actionable steps toward career advancement. The tool is especially valuable for students, job-seekers, and professionals seeking personalized career coaching at scale.