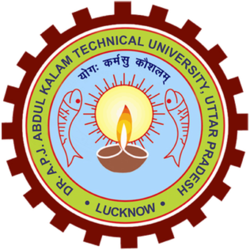
**A.P.J ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW**



**Mini-Project Report on “Project Title”**

AI RESUME & COVER LETTER

Project was carried out by

**NAME:** Swapnil Kesharwani

**UNI. ROLL NO.:** 2300101530170

**UNITED COLLEGE OF ENGINEERING & RESEARCH**

**NAINI, PRAYAGRAJ**



**INDEX:**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Topic | Page No. | Teacher’s Sign. |
| 1. | Introduction |  |  |
| 2. | Objective |  |  |
| 3. | Features |  |  |
| 4. | Core idea |  |  |
| 5. | Architecture Overview |  |  |
| 6. | Technologies |  |  |
| 7. | Future Enhancements |  |  |
| 8. | Workflow |  |  |
| 9. | Supported Languages |  |  |
| 10. | Project Deliverables |  |  |
| 11. | Conclusion |  |  |

**Introduction:**

The job market is becoming increasingly competitive, and having a professionally crafted resume and cover letter is essential. However, many job seekers struggle with formatting, structuring, and writing impactful content. This project introduces an AI-based Resume and Cover Letter Generator that uses Natural Language Processing (NLP) and Large Language Models (LLMs) to automate the creation of personalized documents based on user input.

**Objective:**

To develop a web-based application that uses artificial intelligence (LLMs) to generate **personalized**, **professionally formatted** resumes and cover letters based on user-provided input such as name, job role, skills, experience, and career goals. The tool aims to **simplify** and **speed up** the job application process, especially for users who struggle with writing professional content.

**Features**:

* Auto-generation of professional resumes
* Customizable templates and tone (formal, semi-formal)
* Cover letter with job-specific context
* Downloadable PDF export option
* Option to revise or regenerate

**Core Idea:**

Job seekers often face difficulty in writing impactful resumes and cover letters tailored to each role. This application leverages **AI language models (LLMs)** to:

* Understand user inputs (skills, goals, experience).
* Automatically craft high-quality, customized resume and cover letter content.
* Provide instant and editable outputs via a simple web interface.

The solution combines the **power of transformers** and **ease of use of Gradio** to deliver a one-stop professional writing tool.

**Architecture Overview:**

User Input (via Gradio Interface) -> Input Preprocessing -> Prompt Engineering -> Language Model (Transformers + torch) -> Generated Resume & Cover Letter -> Output Display (Gradio Web UI).

**Technologies:**

| Component | Technology / Library | Purpose |
| --- | --- | --- |
| 🧠 Language Model | transformers >= 4.28 | Generate text using pre-trained models (e.g., GPT-2, Falcon) |
| 🔧 Deep Learning | torch >= 1.9 | Backend computation for LLM inference |
| 🧱 Tokenizer | tokenizers >= 0.13.3, sentencepiece >= 0.1.95 | Efficient tokenization for model input/output |
| 📦 Serialization | protobuf >= 3.20.0 | Handles serialized model data (required by some transformer models) |
| 🌐 Web UI | Gradio | Easy-to-use frontend for users to interact with the AI |
| 📁 (Optional) PDF Export | reportlab / weasyprint | Convert generated text into downloadable resumes/letters |

**Future Enhancements:**

• **Template Selector** (modern, minimalist, corporate styles).

• **Upload Existing Resume** for AI improvement.

• **PDF Download** with styling.

• **Multi-language support.**

• **Job Description Matching** for even better targeting.

**Workflow:**

This is the step-by-step process from user input to output generation:

1. User Input via Web UI (Gradio)

* The user enters:
  + Name
  + Skills (comma-separated)
  + Job Role
  + Experience (years + description)
  + Career Goals

2. Prompt Construction

* The app dynamically constructs a well-structured prompt combining all inputs in natural language, optimized for the language model.

3. AI Generation (LLM Inference)

* The prompt is passed to the transformer-based model (e.g., GPT-2 or T5).
* The model processes the input and returns generated text for:
  + Resume Content
  + Cover Letter Content

4. Post-Processing

* The raw output is formatted or split into:
  + Resume
  + Cover Letter
* Optionally, the output can be styled or exported to PDF.

5. Display Output

* Both resume and cover letter are shown on the Gradio interface.
* User can copy or save content.

**Supported Languages:**

By default, the application uses English since most pre-trained LLMs like GPT-2, Flan-T5, or Falcon are fine-tuned on English corpora.

You can add multilingual support (optional) by:

* Using multilingual models like mBART, mT5, or XLM-R
* Adding a language selector in the UI and adapting the prompt accordingly

Currently Supported:

* English (Primary Language)

Optional Support with Model Change:

* 🇫🇷 French
* 🇩🇪 German
* 🇮🇳 Hindi
* 🇪🇸 Spanish  
  *(based on multilingual model support)*

***Project Deliverables:***

| Deliverable | Description |
| --- | --- |
| 🧠 AI Model Code | Python script using transformers, loading a pre-trained LLM for text generation |
| 🌐 Gradio Frontend | Interactive UI for users to input their data and receive generated results |
| 📄 Resume & Cover Letter Output | Dynamically generated text output, tailored to user inputs |
| 🗂️ Project Repository | Complete GitHub repo with README, requirements.txt, and app.py |
| 📋 Documentation | Includes architecture, setup guide, usage instructions, and tech stack |
| 📁 (Optional) PDF Export Feature | Resume/Cover Letter can be downloaded as styled PDF files |
| 📝 README.md File | Overview of project, installation steps, and usage guide for end users |
| 📊 (Optional) Presentation Slides | PPT or PDF for explaining project overview, architecture, and demo |
| ✅ Test Cases (Optional) | Sample inputs and expected outputs to test the app functionality |

**Conclusion:**

The AI Resume & Cover Letter Generator significantly simplifies the job application process. It leverages the power of LLMs to help users create high-quality documents quickly and efficiently, ensuring they put their best foot forward in job applications.